



Agent-based Group Decision Making for Ordinal Classification : Preference Elicitation and Negotiation-aggregation

Ons Nefla

► To cite this version:

Ons Nefla. Agent-based Group Decision Making for Ordinal Classification : Preference Elicitation and Negotiation-aggregation. Other [cs.OH]. Université Paris sciences et lettres; emlyon business school, 2021. English. ⟨NNT : 2021UPSLD026⟩. ⟨tel-03631128⟩

HAL Id: tel-03631128

<https://theses.hal.science/tel-03631128v1>

Submitted on 5 Apr 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



HAL Authorization



THÈSE DE DOCTORAT
DE L'UNIVERSITÉ PSL

Préparée à l'université Paris-Dauphine et emlyon business school

Agent-based Group Decision Making for Ordinal Classification : preference elicitation and negociation-aggregation

Soutenue par

Ons NEFLA

Le 30/06/2021

École doctorale n°543

ED de Dauphine

Spécialité

Informatique

Composition du jury de soutenance :

Amal EL FALLAH SEGHHROUCHNI

Professeure,

Sorbonne Université

Rapporteuse

Patrick MEYER

Professeur,

IMT Atlantique

Rapporteur

Nicolas MAUDET

Professeur,

Sorbonne Université

Examineur

Marie-Hélène ABEL

Professeure,

Université de Technologie de Compiègne

Présidente

Alexis TSOUKIÀS

Directeur de recherche,

Université Paris-Dauphine

Examineur

Imène BRIGUI

Professeure associée,

emlyon business school

Examinatrice

Paolo VIAPPIANI

Chercheur CNRS,

Sorbonne Université

Examineur

Meltem ÖZTÜRK

Maître de Conférences (HDR),

Université Paris-Dauphine

Directrice

Contents

1	Decision Aiding and Preference elicitation	39
1.1	Introduction	39
1.2	Decision Aiding	40
1.2.1	Decision Aiding Process	41
1.2.2	Preference Modeling	43
1.2.3	Multiple Criteria Decision Aiding	48
1.3	Preference elicitation	56
1.3.1	Preference Elicitation for ordinal classification methods	58
1.3.2	Incremental approach	61
1.4	Synthesis	63
2	Group Decision Making and Automated Negotiation	65
2.1	Introduction	65
2.2	Group decision making	66
2.2.1	Group decision making for ranking problems	67
2.2.2	Group decision making for ordinal classification problems	71
2.3	Basics of Automated negotiation	74
2.3.1	Negotiation object	75
2.3.2	Participants	75
2.3.3	Negotiation protocol	77
2.3.4	Negotiation strategies	82
2.4	Synthesis	85

3	Pre-negotiation process: Interactive Elicitation of an ordinal classification model	87
3.1	Introduction	88
3.2	MR-Sort example	89
3.3	Learning a MR-Sort model from assignments	90
3.3.1	Possible Categories	91
3.3.2	Maximum margin optimization	92
3.4	Incremental MR-Sort	94
3.4.1	Main Framework	94
3.4.2	Question selection strategies	96
3.5	Experiments	99
3.6	Conclusions	104
4	Group-based Ordinal Classification based on a Negotiation Process	107
4.1	Introduction	108
4.2	Multi-agent classification architecture	109
4.3	Classification process	110
4.4	Agents' description	112
4.4.1	User agents	114
4.4.2	Mediator agent	116
4.5	Illustration	119
4.5.1	Context	119
4.5.2	Application of the developed negotiation process	120
4.6	Experimental Approach	123
4.6.1	Evaluation criteria	125
4.6.2	Experimental results	128
4.7	Sensitivity analysis	132
4.7.1	Experimental process	132
4.7.2	Experimental settings	133
4.7.3	Observed indicators	135
4.7.4	Experimental results	136

4.8 Behavioural analysis	141
4.9 Conclusion	144
Conclusion & Perspectives	147
Bibliography	164

List of Figures

1	Architecture de classification multi-agents	33
2	Group-classification process	35
3.1	AE ₂ based on number of questions with $m = 100$, $n = 5$, $p = 2$, $p = 3$, $p = 4$, $ LS = 1$ and 50 iterations	101
3.2	AE ₁ based on number of questions with $m = 300$, $n = 6$, $p = 4$, $ LS =1$, $\lambda = 0.7$ and 50 iterations	102
3.3	AE ₂ based on number of questions with $m = 300$, $n = 6$, $p = 4$, $ LS =1$, $\lambda = 0.7$ and 50 iterations	102
3.4	Classification error AE ₁ based on number of questions with $m = 100$, $n = 4$, $p = 2$, $p = 3$, $p = 4$, $ LS = 1$ and 50 iterations	103
3.5	AE ₂ based on number of questions with $m = 100$, $n = 4$, $p = 2$, $p = 3$, $p = 4$, $ LS = 1$ and 50 iterations	103
3.6	AE ₁ based on number of questions with $m = 100$, $n = 5$, $p = 2$, $p = 3$, $p = 4$, $ LS = 1$ and 50 iterations	104
4.1	Architecture de classification multi-agents	111
4.2	Group-classification process	113
4.3	Average Dissatisfaction (AvD)	129

4.4	Maximum Dissatisfaction ($MaxD$)	130
4.5	Dispersion of Dissatisfaction Index (DD)	131
4.6	Information Privacy Index (IPI)	132
4.7	Sensitivity analysis of group classifications based on MS metric and the variation of evaluation indicators with respect to α	137
4.8	The performed sensitivity analysis of group classifications (based on MM metric) and the variation of evaluation indicators regarding order of requests' variation	140
4.9	The performed sensitivity analysis of group classifications (based on MM metric) and the variation of evaluation indicators regarding acceptance threshold's	140
4.10	Behavioural analysis of evaluation indicators with respect to inflexi- ble agents' percentage	142
4.11	Behavioural analysis results over the process' number of rounds as function of inflexible agents' percentage	143

List of Tables

3.1	Performance of limit profiles b^1, b^2, b^3, b^4 and alternatives x, y, z, t, u .	90
3.2	Comparison between alternatives and profiles and final assignments.	90
3.3	Number of possible categories for the example of hotel categorization.	97
3.4	The score obtained by the different items in the hotel categorization example using the heuristics sum-of-maring (left) and entropy (right)	98
4.1	Agent performatives	113
4.2	The classification of the alternatives $x \in X$ by agents $a_i \in \mathcal{A}$ to the category $cl(a_i, x)$	119
4.3	Negotiation strategies of user-agents a_1, a_2, a_3, a_4	119
4.4	The subutility values $u_i(d(c_i, c_j))$, $fl_i(t)$, and $f_i(r)$ associating user-agent a_i to respectively the deviation $d(c_i, c_j)$, the current time t and number of concessions made previously r	120
4.5	The user agents' feedback and the common classification of x_1 after round 1	122
4.6	The user-agents' feedback and the common classification of x_2 after round 2	122

4.7	The user-agents' feedback and the common classification of x_3 after round 3	122
4.8	The user-agents' feedback and the common classification of x_4 after round 4	122
4.9	The user agents' feedback and the common classification of x_2 after round 5	122
4.10	Distribution of expert classifications on a sample of 10 travel destinations (the total number of experts is 38).	123
4.11	Median and mode values for a sample of travel destinations according to the distribution of expert classification	124
4.12	The average, min, max, standard deviation and confidence interval of the performed sensitivity analysis regarding the process results are functions of the majority threshold	138
4.13	The average, min, max, standard deviation and confidence interval of the performed sensitivity analysis regarding the process results as function of acceptance threshold's	141
4.14	The average, min, max, standard deviation and Confidence interval of the performed sensitivity analysis regarding the process results regarding order of requests	141

Acknowledgements

When I started to write the acknowledgement, I asked myself a question: To whom I owe the quality of this thesis?

First of all, I would like to express my gratitude to my supervisors Meltem ÖZTÜRK, Imène BRIGUI and Paolo VIAPPIANI for teaching me how to sail in the calm and the agitated seas of the Ph.D thesis: You have been present all along the thesis with your precious advices, your encouragements, your inspirations and your constructive remarks. A thesis is like a roller coaster, there are always ups and downs, in the moments of breakdown, you helped me to overcome all the difficulties that came on my way. Thanks to you, I had the chance to learn how to conduct fundamental and experimental research. Lucky are the new PhD students who will work with you.

I would like to thank Professors Amal EL FALLAH SEGHRUCHNI and Patrick MEYER for the interest they have shown in my research work by honoring me to accept the task of reporting this thesis. I would also like to thank Professors Nicolas MAUDET, Marie-Hélène ABEL, Alexis TSOUKIÀS, for their interest in my research work and for agreeing to participate in the evaluation of this thesis.

My sincere gratitude goes to the professors of the LAMSADE laboratory for their academic assistance, their friendship and their encouragement. In particular, I would like to thank Professors Yann CHEVALEYRE, Denis CORNAZ and Denis BOUYSSOU for following my progress during each year of my thesis.

I would like to thank the administration of LAMSADE, particularly Marie Clotilde Quinio, Olivier Rouyer and Eleni Zymari for all their hard work. Thanks to them we do not feel the heaviness of the administration. I would like also to thank all my colleagues and friends in LAMSADE. My Oussama, Raja, Hiba, Mehdi Khosravian, Mehdi Acheli, Hossein, Amin, Beatrice, Manel, Celine, Pierre, Axel, Ahlem, Hajer, Diana, Khalil, Yassine, Felipe, Thomas, Nicolas, Charles, Annaelle. Thank you for the time we spent together, full of happiness and for constructive debates.

Thanks to LAMSADE, I was lucky to meet my charming life partner Oussama RABOUN, one of the smartest persons I met. Such an inspiring and lovely person. Thank you for supporting me throughout my thesis and being the proof that a life partner is for the best and the worse. Your love had a major contribution in my personal and professional progression. I am very excited to share the challenges of life with your wonderful company. I would like also to thank your wonderful family, especially my aunty Soumia, for their encouragement and moral supports.

I would like to express my humble gratitude to the research team at emlyon business school for their scientific assistance, friendship and encouragement. In particular, I would like to thank Sara BELGITHI, Rachid ALAOUI, Christelle AUBERT HASSOUNI and Benjamin HUYBRECHTS for their cheerfulness and constructive discussions. You have as well illuminated the work environment with your enthusiasm.

Special thanks goes to Professor Achref JABEUR TELMOUDI and Professor Imed KACEM. You were the first researchers who inspired me to join the research world.

I thank all my friends that I met throughout my life: Fadia, Emna, Yosra, Nedra, Marwa, Ameni, Safwene, Tesnim, Fadoua, Hamza, Amine, Asmae, Aya, Abbes, Chaimae, Omar, Anis, Amal, Yosra, Abdeslem and Saber. You have taught me that in life happiness multiplies when it is shared.

This part of the acknowledgements would be incomplete if I did not express my gratitude to my family. To my dear father Rachid, Thank you for always believing

in me. I will never forget your words: "My dear girl, one day you will conquer the world". This overconfidence is my source of motivation. I hope I can live up to your confidence. To my dear mother Samia, thank you for being present even in the most difficult moments and for pushing me to advance. I know that my success is your success and I know that the best gift I could possibly offer you is this simple dedication in my thesis. I hope I will always make you proud. I am fortunate to have two older sisters Rihab and Refka who are examples of success to me. Thank you for being my source of inspiration and determination, for always being behind me when I needed you. Thank you for giving birth to 5 mini creatures Fares, Abd Errahmen, Abd Elaziz, Chahd et Meriam to whom I approve an endless love. A special thanks to my brother-in-law Farid. Thank you for being present in my life as if you are my biological brother. I would like to thank my aunts, my cousins and especially my uncle Foued whom I consider as my second father. I cannot conclude this part without thanking my grandparents Fawzia and Abd Elaziz. You were for me the proof that love is ephemeral, your love survives on me although you no longer exist in this life (peace to your souls).

Introduction

Making decision is a natural act in human lives. We decide what we eat for lunch, where we spend our vacations, which professional career we may envisage, etc. In many contexts, this task presents an important issue. For instance, the managers of a company make strategic decisions to launch new investments. Such decisions may impact the future of the company and entail a significant risk. Decision aiding aims at formalizing decision problems and structuring the decision maker's preferences to assist her in making the most appropriate decision. In this thesis, among the problems of decision aiding, we consider the *ordinal classification* problem, where the aim is classifying objects into ordered categories. For instance, classifying hotels into the following ordered categories such as 1-star, 2-stars, 3-stars, etc.

Ordinal classification problems can be considered in the context of multiple criteria decision, where objects are judged according to several points of view called *criteria*. These criteria are not reducible to a single criterion and may be conflicting. As a result, the decision problem becomes more complex. For example, to classify a hotel in the ordered categories, we may consider the following points of view: comfort, price, quality of restoration, etc. Multiple criteria decision aiding process proposes several models to deal with the ordinal classification problems. The parameters of these models, such as the weights of criteria that reflect the importance of criteria for the decision maker, take into account the decision maker's preferences in order to provide her a result corresponding to her preferences. For the example of hotel classification, a decision maker may give more importance to the price than the comfort. Preference elicitation models have been conceived

to better particularize a decision model for a given decision maker. They ensure that the parameters of this model correspond to the preferences of the decision maker and reflect her way of deciding. In the context of ordinal classification, the decision maker may be asked directly about the parameters values of the model. However, this approach requires a deep understanding of the decision model by the decision maker, which may be beyond her cognitive capacities. Instead of directly asking the decision maker about the values of parameters, we may try to retrieve examples representing her preferential judgments and thus she may be asked about her subjective preferences for classifying objects into categories. Then, the parameters of the model will be inferred based on her responses. However, asking the decision maker multiple questions is not an easy task, especially in many practical contexts including applications on the web (such as e-learning or e-commerce platforms). Then, the question is the following: *how can we elicit the preferences of a decision maker in a context of multiple criteria ordinal classification while asking her a limited number of questions?*

The complexity of ordinal classification problems may, as well, occurs when the decision involves several decision makers with conflicting preferences. For example, consider a family that wants to spend a weekend in a hotel and each family member classifies these hotels to the ordered categories in her own way. Converging the family members' classifications to a group classification is not an easy task. Several studies have been proposed to deal with the problem of group decision making. In fact, *social choice theory* aims at conceiving efficient and fair methods for group decision making. Such methods aggregate the preferences of different agents of a society into a final decision over a set of objects. Besides the quality of these methods, communication costs must be considered for practical reasons. Thus, *computational social choice* occurs as an interdisciplinary field with concrete applications such as computer science. Methods designed by computational social choice can be applied by the *Group Recommender System (GRS)*. Such systems aim, as well, at dealing with the problem of group decision making. Hence they seek recommendations of items that are good for a group of decision makers. The recommended objects have to satisfy the individual preferences of all group members. In general, these meth-

ods are based on two approaches to generate recommended objects: a centralized approach based on aggregation procedures (generally conceived by computational social choice theory) and a distributed approach based on negotiation techniques. The first approach proposes recommended items without taking into account the group behavior and the possibility that each group member may change her preferences over time. The second approach allows interactions between several decision makers. Each decision maker has the ability to accept or reject a recommended object, and an agreement has to be reached between all of them to select an item. However, to the best of our knowledge, approaches that we presented previously for group recommender system are not used to deal with group ordinal classification problems. Consequently, a second issue arises: *how to make a group of decision makers converge to a group classification?*

To do so, the aim of this thesis is to propose a group decision making process for ordinal classification problem. The representation of this process will be shown through two main phases, a pre-negotiation and a negotiation phase which are conducted as follows:

- The pre-negotiation phase is an individual preparation phase in which we focus on a single decision maker and propose an interactive elicitation protocol for ordinal classification problem. In fact, we consider an outranking-based model, *MR-Sort*, a popular model for multiple criteria decision analysis, based on pairwise comparisons between alternatives and idealized profiles representing the "limit" of each category. At each step, the decision maker is asked to classify an object to a given category and then her answer is integrated as trained set for the learning model. The protocol aims at assisting the decision maker to better understand her preferences while reducing the cognitive burden on her, by asking questions carefully chosen to be as pertinent as possible. The choice of questions is made based on efficient strategies;
- The negotiation phase is the step in which we generate the group ordinal classification. In fact, after helping each decision maker better understand her preferences and generate classifications of alternatives into ordered categories

independently of other decision makers, we propose a group classification process based on negotiation techniques. This process aims at deriving a group classification through the individual classifications made by each group member. Therefore, we propose a multi-agent system to deal with the group classification problem, where each agent acts on behalf of each group member involved in the classification process. The agents communicate with each other following a negotiation protocol. We consider that agents are cooperative; they all have a common goal, which is reaching a group classification that satisfies as much as possible all group members. The process is conducted by a mediator agent in order to facilitate and accelerate the negotiation process and minimize the disclosed information between agents.

The manuscript is organized, in four chapters, as follow:

- 1) Chapter 1. Decision aiding and preference elicitation: In this chapter, we provide a background of Multiple-Criteria Decision Analysis (MCDA) and preference elicitation in the context of ordinal classification problem.
- 2) Chapter 2. Group decision making and automated negotiation: In this chapter, we present the literature review associated to group decision making and automated negotiation.
- 3) Chapter 3. Pre-negotiation process: Interactive Elicitation of an ordinal classification model: This chapter implements the pre-negotiation phase where we will assist a single decision maker to identify her classifications. Hence, we present a new "incremental" elicitation process. The proposed process aims at eliciting a decision maker preferences in the context of ordinal classification problem by asking her informative questions. Several experimentation are performed in order to evaluate the effectiveness of our process.
- 4) Chapter 4. Group-based Ordinal Classification based on a Negotiation Process: This chapter is dedicated to present our solution adopting a multi-agent decision-making system to derive group ordinal classification. We detail the

process that aims at converging a group members to a group classification based on their individual classifications (derived after the pre-negotiation phase). It makes use of negotiation techniques in order to derive a group classification. The proposed process aims at satisfying fairly all the group members while taking into account dynamic external factors such as the time progression and the number of concessions made during the process which may influence the agent's behavior and more particularly her acceptance flexibility. Moreover, it aims at optimizing the privacy of their individual classifications. We then experiment different intuitions that drove the development of the process.

- 5) Conclusion and perspectives: We conclude this thesis with a summary of the main contributions of the research and some future directions.

Résumé étendu

Prendre des décisions est un acte naturel dans la vie quotidienne. Nous décidons ce que nous mangeons à midi, où nous passons nos vacances, quelle carrière professionnelle nous pouvons envisager, etc. Dans de nombreux contextes, cette tâche représente un enjeu important. Par exemple, les dirigeants d'une entreprise prennent des décisions stratégiques pour lancer de nouveaux investissements. Ces décisions peuvent avoir un impact sur l'avenir de l'entreprise et engendrent un risque important. L'aide à la décision vise à formaliser les problèmes de décision et à structurer les préférences d'un décideur pour l'aider à prendre la décision la plus appropriée. Dans cette thèse, parmi les problèmes d'aide à la décision, nous considérons le problème de la *classification ordinale*, où l'objectif est de classer des objets dans des catégories ordonnées. Par exemple, la classification des hôtels dans les catégories ordonnées suivantes : 1 étoile, 2 étoiles, 3 étoiles, etc.

Les problèmes de classification ordinale peuvent être considérés dans le contexte de la décision à critères multiples, où les objets sont jugés selon plusieurs points de vue appelés *critères*. Ces critères ne sont pas réductibles à un seul critère et peuvent être contradictoires. Par conséquent, le problème de décision devient plus complexe. Par exemple, pour classer un hôtel dans les catégories ordonnées, on peut considérer les points de vue suivants : confort, prix, qualité de la restauration, etc. Le processus d'aide à la décision à critères multiples propose plusieurs modèles pour traiter les problèmes de classification ordinale. Les paramètres de ces modèles, tels que les poids des critères qui reflètent l'importance des critères pour le décideur, prennent en compte les préférences du décideur afin de lui fournir

un résultat correspondant à ses préférences. Dans l'exemple de la classification des hôtels, un décideur peut accorder plus d'importance au prix qu'au confort. Les modèles d'élicitation des préférences ont été conçus pour mieux particulariser un modèle de décision pour un décideur donné. Ils garantissent que les paramètres de ce modèle correspondent aux préférences du décideur et reflètent sa façon de décider. Dans le contexte de la classification ordinaire, le décideur peut être interrogé directement sur les valeurs des paramètres du modèle. Cependant, cette approche nécessite une compréhension approfondie du modèle de décision par le décideur, ce qui peut être au-delà de ses capacités cognitives. Au lieu d'interroger directement le décideur sur les valeurs des paramètres, nous pouvons récupérer des exemples représentant ses jugements préférentiels et ainsi l'interroger sur ses préférences subjectives pour classer les objets aux catégories. Ensuite, les paramètres du modèle seront déduits sur la base de ses réponses. Cependant, poser plusieurs questions au décideur n'est pas une tâche facile, surtout dans de nombreux contextes pratiques, y compris les applications sur le Web (comme les plateformes d'apprentissage en ligne ou de commerce électronique). La question est alors la suivante : *comment pouvons-nous obtenir les préférences d'un décideur dans un contexte de classification ordinaire à critères multiples tout en lui posant un nombre limité de questions ?*

La complexité des problèmes de classification ordinaire peut également survenir lorsque la décision implique plusieurs décideurs aux préférences conflictuelles. Par exemple, considérons une famille qui souhaite passer un week-end dans un hôtel et chaque membre de la famille classe ces hôtels dans les catégories ordonnées à sa manière. Faire converger les classifications des membres de la famille vers une classification de groupe n'est pas une tâche facile. Plusieurs travaux ont été proposés pour traiter le problème de la prise de décision en groupe.

La *théorie du choix social* vise à concevoir des méthodes efficaces et équitables pour la prise de décision en groupe. Ces méthodes permettent d'agréger les préférences des différents agents d'une société en une décision finale sur un ensemble d'objets. Outre la qualité de ces méthodes, les coûts de communication doivent

être pris en compte pour des raisons pratiques. Ainsi, *choix social computationnel* apparaît comme un domaine interdisciplinaire avec des applications concrètes telles que l'informatique. Les méthodes conçues par le choix social computationnel peuvent être appliquées par le *système de recommandation de groupe (GRS)*. Ces systèmes visent également à traiter le problème de la prise de décision de groupe. Ils cherchent donc à recommander des objets qui sont bons pour un groupe de décideurs. Les objets recommandés doivent satisfaire les préférences individuelles de tous les membres du groupe. En général, ces méthodes sont basées sur deux approches pour générer des objets recommandés : une approche centralisée basée sur des procédures d'agrégation (généralement conçue par la théorie du choix social computationnel) et une approche distribuée basée sur des techniques de négociation. La première approche propose des objets recommandés sans tenir compte du comportement du groupe et de la possibilité que chaque membre du groupe puisse changer ses préférences au fil du temps. La seconde approche permet des interactions entre plusieurs décideurs. Chaque décideur a la possibilité d'accepter ou de rejeter un objet recommandé, et un accord doit être trouvé entre tous les décideurs pour sélectionner un objet. Cependant, à notre connaissance, les approches que nous avons présentées précédemment pour les systèmes de recommandation de groupe ne sont pas utilisées pour traiter les problèmes de classification ordinale de groupe. Par conséquent, une deuxième question se pose : *comment faire converger un groupe de décideurs vers une classification de groupe ?*

Pour ce faire, l'objectif de cette thèse est de proposer un processus de décision de groupe pour un problème de classification ordinale. La représentation de ce processus se fait à travers deux phases principales, une phase de pré-négociation et une phase de négociation qui se déroulent comme suit :

La phase de pré-négociation

Pour cette phase, nous considérons le problème de l'élicitation d'un modèle de classification ordinale pour un seul décideur. En particulier, nous considérons le tri par

règle de majorité (MR-Sort). La caractéristique remarquable de la procédure MR-Sort est que ses classifications sont facilement explicables aux utilisateurs. Selon MR-Sort, une alternative est affectée à une catégorie lorsqu'elle est "au moins aussi bonne" qu'un profil décrivant la "limite" inférieure de la catégorie et "pas meilleure" que la limite supérieure de la catégorie. Les profils limites indiquent les caractéristiques qu'une alternative devrait avoir pour être affectée à une classe particulière selon le décideur (par exemple : un hôtel 4 étoiles devrait avoir des chambres d'au moins une certaine taille, il devrait avoir une piscine, un hôtel 5 étoiles devrait avoir un restaurant chic, etc). La relation entre les alternatives et les profils limites est basée sur une règle de majorité pondérée (dans Sobrie et al. [131] un modèle non linéaire pour combiner les poids est considéré). Afin d'utiliser MR-Sort dans la pratique, il est nécessaire de procéder à une phase d'élicitation des préférences afin d'évaluer les paramètres du modèle. L'approche "classique" de l'élicitation consiste en une évaluation fine des paramètres du modèle. Cependant, il n'est souvent pas raisonnable de demander directement au décideur les valeurs des paramètres du modèle. La nouveauté de ce travail est d'adopter une approche incrémentale d'élicitation afin d'apprendre les paramètres de MR-Sort. Pour ce faire, nous supposons que le décideur peut facilement classer une alternative donnée dans l'une des catégories ; les informations fournies par l'utilisateur peuvent être utilisées pour mettre à jour le modèle et fournir un meilleur résultat. L'intuition est que, en choisissant sélectivement les éléments que le décideur peut classer, nous pourrions obtenir une bonne approximation du modèle "correct" du décideur (DM) même avec un petit nombre d'exemples. Nous soulignons également que l'information fournie par l'utilisateur peut être bruyante, car il peut commettre une erreur dans l'évaluation de la bonne catégorie pour l'alternative.

Pour ce faire, nous adoptons l'optimisation à marge maximale, dont le principe principal est le suivant. Les affectations connues des alternatives aux catégories sont exprimées par un ensemble d'inégalités sur les paramètres réalisables ; une marge non négative partagée est introduite comme une variable de décision qui est maximisée dans la fonction objectif. Les réponses bruyantes sont traitées en relaxant les contraintes à l'aide de variables d'écart et en ajoutant un terme de

pénalité dans la fonction objectif pour les contraintes violées.

Nous proposons ainsi une approche incrémentale pour l'élicitation des paramètres d'un modèle MR-Sort ; à chaque étape de la procédure d'élicitation, le système pose une question à l'utilisateur et la question est spécifiquement choisie afin d'être la plus "informative" possible. Notre algorithme principal utilise de manière répétée la procédure d'optimisation de la marge maximale avec deux objectifs :

- 1) faire une estimation des paramètres du modèle compte tenu des informations actuelles ;
- 2) déterminer une mesure d'incertitude associée à une question potentielle.

Pour le second objectif, nous considérons différentes stratégies basées sur le calcul de la marge maximale a posteriori (en considérant les différentes réponses que le décideur pourrait donner).

Nous évaluons la procédure d'élicitation interactive proposée en comparant les performances des différentes stratégies, et nous montrons que la procédure converge rapidement vers la classification optimale réelle en très peu de cycles d'interaction.

• Apprentissage d'un modèle MR-Sort à partir des affectations

Nous supposons que certaines affectations d'alternatives aux catégories sont données. Dans ce travail, nous supposons que les profils limites sont donnés. Dans ce qui suit, nous introduisons d'abord la notion de catégories possibles et nécessaires, puis nous présentons une optimisation à marge maximale pour apprendre les paramètres d'un modèle MR-Sort.

- 1) Catégories possibles: chaque alternative est évaluée selon un ensemble de n critères $\mathcal{C} = \{cr_1, \dots, cr_n\}$, l'évaluation de x est un vecteur $(x_1, x_2, \dots, x_n)$ et l'ensemble des catégories est noté par $\mathcal{C} = \{C_1, C_2, \dots, C_p\}$. Soit $\mathcal{A}$ l'ensemble des alternatives dont la catégorie est connue et LS l'"ensemble d'apprentissage" (paires d'alternatives et de catégories) : $LS = \{(x, C_h), x \in \mathcal{A}, x \rightarrow C_h\}$.

À partir de LS, en supposant avoir un modèle MR-Sort sous-jacent, nous pouvons raisonner sur les poids qui sont cohérents avec les connaissances actuelles. Selon l'équation (1.1) et (1.2), une affectation du type $x \xrightarrow{\text{DM}} C_h$ (effectuée par le décideur) correspond à l'imposition des contraintes linéaires :

$$x \succeq b^h \iff \sum_{i: x_i \geq b_i^h} w_i \geq \lambda \quad (1)$$

$$x \not\succeq b^{h+1} \iff \sum_{i: x_i \geq b_i^{h+1}} w_i < \lambda \quad (2)$$

Soit $\Theta(\text{LS})$ un ensemble de paramètres compatibles avec l'ensemble d'apprentissage LS, satisfaisant les contraintes (3.1) et (3.2) pour toutes les paires dans LS, et les exigences que les poids soient non négatifs et normalisés. Lorsque l'on traite des modèles de préférence partiellement spécifiés, il est courant de raisonner sur les informations de préférence possibles et nécessaires [66]. Dans notre contexte, étant donné une alternative et son évaluation de performance, nous pouvons raisonner sur les paramètres qui sont cohérents avec les connaissances actuelles. Maintenant, étant donné un élément $x \notin \mathcal{A}$, nous définissons l'ensemble des catégories possibles $\text{PC}(x; \text{LS})$ comme l'ensemble des catégories telles qu'il existe une instantiation des paramètres $\theta = \{w_1, \dots, w_n, \lambda\}$ cohérente avec l'affectation *étant donné les affectations connues précédemment* LS.

$$\text{PC}(x; \text{LS}) = \{C_i \in \mathcal{C} \mid \exists \theta \in \Theta(\text{LS}) : x \xrightarrow{\theta} C_i\}$$

où nous écrivons $x \xrightarrow{\theta} C_i$ pour souligner la dépendance entre les paramètres θ et l'affectation. S'il n'y a qu'une seule catégorie possible, c'est-à-dire $|\text{PC}(x)| = 1$, cela signifie que l'alternative doit être *nécessairement* assignée à la seule catégorie dans PC (en supposant que le modèle est cohérent avec l'ensemble d'apprentissage). En pratique, il est possible qu'une connaissance partielle des paramètres nous permette de déterminer dans quelles catégories nous devons placer plusieurs alternatives.

Nous prévoyons que le concept de catégories possibles joue un rôle important dans nos stratégies d'élicitation

- 2) Optimisation de la marge maximale: nous abordons maintenant le problème de l'apprentissage des paramètres $\theta = \{w_1, \dots, w_n, \lambda\}$ d'un modèle MR-Sort (étant donné un ensemble d'affectations d'apprentissage) avec un programme linéaire. Nous supposons qu'on nous donne en entrée les valeurs suivantes : le nombre de critères n , le nombre de catégories p , et les profils limites b^h pour chaque $h \in [p]$. Notre objectif est d'évaluer les poids w et le seuil de majorité λ , qui sont des variables de décision pour le problème d'optimisation. Afin de discriminer entre différents choix de paramètres, la première étape consiste à introduire une marge partagée μ comme variable de décision supplémentaire. Le premier programme linéaire (LP) suppose que les données en LS sont parfaitement cohérentes avec le modèle MR-Sort.

$$\mu^* = \max \quad \mu \quad (3)$$

$$\text{s.t.} \quad \sum_{i: x_i \geq b_i^h} w_i \geq \lambda + \mu \quad \forall (x, C_h) \in \text{LS} \quad (4)$$

$$\sum_{i: x_i \geq b_i^{h+1}} w_i \leq \lambda - \mu \quad \forall (x, C_h) \in \text{LS} \quad (5)$$

$$\sum_{i=1}^n w_i = 1 \quad (6)$$

$$\mu \in \mathbb{R} \quad (7)$$

$$w_i \geq 0 \quad \forall i \in [n] \quad (8)$$

Les contraintes (3.4) et (3.5) correspondent aux équations (3.1) et (3.2) avec l'inclusion de la marge partagée μ . Nous observons qu'en général, il peut y avoir des incohérences occasionnelles dans les réponses des décideurs, ou simplement les données peuvent ne pas être compatibles avec un modèle MR-Sort. La façon "typique" de traiter ce problème est d'introduire des variables d'écart (dont on cherche à minimiser la somme); cette idée remonte à l'approche UTA [77] et à plusieurs mod-

èles ultérieurs [2, 55]. Nous formalisons le problème de la recherche des poids d'un modèle MR-Sort (étant donné un ensemble d'affectations d'apprentissage) par le programme linéaire suivant.

$$m^* = \max \quad \mu - \alpha \sum_{(x, C_h) \in \text{LS}} \xi_{i,j} \quad (9)$$

$$\text{s.t.} \quad \sum_{i: x_i \geq b_i^h} w_i + \xi_{x,h} \geq \lambda + \mu \quad \forall (x, C_h) \in \text{LS} \quad (10)$$

$$\sum_{i: x_i \geq b_i^{h+1}} w_i - \xi_{x,h} \leq \lambda - \mu \quad \forall (x, C_h) \in \text{LS} \quad (11)$$

$$\sum_{i=1}^n w_i = 1 \quad (12)$$

$$\mu \in \mathbb{R} \quad (13)$$

$$\xi_{x,h} \geq 0 \quad \forall (x, C_h) \in \text{LS} \quad (14)$$

$$w_i \geq 0 \quad \forall i \in [n] \quad (15)$$

Les variables sont les suivantes : $\lambda \in [0, 1]$, $w_i \in [0, 1] \forall i \in [n]$, $\mu \in [0, 1]$ and $\xi_{x,h} \in [0, 1] \forall x \in \text{LS}$ such that x assigned to C_h . Nous utilisons un paramètre α pour exprimer le “coût” de la violation d’une contrainte représentant les affectations de l’ensemble d’apprentissage. L’objectif (équation 3.9) est de maximiser la marge partagée μ (que nous voulons maximiser) moins la somme des violations des contraintes $\xi_{i,j}$ (que nous voulons minimiser). Afin de pouvoir gérer l’incohérence des réponses de l’utilisateur, nous ajoutons une variable muette ($\xi_{x,h}$) pour chaque $x \in \mathcal{A}$ et pour chaque $h \in \{1, \dots, p\}$. La contrainte 3.10 et la contrainte 3.11 gèrent les conditions liées aux affectations des alternatives dans l’ensemble d’apprentissage LS. La contrainte 3.12 impose que les poids soient normalisés.

- **Processus d’éllicitation interactif**

Nous proposons une méthode d’éllicitation pour un problème de classification ordinaire en supposant que les préférences peuvent être modélisées par

Algorithme 1 : L'algorithme MR-sort-Inc. K est le nombre de questions, X l'ensemble des alternatives, C l'ensemble des catégories, LS ($LS = \{(x, C_h), x \in \mathcal{A}, x \rightarrow C_h\}$) l'ensemble d'apprentissage, s la stratégie.

```

1 Procedure MR-Sort-Inc( $K, X, C, LS, s$ ):
2   for  $j \leftarrow 1, \dots, K$  do
3     Calculer  $m^*(LS)$  and et  $w^*$  associé et  $\lambda^*$  avec un programme linéaire
4     Classer les alternatives dans  $X \setminus \mathcal{A}$  en utilisant  $w^*$  et  $\lambda^*$ 
5     Calculer l'erreur de classification /* Ceci n'est possible qu'en
      simulations */
6     for  $x \in X \setminus \mathcal{A}$  do
7       for  $C_i \in C$  do
8          $LS' \leftarrow LS \cup \{x \rightarrow C_i\}$  /* Inclure une affectation
          supplémentaire */
9          $v_i(x) \leftarrow m^*(LS')$  /* Calculer la marge à posteriori */
10      end
11       $S_s(x) \leftarrow$  agrégation  $v(x)$  selon la stratégie  $s$ 
12    end
13     $x^* \leftarrow \arg \max_x (S_s(x))$  /* Décider la question suivante */
14     $C_{x^*} \leftarrow \text{Answer}(x^* \rightarrow ?)$  /* Poser une question au DM */
15     $LS \leftarrow LS \cup \{(x^*, C_{x^*})\}$ 
16  end
17  return  $w^*, \lambda^*$ 

```

MR-sort. Notre procédure d'élicitation commence avec un petit ensemble d'apprentissage LS auquel nous ajoutons un par un de nouveaux exemples d'affectation. Nous posons des questions du type “*Dans quelle catégorie faut-il classer une alternative x ?*” ; il est crucial de sélectionner des éléments “informatifs” à interroger, afin de converger rapidement (en quelques cycles d'interaction) vers un bon modèle de classification.

Un modèle MR-sort est défini par les paramètres suivants : les poids $(w_1, \dots, w_n)$, les profils limites $(b^1, \dots, b^{p+1})$ et le seuil de majorité λ . Nous fixons les profils limites avant le début de l'élicitation. Dans les expériences ci-dessous, les profils limites sont choisis de manière à répartir uniformément l'échelle des critères. Les profils limites sont choisis pour maximiser la différence $b_i^{h+1} - b_i^h, \forall i \in [n], \forall h \in [p]$. Par conséquent, les poids sont les seuls paramètres à apprendre. Les principales étapes de notre approche sont présentées dans l'Algorithme 2. En bref, nous commençons avec certaines données de l'ensemble d'apprentissage LS (alternatives d'apprentissage $\mathcal{A}$ et leur classification) et

nous les incluons comme contraintes dans notre programme linéaire (LP). Étant donné l'ensemble d'apprentissage, nous pouvons estimer les poids w^* et λ^* à l'aide du programme linéaire; dans les simulations, nous pouvons également évaluer l'erreur de classification. Ensuite, nous testons l'affectation de chaque élément non affecté à chaque catégorie et récupérons les valeurs de marge obtenues en utilisant la routine d'optimisation avec chacune des affectations supplémentaires. Cela nous donne un vecteur de marges "a posteriori", qui sont agrégées différemment selon la "stratégie" donnant un score pour chaque élément. L'élément ayant le score le plus élevé est choisi : l'utilisateur est interrogé sur l'affectation de cet élément et l'ensemble d'apprentissage est augmenté. Cette procédure se poursuit jusqu'à une condition d'arrêt (dans nos expériences, lorsque nous atteignons un nombre fixe de questions, mais dans les applications réelles, la fin peut être laissée à l'utilisateur). Le dernier programme linéaire contient toutes les affectations de l'ensemble d'apprentissage calculé, il fournit donc les poids que nous recherchons.

- **Stratégies de sélection des questions:** dans ce qui suit, nous présentons plusieurs stratégies permettant de sélectionner la prochaine question à poser à l'utilisateur. Ces stratégies sont utilisées dans le cadre de notre paradigme d'élicitation interactive décrit dans l'algorithme 2. Les stratégies proposées utilisent les programmes d'optimisation de la marge maximale pour identifier les questions informatives.

- 1) **Most uncertain:** avec cette stratégie, nous cherchons à demander au DM de classer l'item *most uncertain* dans $X \setminus \mathcal{A}$, c'est-à-dire l'item qui est compatible (selon les contraintes dérivées des affectations connues de l'ensemble d'apprentissage) avec le plus grand nombre de catégories. Le score que cette stratégie attribue à une alternative x est donné par la cardinalité de l'ensemble des catégories possibles : $S_{MU}(x) = |\text{PC}(x; \text{LS})|$. Le calcul de l'ensemble $\text{PC}(x)$ pour un élément x fait appel au programme linéaire pour l'apprentissage des paramètres d'un MR-Sort sans les variables d'écart (Equations 3.3-3.8). Pour chaque catégorie C_i , nous contraignons alors l'élément x à être assigné à C_i et vérifions simplement

si la marge est non négative ; en effet $\mu^*(\text{LS} \cup (x, C_i)) > 0$ si et seulement si C_i est une catégorie possible ($C_i \in \text{PC}(x; \text{LS})$). Le score de cette stratégie peut être écrit de manière compacte comme :

$$S_{\text{MU}}(x) = |\text{PC}(x; \text{LS})| = \sum_{i=1}^p H[\mu^*(\text{LS} \cup (x, C_i))]$$

où $H(\cdot)$ est la fonction de pas.

- 2) Sum-of-margin : le problème de la stratégie *most uncertain* est que, en gros, elle est agnostique quant à savoir si une affectation potentielle est cohérente avec une grande partie de l'espace des paramètres ou seulement avec une petite partie. L'intuition de *sum-of-margin* est d'utiliser la valeur de la fonction objective de la PL comme une mesure de substitution du "degré" de satisfaction d'une affectation $x \rightarrow C_h$. Intuitivement, nous devrions poser des questions sur des éléments qui pourraient bien entrer dans plusieurs catégories différentes. Nous incluons également la possibilité que certaines contraintes puissent être violées, et utilisons donc la deuxième LP (équations 3.9-3.15), avec les variables de pénalité ξ pour les affectations violées de l'ensemble d'apprentissage.

En considérant une alternative $x \in X \setminus \mathcal{A}$, nous évaluons la marge pénalisée m^* en ajoutant l'affectation $x \rightarrow C_i$ pour tous les $i \in [p]$ et construisons le vecteur suivant :

$$v(x) = (m^*(\text{LS} \cup (x, C_1)), \dots, m^*(\text{LS} \cup (x, C_p))).$$

Afin d'agréger ce vecteur en une seule mesure numérique, nous adoptons la somme. Par conséquent, le score de l'alternative x est calculé comme suit

$$S_{\Sigma}(x) = \sum_{i=1}^p v_i(x) = \sum_{i=1}^p m^*(\text{LS} \cup (x, C_i)).$$

- 3) Entropy: cette stratégie adopte la notion d'entropie pour évaluer l'incertitude pour une alternative donnée. Comme la stratégie *sum of margin*, nous calculons m^* qui représente la marge pénalisée maximale si nous as-

signons l'alternative x à la catégorie C_i , pour tous les $i \in [p]$. Nous combinons ensuite ces valeurs dans un vecteur d'évaluation v' qui filtre les valeurs négatives (en leur attribuant une valeur nulle) et nous normalisons pour que la somme des valeurs soit égale à un :

$$v'_i(x) = \frac{R(m^*(LS \cup (x, C_i)))}{\sum_{j=1}^p R(m^*(LS \cup (x, C_j)))}$$

où $R(\cdot)$ est la fonction de rampe. Nous appliquons ensuite l'entropie pour calculer le score final de l'alternative.

$$S_E(x) = - \sum_{i \in [p]: v'_i(x) > 0} v'_i(x) \log(v'_i(x))$$

Intuitivement, cette méthode devrait favoriser l'interrogation sur les éléments dont l'incertitude est la plus grande au sens de la théorie de l'information, en considérant le vecteur v' (composé de valeurs de marge non négatives normalisées) comme une mesure de substitution de la distribution de probabilité de la catégorie de x (c'est-à-dire la probabilité que x soit dans chacune des catégories).

- 4) Random: comme base de référence, nous considérons la sélection aléatoire d'un objet x à partir de $X \setminus \mathcal{A}$. L'utilisation de cette stratégie revient à utiliser une procédure d'élicitation non incrémentale avec $|LS| + K$ (LS étant l'ensemble d'apprentissage initial de l'algorithme 2 et K étant le nombre de questions) alternatives dans l'ensemble d'apprentissage.

La phase de négociation

Une fois que nous avons assisté un décideur unique, dans la phase de pré-négotiation, à comprendre ses préférences et à classer les alternatives dans des catégories ordonnées, la phase de négociation commence. L'objectif principal de cette phase est de proposer un processus de classification ordinaire basé sur la négociation basée sur un système multi-agents. Le processus dérive une classification de groupe qui tient

compte des préférences individuelles et qui est considéré comme "équitable" par le groupe, tout en limitant l'échange d'informations privées (afin de garantir un certain degré de confidentialité sur les préférences individuelles). Il est important de comprendre que l'équité n'est pas imposée de l'extérieur mais qu'elle est obtenue par le processus de négociation lui-même.

Le processus de choix d'une classification ordinale d'un ensemble d'alternatives implique une série de négociations, une pour chaque alternative. Nous modélisons explicitement le fait que les attitudes du décideur par rapport à la négociation peuvent changer dans le temps pour de nombreuses raisons. Par exemple, le comportement d'un décideur peut être influencé par le nombre de concessions qui ont été faites dans le passé et le temps restant : imaginez que pour un décideur donné, toutes les classifications communes passées étaient différentes de ses propres classifications préférées : cela pourrait conduire un décideur à être moins flexible (ne pas accepter de déviations par rapport à son choix préféré) à l'avenir. En effet, lorsqu'un décideur fait de multiples concessions en acceptant des classifications différentes de sa propre classification préférée, il peut avoir tendance à devenir moins coopératif, et donc à faire moins de concessions à l'avenir. Le temps représente également un facteur important influençant le comportement des décideurs : à mesure que l'échéance (où une décision doit être prise) se rapproche, les décideurs peuvent avoir tendance à devenir plus coopératifs, afin d'atteindre une décision, et ainsi, accepter des choix qu'ils n'auraient pas acceptés au début du processus.

Notre processus de prise de décision collective est basé sur un système multi-agent où chaque décideur impliqué dans le processus de classification est représenté par un agent utilisateur agissant en son nom. Chaque agent utilisateur se comporte de manière dynamique tout au long du processus ; les agents utilisateurs ont une architecture similaire pour représenter les attitudes de négociation. Les décisions (acceptation ou rejet d'une proposition donnée) sont basées sur le résultat de l'agrégation de plusieurs critères représentant leur comportement, comme la flexibilité dans le temps. Le protocole de communication est caractérisé par un certain degré de confidentialité, car les agents décident de révéler (partiellement)

leurs préférences.

Dans ce travail, nous supposons que tous les agents sont, au moins dans une certaine mesure, coopératifs et partagent l'objectif commun de parvenir à un accord. Cependant, les attitudes individuelles des agents peuvent être différentes ou même contradictoires et par conséquent, dans certains cas, il peut être difficile de parvenir à une classification commune. Afin de pallier ces situations conflictuelles, un agent médiateur est chargé d'initier et de conduire le processus de négociation. Un rôle clé de l'agent médiateur est d'assurer la fin du processus en prenant une décision objective lorsque la négociation ne mène pas à un accord, assurant ainsi le succès du processus.

Nous présentons des résultats expérimentaux montrant que notre approche est efficace pour produire des recommandations qui réalisent un bon compromis entre la satisfaction individuelle (mesurée par la satisfaction moyenne ou l'utilité) et l'équité (mesurée en considérant l'agent le moins satisfait et l'indice de Gini), tout en assurant le maintien d'un certain degré de confidentialité. Les résultats de la simulation sont examinés en détail avec une analyse de sensibilité, montrant que notre approche se compare favorablement par rapport aux méthodes centralisées basées sur des règles d'agrégation communes.

- **Architecture de classification multi-agents**

Pour dériver une classification de groupe, nous avons utilisé la théorie multi-agents pour faire face aux problèmes de communication et de collaboration qui sont cruciaux dans tout système de prise de décision. Les agents sont capables d'apprendre de leur expérience passée et par conséquent d'améliorer leur processus de prise de décision en ajustant leurs stratégies. Ainsi, le comportement des agents peut changer en fonction de l'environnement dans lequel ils sont déployés.

En outre, il existe des propriétés communes aux agents, notamment l'autonomie, la proactivité, la réactivité et l'adaptabilité. Par conséquent, les agents sont également capables de connaître et d'apprendre les préférences des utilisateurs

humains lorsqu'ils interagissent avec eux [80].

Les agents ont généralement un contrôle partiel sur leur environnement. Chaque agent a un point de vue limité et des informations incomplètes. La figure 4.1 représente l'architecture distribuée de notre système multi-agent pour dériver la classification de groupe. Elle montre d'une part la phase de pré-négociation et d'autre part la phase de négociation. Une fois que chaque agent utilis-

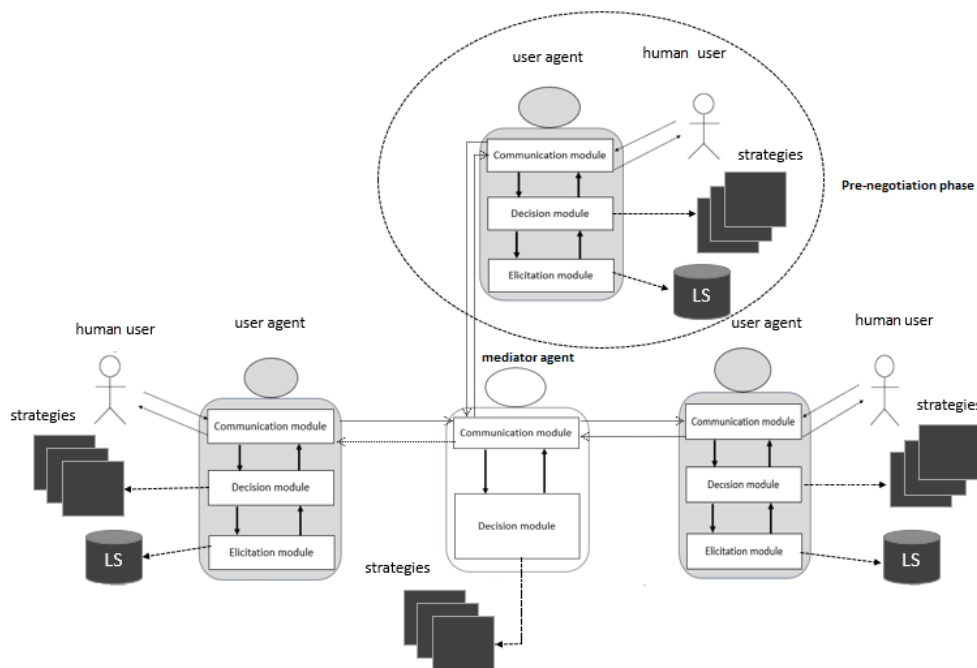


Figure 1 – Architecture de classification multi-agents

teur a généré la classification ordinale de toutes les alternatives, la phase de négociation commence. Cette phase est guidée par l'agent médiateur qui possède également un module de communication afin de communiquer avec les agents utilisateurs selon un protocole de communication. L'agent médiateur dispose d'un module de décision qui lui permet de prendre une décision dans le cas où les agents utilisateurs ne parviennent pas à se mettre d'accord sur la classification d'une alternative donnée.

- **Processus de classification** ce processus de classification est une négociation initiée par l'agent médiateur. A chaque tour t , l'agent médiateur demande à tous les agents utilisateurs comment ils classifieraient une alternative donnée. Chaque agent utilisateur a son propre comportement basé sur les préférences ; un agent peut se contenter de communiquer sa catégorie préférée tandis qu'un autre agent utilisateur communique un ensemble de "catégories possibles" (qui dépend du fait qu'il soit flexible ou égoïste, de sa sensibilité au temps qui passe, etc.)

Une fois que tous les agents utilisateurs ont envoyé leurs catégories possibles, l'agent médiateur prend l'une des 2 décisions suivantes :

- elle classe l'alternative : lorsqu'une majorité qualifiée d'agents utilisateurs est d'accord avec la classification (pour départager les ex-aequo, le médiateur utilise la règle de la pluralité, c'est-à-dire qu'il choisit celle qui est soutenue par le plus grand nombre d'agents);
- elle reporte la classification : lorsqu'il n'y a pas de majorité suffisante pour soutenir la classification de l'alternative dans une catégorie quelconque.

Après avoir proposé aux agents utilisateurs toutes les alternatives de classification et tant que la date limite du processus n'est pas dépassée, l'agent médiateur applique l'ensemble du processus aux alternatives pour lesquelles la classification a été reportée. Certains agents utilisateurs peuvent modifier leur ensemble de catégories possibles au cours du processus (par exemple, ils peuvent être plus flexibles lorsque la date limite est proche). Lorsque le temps de négociation est terminé, l'agent médiateur utilise la stratégie de pluralité pour classer les alternatives restantes. La figure 4.2 donne un aperçu de notre processus de classification.

Le manuscrit est organisé, en quatre chapitres, comme suit :

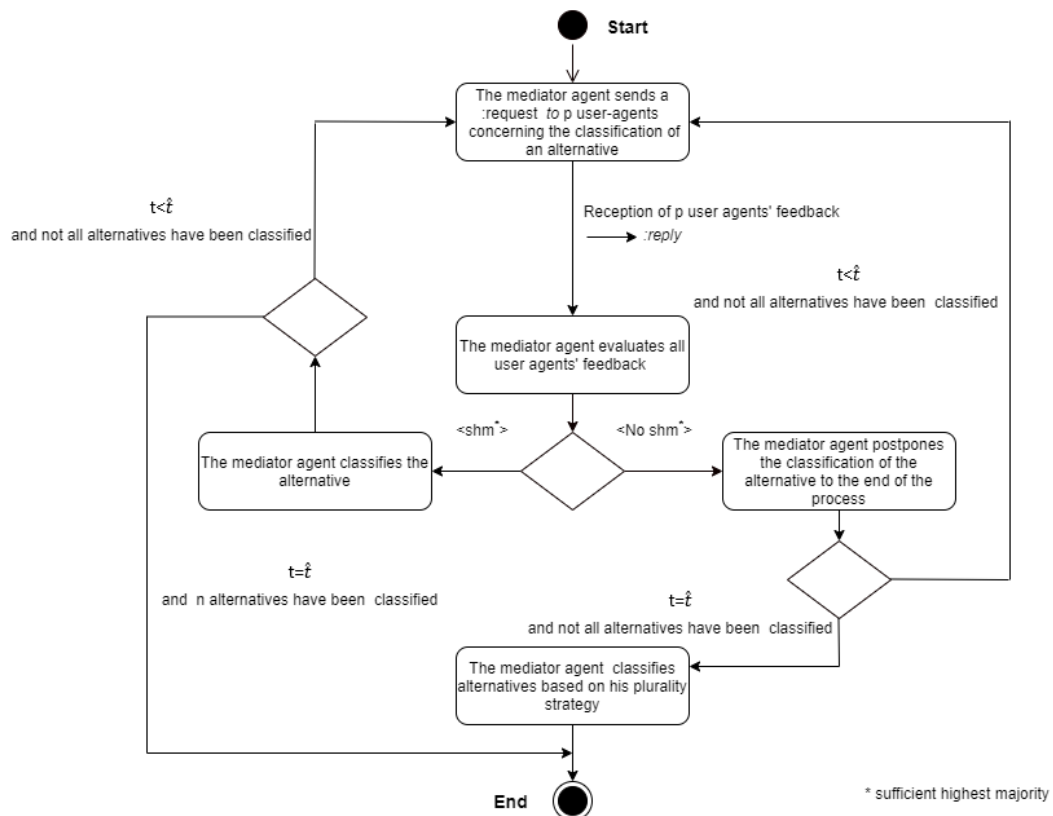


Figure 2 – Group-classification process

1) Chapitre 1. Aide à la décision et élicitation des préférences: Dans cette thèse, nous développons un processus de classification ordinaire de groupe. La première contribution est une méthodologie permettant d'apprendre de manière incrémentale les préférences des agents avec une charge cognitive minimale. Les préférences apprises de chaque agent seront ensuite utilisées pour l'assister dans la classification d'un ensemble d'alternatives. Ce chapitre est dédié à la présentation de la revue de la littérature associée aux principaux développements théoriques de cette problématique est organisé comme suit :

- La littérature sur l'aide à la décision, y compris la modélisation des préférences et le cas particulier de l'aide à la décision à critères multiples (MCDA).
- La littérature sur l'élicitation des préférences avec les cas particuliers de la classification ordinaire et les approches incrémentales.

2) Chapitre 2. La prise de décision en groupe et la négociation automatisée:

L'objectif principal de cette thèse consiste à dériver une classification de groupe à partir de classifications individuelles basées sur un processus de négociation. Dans ce chapitre, nous présentons la revue de la littérature relative à ce sujet de la manière suivante :

- la littérature sur la prise de décision en groupe dans le contexte de la classification ordinale.
- les concepts de base de la négociation automatisée.

- 3) Chapitre 3. Processus de pré-négociation: Elicitation interactive d'un modèle de classification ordinale : Ce chapitre met en œuvre la phase de pré-négociation où nous considérons le problème de l'élicitation interactive des préférences d'un seul décideur pour des problèmes de classification ordinale.

Notre protocole d'élicitation interactive demande, à chaque étape, au décideur de classer une alternative ; ces affectations sont utilisées comme un ensemble d'entraînement pour l'apprentissage du modèle. Comme nous souhaitons limiter la charge cognitive de l'élicitation, nous cherchons à poser des questions informatives afin de trouver une bonne approximation de la classification optimale en un nombre limité d'étapes d'élicitation. Notre méthode est tolérante aux erreurs. Nous proposons des stratégies efficaces pour calculer la question suivante et montrons comment son calcul peut être formulé comme un programme linéaire. Nous présentons des résultats expérimentaux montrant l'efficacité de notre approche.

- 4) Chapitre 4. Classification ordinale de groupe basée sur un processus de négociation : Ce chapitre est consacré à la présentation de notre solution adoptant un système décisionnel multi-agent pour dériver une classification ordinale de groupe. Nous détaillons le processus qui vise à faire converger les membres d'un groupe vers une classification de groupe basée sur leurs classifications individuelles (dérivées après la phase de pré-négociation). Il fait appel à des techniques de négociation afin de dériver une classification de groupe. Le processus proposé vise à satisfaire équitablement tous les membres du groupe tout en tenant compte de facteurs externes dynamiques tels que la progres-

sion dans le temps et le nombre de concessions faites au cours du processus qui peuvent influencer le comportement de l'agent et plus particulièrement sa flexibilité d'acceptation. De plus, il vise à optimiser la confidentialité de leurs classifications individuelles. Nous expérimentons ensuite les différentes intuitions qui ont présidé au développement du processus.

- 5) Conclusion et perspectives : Nous concluons cette thèse par un résumé des principales contributions de la recherche et de certaines orientations futures. Les recherches menées dans cette thèse peuvent être développées davantage. Discutons des adaptations spécifiques de chaque contribution pour des recherches futures.

Dans le chapitre 3, nous avons développé un processus incrémental d'élicitation des préférences pour le modèle MR-Sort, en supposant que les profils des catégories ont été définis a priori. Des recherches supplémentaires pourraient être menées pour assouplir cette hypothèse. En fait, l'apprentissage de tous les paramètres de MR-Sort sera difficile, car l'introduction de variables entières dans un problème d'optimisation nécessite un temps de calcul élevé. Nous souhaitons également réaliser des simulations comparant notre approche à d'autres approches d'élicitation, comme les travaux récents d'Olteanu [107], et les approches basées sur le regret minimax [8, 9], sur la base de données réelles. Nous avons l'intention d'étudier la connexion entre notre approche et le domaine de l'apprentissage automatique pour de nombreuses raisons telles que les similitudes entre la marge maximale et les machines à vecteurs de support ou la similitude entre l'élicitation incrémentale et l'apprentissage actif. De plus, nous envisageons d'étendre notre processus pour éliciter simultanément les préférences de plusieurs décideurs comme dans Teso et al. [136].

Dans le chapitre 4, nous avons développé un processus de négociation visant à dériver une classification de groupe. Les stratégies reflétant le comportement des agents sont menées sur la base d'une fonction d'utilité définie. Nous souhaitons tester notre approche avec des stratégies de décision plus sophistiquées, en incluant la possibilité pour les agents d'opposer leur veto à une classification proposée. De plus, nous supposons que les alternatives ont la

même importance. Des recherches supplémentaires pourraient être menées pour assouplir cette hypothèse. De même, nous pourrions modéliser les différents degrés d'expertise reflétant la confiance d'un décideur sur la classification des alternatives.

Cette thèse se situe au carrefour de plusieurs disciplines : la prise de décision multicritères, le choix social, les systèmes multi-agents et la négociation automatisée.

Chapter 1

Decision Aiding and Preference elicitation

Contents

1.1	Introduction	39
1.2	Decision Aiding	40
1.2.1	Decision Aiding Process	41
1.2.2	Preference Modeling	43
1.2.3	Multiple Criteria Decision Aiding	48
1.3	Preference elicitation	56
1.3.1	Preference Elicitation for ordinal classification methods	58
1.3.2	Incremental approach	61
1.4	Synthesis	63

1.1 Introduction

In this thesis, we develop a group ordinal classification process. The first contribution consists in a methodology allowing to learn incrementally the agents' preferences with a minimum cognitive burden. The learned preferences of each agent will

be then used to assist her in the classification of a set of alternatives. This chapter is dedicated to present the literature review associated to the main theoretical developments in this issue, and is organized as follows:

- The decision aiding literature, including the preference modeling and the particular case of multiple criteria decision aiding (MCDA) in Section 1.2.
- The preference elicitation literature with the particular cases of ordinal classification and the incremental approaches in Section 1.3.

We will then review the literature on group decision making, multi-agent system, and negotiation in Chapter 2.

1.2 Decision Aiding

In this section, we introduce the main definitions in the decision aiding literature, the steps of a decision aiding process, the main definitions of preference modeling and an introduction to the different MCDA methods.

Decision aiding is defined by Roy [117] as “the activity of the person who, through the use of explicit but not necessarily completely formalized models, helps to obtain elements of responses to the questions asked by a stakeholder of a decision process. These elements work towards clarifying the decision and usually towards recommending, or simply favoring, a behavior that will increase the consistency between the evolution of the process and this stakeholder’s objectives and value system.”

In many real-life situations, making good decisions is not an easy task. It requires considering the characteristics of the environment in which the decision will be taken and the different dimensions from which the problem might be apprehended [138]. These characteristics can be either:

- The stakeholders' characteristics: The stakeholders are the different entities either concerned by the outcome of the decision, such as a user or a citizen, or intervening in the decision aiding processes as decision makers or analysts. The characteristics of these stakeholders, such as their positions and their preferences, should be taken into consideration all along the decision aiding process.
- The problem's characteristics: Most of decision making problems involve multiple dimensions, ranging from multiple endogenous and exogenous sources of uncertainty, multiple criteria characterizing the decision problem, multiple decision makers, and time constraints.

Given the various dimensions of a decision situation, making a choice may be difficult in many situations; some form of help or *decision aid* may be necessary. For instance, seeking for a travel destination after consulting several blogs may lead to a hazy situation. Such confusion might be either due to an important number of alternatives, uncertainties related to the weather during vacations or alternatives are evaluated on several points of view that may be conflicting such as the price, the weather, etc. Thus, in this example, the traveler preferences should be learned and the different uncertainties should be modeled. Let's mention that modelling the uncertainties is out of the scope of this thesis. The complexity surrounding many decision making problems has led to the emergence of many developments on the way a decision aiding should be conducted, under the name of decision aiding process.

1.2.1 Decision Aiding Process

Decision aiding process is an interactive exchange between at least two actors : the *decision maker* and the *analyst* [117,137]:

- Decision Maker is an entity responsible of making decisions. Such entity is not necessarily aware of the decision aiding tools. When a decision making

problem is complex, the decision maker requests the help of an analyst to facilitate the task.

- Decision Analyst is an entity in charge of using analytical tools in order to provide the decision maker by synthesized information allowing her to take informed decisions. The decision analyst is supposed to be aware of the way a decision aiding should be conducted.

Decision aiding process consists, on the one hand, in structuring and formalizing an ill-structured decision problem in order to make it more understandable. On the other hand, it consists in using the analytics in order to assist the decision maker in her decision making problem. The formalization of a decision aiding process can be performed as follows [137]:

- **Representation of the problem situation:** This step allows the analyst to structure the decision problem by highlighting the following elements:
 - stakeholders involved in the decision aiding process;
 - characteristics related to decision problems;
 - the legitimacy of the decision maker in the decision aiding process.
- **Problem formulation:** This step allows the analyst to formalize the decision problem by highlighting the following elements:
 - potential actions;
 - point of view evaluating objects;
 - problem statement.

Problem statement refers to the type of the decision aiding problem corresponding to the decision maker's needs.

- **The evaluation model:** After formalizing the decision problem, the analyst synthesizes the collected information. Then, she chooses an adapted model for the decision problem. The model should be explicit and understandable

by the decision maker and use correctly the generated information. Hence, it allows the decision maker to make a justifiable decision.

- **Final recommendation:** During the decision aiding process, the decision maker benefits from the assistance of the analyst who facilitates the problem's understanding and propose to her a final recommendation derived from the evaluation model. The final recommendation should satisfy the following conditions:
 - Non-sensitivity to non-significant variations of evaluation model parameters;
 - The model is understandable to the decision maker and reflect her preferences;
 - The decision maker is legitimate or not to make such decision.

Decision aiding problems are characterized by a set of decision objects also called *alternatives* in the decision aiding literature. Thus, all along this document, we will refer to the decision objects by alternatives and X to refer to the set of alternatives. In principle, this set could be infinite; nevertheless, in this thesis, we focus on ordinal classification problems in which the set of alternatives X is a finite set of m alternatives.

Decision aiding requires representing the decision maker's preferences. For this aim, one of the main steps that should be carried out by the decision analyst consists in using analytical techniques for preference modeling.

1.2.2 Preference Modeling

Modeling the decision maker's preferences is one of the main steps in the majority of decision aiding processes. It consists in identifying and modeling the preferences of a decision maker over a set of alternatives characterizing the problem situation. Many preference modeling techniques are based on pairwise comparison of alternatives.

The notion of binary relation [12, 109] is established in order to compare the alternatives.

Definition 1 (*binary relation*) Given a set X , a binary relation R on the set X is a subset of the cartesian product $X \times X$, that is, a set of ordered pairs (x, y) such that x and y are in X : $R \subseteq X \times X$. For an ordered pair (x, y) which belongs to R , we generally use the notation: xRy .

Each binary relation has characteristics and properties that we mention in the following definition.

Definition 2 For the set X , a binary relation R is said to be

- Reflexive if $xRx; \forall x \in X$
- Irreflexive if $\neg(xRx); \forall x \in X$
- Symmetric if $xRy \implies yRx; \forall x, y \in X$
- Antisymmetric if $(xRy \text{ and } yRx) \implies x = y; \forall x, y \in X$
- Asymmetric if $xRy \implies \neg(yRx); \forall x, y \in X$
- complete if $(xRy \text{ or } yRx); \forall x \neq y \in X$
- Strongly complete if $xRy \text{ or } yRx; \forall x, y \in X$
- Transitive if $(xRy \text{ and } yRz) \implies xRz; \forall x, y, z \in X$
- Negatively transitive if $xRy \implies (xRz \text{ or } zRy); \forall x, y, z \in X$
- Semi transitive if $(xRy \text{ and } yRz) \implies (xRt \text{ or } tRz); \forall x, y, z, t \in X$
- Ferrers relation if $xRy \text{ and } zRt \implies xRt \text{ or } zRy; \forall x, y, z, t \in X$

In general, when the decision maker confronts the situation of comparing two alternatives, three attitudes may be considered. For any x and y in X , one and only one of the followings situations hold:

- the decision maker prefers x to y (or y to x). Such a relation is denoted by xPy .
- the decision maker is indifferent between x and y . Such a relation is denoted by xIy .
- the decision maker can not compare x and y and these two alternatives are considered as incomparable. Such a relation is denoted by xJy .

The incomparability relation occurs in certain situations where the decision maker is not able to compare two alternatives (she can not express the preference or indifference between two alternatives) such as lack of information, uncertainty, ambiguity, multidimensional and conflicting preferences. These three relations allow to define a structure on the set of alternatives X .

Definition 3 (*Preference structure*) A preference structure is a tuple of binary relations $\langle P, I, J \rangle$ defined on the set X verifying the following properties:

- I is reflexive, symmetric;
- J is irreflexive, symmetric;
- P is asymmetric;
- P , I and J are mutually exclusive: $P \cap I = I \cap J = J \cap P = \emptyset$;
- $P \cup I \cup J \cup P^{-1}$ is complete.

For each preference structure, there is a unique relation from which the different relations composing the preference structure can be considered.

Definition 4 ($\langle P, I \rangle$ Structure) A $\langle P, I \rangle$ preference structure is a tuple defined on the set X such that:

- I is reflexive, symmetric;

- P is asymmetric;
- P and I are complete and exclusive;

Given the structure $\langle P, I \rangle$, the binary relation $\succeq := P \cup I$ is called the characteristic relation:

$$x \succeq y \iff x(P \cup I)y.$$

P and I can be defined from $\succeq$ as follows:

$$\begin{cases} xPy \iff x \succeq y \text{ and } \neg(y \succeq x) \\ xIy \iff x \succeq y \text{ and } y \succeq x \end{cases}$$

I represents the symmetric part of $\succeq$ and P represents the asymmetric part of $\succeq$. This relation is generally referred as "outranking". In fact, $x \succeq y$ or x outranks y means that x is at least as good as y . Based on the properties of binary relations, we can define different types of structures.

Definition 5 (Total order) Let $\succeq$ be a binary relation on the set X , $\succeq$ being a characteristic relation of $\langle P, I \rangle$, the following definitions are equivalent:

- i. $\succeq$ is a total order;
- ii. $\succeq$ is reflexive, antisymmetric, complete and transitive ;
- iii. $\begin{cases} I = \{(x, x), \forall x \in X\} \\ P \text{ is transitive} \\ P \cup I \text{ is complete} \end{cases}$

The total order consists in a ranking of objects from the best one to the worst one without any ex-aequo. We now introduce the notion of weak orders, allowing ties between alternatives

Definition 6 (Weak order) Let $\succeq$ be a binary relation on the set X , $\succeq$ being a characteristic relation of $\langle P, I \rangle$, the following definitions are equivalent:

- i. $\succeq$ is a weak order;
- ii. $\succeq$ is a reflexive, complete and transitive;
- iii. $\left\{ \begin{array}{l} I \text{ is transitive} \\ P \text{ is transitive} \\ P \cup I \text{ is reflexive and complete} \end{array} \right.$

Contrary to the precedent order, weak order allows ex-aequo. These two structures consider indifference as a transitive relation. However, the decision maker may express an intransitive indifference (we mention in this context the famous example of a cup of sweetened coffee presented by Luce [95]). Such intransitivity can be modeled by introducing preference relation with a discrimination thresholds [10]. Relaxing the property of indifference's transitivity derives two structures: semi-order and interval order. The case of intransitive indifference will not be detailed in this chapter, as we do not consider it in our work.

The numerical representations of total orders and weak orders make use of functions, so that each alternative can be represented by a real number. Other orders having some more complex properties demand more complicated representations such as intervals.

For instance the numerical representation of weak order:

Theorem 1 *Let $\succeq$ be a reflexive relation of $\langle P, I \rangle$ on a finite set X , the following definitions are equivalent:*

- i. $\succeq$ is a weak order;
- ii. $\exists g : X \rightarrow \mathbb{R}^+$ satisfying $\forall x, y \in X$:

$$\left\{ \begin{array}{l} xPy \iff g(x) > g(y) \\ xIy \iff g(x) = g(y) \end{array} \right.$$
- iii. $\exists g : X \rightarrow \mathbb{R}^+$ satisfying $\forall x, y \in X$:

$$x \succeq y \iff g(x) \geq g(y)$$

When incomparability is added to the preceding orders, partial structures are derived (partial order, partial weak order, partial semi-order and partial interval order). This means that the incomparability relation is not empty.

Definition 7 (*Partial weak order*) Let $\succeq$ be a binary relation ($\succeq = P \cup I \cup J$) on the set X , $\succeq$ being a characteristic relation of $\langle P, I, J \rangle$, the following definitions are equivalent:

- i. $\succeq$ is a partial weak order;
- ii. $\succeq$ is reflexive and transitive
- iii. $\left\{ \begin{array}{l} P \text{ is irreflexive, asymmetric and transitive} \\ I \text{ is reflexive, symmetric and transitive} \\ J \text{ is irreflexive and symmetric} \\ P.I \cup I.P \subset P \end{array} \right.$.

In our work, we used a decision aiding method based on partial weak orders to compare alternatives evaluated based on several dimensions called criteria. In the next section, we will detail the particular case, related to our work, where decision aiding involves multiple criteria.

1.2.3 Multiple Criteria Decision Aiding

Using different preference modeling techniques, the alternatives need to be analyzed and assessed by the decision analyst in order to give recommendations to the decision maker. For some decision aiding problems, the alternatives can be assessed with respect to one dimension. In such case, we are facing a mono-objective optimization problem. However, more often, the assessment of alternatives involves different points of view. Multiple Criteria Decision Aiding (MCDA) is a branch of decision science taking into account different point of view from which the alternatives might be evaluated. Each point of view can be seen as a criterion.

Example 1.2.1 *Let's take the example of Beatrice who wants to travel and who has consulted several travel blogs on the internet in order to have recommendations of destinations. At the end of her research, she collected several options. Let's focus on two of them: New York and Bali. Each destination is evaluated on two criteria: climate and cost of living. Beatrice decides to travel in winter and she prefers a destination that has a cold climate and that is not expensive. Therefore, the criteria are conflicting. New York is a destination that has a cold climate but has a high cost of living, while Bali has a warm climate and a low cost of living. Hence, the task of making a decision becomes more difficult.*

The previous example shows the difficulty of decision making when alternatives are evaluated on several criteria that can be conflicting. In fact, in the MCDA frameworks, we don't seek for an optimal solution¹ but for a compromise solutions. Such a solution depends on the problem statement. According to Roy [117], multiple criteria decision problems may be classified into three types:

- *choice*, consists in selecting a singleton or a subset of the "best" possible objects (for example choosing the best job candidate after a hiring process);
- *ranking*, consists in ordering objects from the best to the worst (for example ranking students according to their annual marks);
- *sorting*, consists in classifying objects into ordered categories (for example sorting application files of scholarships' allocation into "refused", "accepted" categories)

Unlike the choice or ranking problems which depend on the relative evaluations of the alternatives (i.e. the ranking or choice of alternatives is done by comparing them to each other), the problems of sorting depend on the absolute evaluations of the alternatives (i.e. the classification of an alternative to a category depends on its intrinsic evaluations). For instance, choosing a travel destination is based

¹In case of conflicting criteria, an optimal solution, a solution having the best performance with respect to all criteria, might not exist.

on pairwise comparison between the alternatives. However, the chosen destination, the best one compared to the other alternatives, can be a bad one. Classifying it as “bad” or “good” requires qualifying its performance independently of the other destinations.

In this thesis, we will mainly focus on sorting problems also called *ordinal classification problems*.

Example 1.2.2 *Beatrice has decided to request a specialist (decision analyst) to assist her in making an appropriate decision regarding the travel destination. The specialist is supposed to help Beatrice to classify her options into three categories ordered from the worst to the best: “Bad”, “Medium”, “Good”. As mentioned in the example 1.2.1, each alternative is evaluated on climate and cost of living criteria. Beatrice gives more importance to the cost of living criteria compared to the climate. Therefore, the specialist is supposed to classify these destinations considering Beatrice’s preferences.*

Before presenting the multiple criteria methods, we will first present different concepts and notations related to the ordinal classification problems.

- **Criteria**

Each alternative is assessed under a set of n criteria $\mathcal{C} = \{cr_1, \dots, cr_n\}$. Criteria associate alternatives with a performance evaluation; (with little abuse of notation, we use x to refer to both the alternative and its evaluation vector). The *evaluation of x* is a vector $(x_1, x_2, \dots, x_n) \in \mathcal{E}_1 \times \dots \mathcal{E}_n$. The sets $\mathcal{E}_1, \dots, \mathcal{E}_n$ are totally ordered and represent the space of criteria evaluations. In our work, we are interested in ordinal classification methods that allow the criteria evaluation to be defined on ordinal scales.

- **Categories** Multi-criteria ordinal classification methods allow to assign alternatives to ordered categories. The *set of categories* is denoted by $\mathcal{C} = \{C_1, C_2, \dots, C_p\}$. Categories are ordinal, C_1 being the worst and C_p the best one.

Main multicriteria methods

There exist three types of Multicriteria methods. Scoring or utility-based methods, Outranking methods and Rule-based methods.

The first type is based on aggregating criteria values for each alternative into a single score [22, 37, 38, 54, 77, 83, 151]. These scores, as synthesized information, will be compared either in order to classify the associated alternatives, to rank them, or to choose the best one(s).

The second type is outranking methods [3, 29, 49, 50, 62, 92, 118, 148]. This type is based on outranking relations, used in order to assess the way the alternatives are compared to each other without aggregating their criteria values. An alternative x outranks an alternative y , if there exists a qualified majority of criteria in favor of x , and there is no qualified minority of criteria strongly rejecting such preference.

The third type is rule-based approaches [63–65, 67]. These approaches consist in learning decision rules (logical rules that separate the criteria space into multiple regions) from sets of predefined examples (learning and validation sets). The generated rules will be then tested on unseen data (a testing set), and based on their performance, they will be used in order to make a classification.

These types of methods will be detailed in the rest of this section:

- Multiattribute Utility Theory [54, 83]: Models based on multi-attribute utility theory are applied in the case where the decision maker's preferences form a weak order (see Definition 8). Thus, a numerical function can present its preferences (see Theorem 1). The numerical representation of two alternatives $x, y \in X$ is presented as follows:

$$x \succeq y \iff u(x) \geq u(y)$$

The numerical representation function is called the utility function. The utility function reflects the order of preference between x and y .

Using this utility function, we do not need to define a binary relation for each pair of alternatives in X , but simply assign a real number for each alternative in order to describe the decision maker's preferences. The computation of this real number is done by decomposing the global utility into sub-components for each criterion.

The most widely used utility function for measuring the decision maker's preferences is the additive utility function that we define as follows [77]:

$$U : \begin{cases} A & \longrightarrow \mathbb{R} \\ x & \longmapsto \sum_{i=1}^m u_i(x) \end{cases}$$

where $u_i(x)$ refers to the marginal (or local) utility function of criterion cr_i . These marginal utilities can be assessed either through a direct or indirect protocol such as for UTA² method [76]. UTA is an additive utility method, more precisely an ordinal regression method, which estimates, through linear programming, the parameters of the utility function based on information given by a subjective ranking.

The additive utility function is compensatory; intuitively this means that a bad performance on some criteria can be compensated by good ones on other criteria. Since the performances of alternatives are represented by scores, the comparison between alternatives is transitive.

The UTADIS (UTilités Additives DIScriminantes) [35, 152] method is a well-known variant of the UTA method, which is adapted to ordinal classification problems. Hence, it performs comparisons between the alternatives and the utility thresholds defining the lower bounds of each category. This method can be formulated as follows:

$$\begin{array}{ll} U(x) < u_1 & \implies x \in \mathcal{C}_1 \\ u_1 \leq U(x) < u_2 & \implies x \in \mathcal{C}_2 \\ \dots\dots\dots & \\ u_{p-1} \leq U(x) & \implies x \in \mathcal{C}_p \end{array}$$

²It stands for UTilité Additive which is the french translation of Additive utility.

- **Decision rule-based methods:** Decision rule-based methods are function-free. They are represented by their conditional form of the type “if ... then ...” decision rules. Among these methods, DRSA (Dominance-based Rough-Set Approach) [63–65] is one of the most widely used methods. DRSA is based on rough set theory which is a powerful tool to analyse and represent ambiguous preferences. It consists in assignment of some alternatives to some pre-defined and preference-ordered decision categories. The emphasis of DRSA is replacing the indiscernibility relation between objects, used in the rough set theory, by the dominance relation. This method is able to deal with the heterogeneous information including qualitative and quantitative and can handle inconsistent information. It consists in assignment of some alternatives to some pre-defined and preference-ordered decision categories.
- **Outranking methods:** Such methods [3, 4, 49, 92, 118] start by comparing alternatives according to each criterion and then aggregates these local comparisons based on two main conditions:
 - The concordance: it consists in the existence of a qualified majority supporting the outranking
 - The discordance: it consists in the absence of any qualified minority rejecting the outranking relation

Several outranking relation were mentioned in the beginning of this section. In the next section we will present in detail the ELECTRE family since a major part of our work will focus on its variant MR-Sort.

ELECTRE Family

To compare pairs of alternatives for each criterion, outranking methods are based on outranking relations. Such relation is built based on two main conditions the concordance and the discordance [51, 139].

Once outranking relation is built, it is necessary to formalize a recommendation

based on an exploitation procedure. Such procedure depends on the problem statement [50,51,61,62]. Within the ELECTRE Family, the methods can be partitioned into three classes based on the problem type:

- Choice problem: ELECTRE I, ELECTRE IV, and ELECTRE IS.
- Ranking problem: ELECTRE II, ELECTRE III, and ELECTRE IV.
- Sorting problem: ELECTRE TRI-B, ELECTRE TRI-nB, ELECTRE TRI-V, ELECTRE TRI-C, ELECTRE TRI-nC and MR-Sort.

ELECTRE TRI is an outranking method, made for ordinal classification problems. It is based on pairwise comparisons, using outranking relations, between alternatives and the reference profiles characterizing each category. In the original version of ELECTRE TRI, two exploitation procedures can be used in order to classify the alternatives: the *pessimistic* and the *optimistic* one.

The pessimistic procedure is based on pairwise comparison between each alternative and the reference profiles starting from the worst to the best category. This procedure ends when a limiting profile outranks the alternative. The optimistic procedure is based on pairwise comparison between alternatives and limiting profiles starting from the best to the worst category. The procedure stops when the alternative outranks a limiting profile.

The major difference between the ELECTRE TRI variants is essentially based on how the categories are represented. In fact, such difference is presented as following:

- ELECTRE TRI-B [118,148]: categories are limited by single reference profiles.
- ELECTRE TRI-C [3]: categories are characterized by central profiles. The pairwise comparison between alternatives and central profiles is done simultaneously via the optimistic and pessimistic and procedures.
- ELECTRE TRI-nB [49]: categories are limited by a set of limiting profiles;

- ELECTRE TRI-nC [4]: categories are characterized by a set of central profiles.

Another variant of ELECTRE TRI is the *Majority Rule Sorting (MR-Sort)*. MR-Sort [93] is a multi-criteria ordinal classification method allowing to learn the model's parameters from a set of assignment examples (a learning set) and to use the learned model in the assignment of alternatives to ordered categories. Each category C_h is characterized by two parameters (to be learned) corresponding to “fictitious” items $b^h = (b_1^h, \dots, b_n^h) \in \mathcal{E}_1 \times \dots \mathcal{E}_n$ and $b^{h+1} = (b_1^{h+1}, \dots, b_n^{h+1}) \in \mathcal{E}_1 \times \dots \mathcal{E}_n$. These parameters are called *limiting profiles*. The set of limiting profiles is denoted by $B = \{b^1, b^2, \dots, b^{p+1}\}$. Limiting profiles play the roles of lower and upper bounds of the categories. One of the main characteristics of these profiles consists in a dominance relation between the profiles characterizing each adjacent categories. Thus, a limiting profile characterizing a category will dominate all the limiting profiles characterizing lower categories, this can be formulated as follows:

$$\forall k \in \{1, \dots, n\}, b_k^{i+1} \geq_k b_k^i \text{ and } \exists j \in \{1, \dots, n\}, b_j^{i+1} >_k b_j^i$$

where $\geq_k$ is a weak order comparing the assessment of the alternatives under the cr_k . There are two special limiting profiles, b^1 and b^{p+1} that are defined as the minimum and the maximum values: $b^1 = (b_1^1, \dots, b_n^1)$, such that $\forall i, b_i^1 = \min_{x \in X}(x_i)$ and $b^{p+1} = (b_1^{p+1}, \dots, b_n^{p+1})$, such that $b_i^{p+1} = \max_{x \in X}(x_i)$.

In the MR-Sort model, we need to learn the association of each criterion cr_i with a numerical weight w_i , that intuitively represents its importance to the decision maker. In addition to the weights and the limiting profiles, we need also to learn the *majority threshold* (λ), representing the qualified global weight allowing to have an outranking relation $\succeq$.

When the MR-Sort model's parameters are learned, the model can be used for pairwise comparisons between objects of X and limiting profiles. An alternative x is assigned to the category C_h if it is “at least as good as” the lower limiting profile b^h and it is not “at least as good as” the upper limiting profile b^{h+1} according to

an outranking relation $\succeq$.

$$x \rightarrow C_h \iff x \succeq b^h \text{ and } x \not\succeq b^{h+1} \quad (1.1)$$

where $x \rightarrow C_h$ means that alternative x is assigned to the category C_h . An item x that is indifferent³ to the upper limit profile of the best category ($x \sim b^{p+1}$, meaning $x \succeq b^{p+1}$ and $x \preceq b^{p+1}$) is assigned to C_p .

An important part of this thesis is based on MR-Sort method. Although MR-Sort allows also to include vetoes (show strong rejection of binary relations by qualified minority of criteria) in the assessment of the MR-Sort model, in our work, we will consider a simplified version based only on the concordance: $\succeq \subseteq (\mathcal{E}_1 \times \dots \mathcal{E}_n) \times (\mathcal{E}_1 \times \dots \mathcal{E}_n)$:

$$x \succeq b^h \iff \sum_{i: x_i \geq b_i^h} w_i \geq \lambda \quad (1.2)$$

Outranking methods are characterized by their non-compensatory nature. Compensation of performances among criteria is controlled. On the other hand, they deal with criteria evaluated on an ordinal scale. Outranking methods can model complex preferences such as the incomparability and the non-transitivity. Incomparability occurs when we cannot have an outranking relation between the compared alternatives. Non-transitivity might come from the Condorcet paradox [60] where preferences might be non-transitive.

1.3 Preference elicitation

Preference elicitation aims at eliciting the decision maker's preferences and incorporate them in the decision model in order to provide her a personalised recommendation. Several methods have been proposed to learn decision makers' preferences. Some of these methods are based on an interactive process between the analyst and

³Indifference is defined as: $x \sim y \iff x \succeq y \text{ and } y \succeq x$.

the decision maker in which the analyst asks several questions in order to estimate and assess the model's parameters. For instance, the Simos' direct elicitation procedure was developed by Simos [126,127] and extended later by Figuera and Roy [50] to determine the ELECTRE TRI parameters. Such a method elicitates, based on an arrangement of criteria cards, the criteria importance for a decision maker.

One of the major issues when learning directly the model's parameters, consists in the variety of the meanings that can be taken by the weights of criteria as we change the aggregation method [119]. To illustrate, let's consider the example cited by Mousseau [102], that consists in two alternatives $a = (15, 10, 10)$ and $b = (10, 12, 12)$ and a decision maker who has the same preference for the three criteria. By applying the weighted sum method, and considering the same weight, a will be preferred to b since the score of a (11.67) is higher than the score of b (11.33). Instead, if the majority method, choosing the alternative that is preferred by the majority of criteria, is applied, b will be preferred to a since b is greater than a in two criteria. Thus, we have two different preference relations by setting the weights of the criteria but changing the aggregation method. This is because the weights in a weighting sum aggregation method represent the trade-off between criteria while in the Electre methods they represent the voting power of criteria. Thus, the weights take two different semantics. To overcome this problem, two solutions are possible. On the one hand, the analyst can try to explain the aggregation method to the decision maker before getting involved in a direct elicitation process. However, the decision maker is not supposed to understand the analytics behind the decision aiding methods. Alternatively, indirect approaches allow to infer the preference parameters with a minimum cognitive burden. The decision maker is asked about her preferences over alternatives such as comparing two alternatives or assigning an alternative to a given category. However, the decision maker cannot be asked directly to assess the parameters such as weight, etc. Based on these preferential information, a *disaggregation* step is engaged to infer model parameters. This approach is called as well *aggregation-disaggregation*. Such approach derives an aggregation model compatible with known preferences. These preferences are determined by asking the decision maker about alternatives and the question de-

depends on the problem statement. In fact, she can be asked to rank alternatives, make pairwise comparison between them or assigning them to predefined categories. Then, her answer is incorporated as input in the disaggregation model. The following section presents elicitation approaches in the context of ordinal classification problems.

1.3.1 Preference Elicitation for ordinal classification methods

We introduce the preference elicitation methods for the main multicriteria ordinal classification procedure presented in section 1.2.3 except for decision rule based methods which don't imply explicit parameters as for MAUT or outranking methods.

- Multiattribute Utility Theory (MAUT): Disaggregation approaches are considered for MAUT methods to set the values of involved parameters (mainly the set of additive utility function and utility thresholds). For sorting problem, UTADis [32] is based on an aggregation disaggregation approach: disaggregation estimates the parameters of the model. In fact, the decision maker is asked to classify alternatives to categories and her answers are integrated into an optimization model. Such an optimization model uses as variables the set of additive utility functions and utility thresholds in order to infer them and, represents examples of assignments given by the decision maker as constraints. There exist two types of errors relative to the global utility $U(x)$ of an alternative $x \in X$: the over-estimation error $\sigma^+(x)$ and the under-estimation error $\sigma^-(x)$. An over-estimation error occurs in case an alternative according to its utility is classified in a lower category than its initially predefined category. Thus, the value of $\sigma^+(x)$ must be added to its utility to correct its classification. An under-estimation error occurs in case an alternative according to its utility is classified in a higher category than its initially predefined category. Thus, the value of $\sigma^-(x)$ must be subtracted from its utility to correct its classification.

The values of $\sigma^-(x)$, $\forall x \in X$ and $\sigma^+(x)$, $\forall x \in X$ are the decision variables of the linear program. The examples of the decision makers' assignments are

respected by the following two constraints:

$$\left. \begin{array}{l} U(x) + \sigma^-(x) < u_{k-1} \\ U(x) - \sigma^+(x) \geq u_k \end{array} \right\} \forall x \in C_k$$

The parameters of the model are learned using a linear program whose objective function consists in minimizing the sum of errors (misclassification errors) which represents the difference between the classification of the alternatives derived from the model and their pre-defined classifications:

$$\min_{\forall x \in X} \sigma^+(x) + \sigma^-(x)$$

when $\sigma^+(x) + \sigma^-(x) = 0$, it means that the model reflects perfectly all decision maker's assignment examples.

- **Outranking methods:** In order to use ordinal classification methods in practice, it is necessary to undertake a preference elicitation phase in order to assess the parameters of the model. Initially, Mousseau and Słowiński [104] have proposed a model that infers all the parameters of ELECTRE TRI based on a non-linear optimization model. Such a model learns simultaneously profiles, weights and thresholds by representing the assignment examples of a decision maker as constraints. Then, Mousseau et al., [103] proposed a linear program that partially infers the parameters of ELECTRE TRI assuming that the other parameters are known.

Dampous et al., [36] propose an evolutionary approach, based on the differential evolution algorithm that learns all ELECTRI TRI parameters. Hence, the differential evolution algorithm makes use of genetic operators such as mutation crossover and selection to determine an initial population of solution until a termination criteria is reached.

Leroy et al. [92] proposed disaggregation tools of MR-Sort model without vetoes. The first tool is based on MIP formulation which takes a set of assignment examples and their performance vectors as a learning set and infers the parameters of MR-Sort. This formulation is not adapted for large datasets

since it requires high computing time to generate the model's parameters. Hence a heuristic algorithm [130] is defined. This heuristic initializes several populations of MR-Sort model then performs the following two steps:

- Apply a linear model optimizing the weight and majority threshold based on assignment examples and setting the reference profiles;
- Apply a heuristic that considers the given inferred weight and majority thresholds and adjusts the reference profiles based on the assignment examples.

These two steps are repeated for each population until obtaining a model that restores all assignment examples. These tools learn the parameters of MR-Sort in one shot based on the decision maker's assignment examples. However, in case we are dealing with an important number of alternatives, it is not practical to ask the decision maker to assign all of them in order to generate a model satisfying her preferences. Thus, inferring MR-Sort parameters through an incremental approach allow to reduce the cognitive burden while learning the decision maker's preferences.

Another work presents an elicitation process for an ordinal classification problem. Such process is based on a stochastic approach using a Monte Carlo procedure [30, 31]. It starts with a learning set provided by the decision maker. Then, for each step, the process picks randomly a remaining alternative (that was not classified by the decision maker) and classify it to one category. Such classification is done randomly while respecting a monotonicity constraint regarding the classifications provided by the decision maker and those previously generated by the process. Moreover, it will generate new constraints for the next classifications because of the monotonicity. Since the choice of the remaining alternatives and their classifications are randomly generated by the process, several assignment scenarios are possible and each scenario is considered as a trial. In order to derive a single classification that better reflects the learning set provided by the decision maker, the process generates several trials that are probabilistically independent. At the end, the consecutive results

collected from the trials are aggregated to derive a single classification. This process may be considered as a black box for some decision makers. However, providing explanations to the decision maker is desirable in certain contexts such as policy decision making.

1.3.2 Incremental approach

To limit the cognitive burden of elicitation and the number of questions, an “incremental” approach [146] can be proposed. Such an approach asks interactively “informative” question, specifically chosen, to the decision makers.

Example 1.3.1 *As we showed in the example 1.2.2, Beatrice asked a decision analyst to help her in classifying one hundred destinations. This decision analyst knows that Beatrice classifies New York in the category “Good” and Bali in the category “Medium”. However, he has no idea about Beatrice’s classifications for the rest of the destinations and her preferences (such as which criterion between climate and cost of living is more important for her). Consequently, he will try to understand her preferences by asking her questions about destinations. Such questions should be as informative as possible in order to avoid an important cognitive effort on Beatrice.*

Interactive utility elicitation is well studied in the literature. In many decision aiding problems, decision maker’s utility functions might be either uncertain or unknown. One way is to compute it through a prior distribution by adopting a Bayesian approach and assuming a prior distribution on the parameters of utility function. Such distribution might be assessed by learning from historical data characterizing similar decision makers. In case the interaction with the decision maker is not possible, alternatives might be assessed based on the expected utility function. However, in case there is a possibility to interact with the decision maker, the distribution is updated with a Bayesian approach [20, 68, 143]. Thus, the prior distribution can be updated based on the decision maker’s feedback. This approach is proceeding by asking sequentially informative questions that improve the decision

maker's expected utility. The Bayesian approach derives relevant recommendations by asking relatively few questions to the decision maker. Such approaches might be used in the context of choice, ranking and ordinal classification. However, it is computationally expensive due to the need of maintaining a probability distribution over the utility function's parameters and performing costly Bayesian updates.

Max-margin optimization is an alternative to the Bayesian approach. The main idea of max-margin is to estimate utility functions which are defined in terms of a set of weight vectors from the preferences provided by the decision maker. These preferences are generally expressed by a set of constraints indicating the preference of an alternative compared to another one in order to infer the weight values satisfying these constraints. With the aim of maximizing the satisfaction degree of constraints, a shared margin is added as a decision variable and maximized in the objective function. At the same time, slack variables are used to ensure the success of optimization even in the presence of contradictory constraints (the decision maker may express her feedback in an inconsistent way regarding her previous ones). Gajos and Weld [55] proposed a system based on max-margin optimization aiming at eliciting interactively the preferences of a user in the context of personalized user interfaces. The choice of the next question within this system is based on heuristic approaches in the context of SVM (Support Vector Machine) training process that aims at choosing the question that maximally reduces the size of the remaining space presenting the set of all feasible solutions. MR-Sort method is also based on max-margin optimization. Hence, to generate the parameters of MR-Sort, users' feedback representing the classifications of alternatives to categories are presented as constraints. A shared margin is used to maximise the degree of certainty of these classifications and slack variables are added to handle the contradictory constraints. In fact, as we mention in the previous section, the elicitation of a MR-Sort model is done in one shot. The decision maker's feedback is given as a learning set and based on that, the parameters of the model are inferred.

Another method that extends max-margin to generate a set of solutions rather than a single one is setwise max-margin [135, 136]. Such a method has several

purposes: it produces a set of "diverse" alternatives used to ask informative questions to the decision maker; it encourages the sparsity of utility functions in terms of weight combinations; it optimizes the recommendation by taking into account the collected information and it deals with inconsistencies with respect to decision maker responses. The recommendation is made by solving a constrained optimization problem.

For ordinal classification problems, Bennabou et al. [8,9] recently proposed incremental elicitation methods based on minmax regret. The choice of question is made in an iterative way until a certain minmax regret value is lower than a threshold representing a bound of the error considered acceptable by the decision maker. At this point, the system can recommend to the decision maker an optimal solution in terms of the minimax regret criterion. Such a solution minimizes the value of max regret (which represents the greatest loss in utility) that can be generated by recommending one alternative rather than the "true" optimal one. Recommending such a solution ensures that the loss of utility in the worst case is minimized. However, this method is not tolerant to errors in the responses of the decision maker.

1.4 Synthesis

Designing a decision model able to reflect the decision maker's preferences in real-life situations is not an easy task. Several decision aiding methods have been developed to solve choice, ranking and classification problems, assessed under multiple conflicting criteria. By learning the decision maker's preferences, these methods provide her by personalized recommendations. A detailed literature review associated to these methods is proposed in this chapter. These methods can be partitioned in compensatory methods and non compensatory methods (including the methods for which the compensation is limited).

In our work, we are interested in outranking methods which are non compensatory. These methods are based on pairwise comparisons, either among the alter-

natives or between the alternatives and reference profiles. MR-Sort is one of these methods. It allows to learn, through an aggregation/disaggregation approach, the parameters characterizing a decision maker's preferences based on a set of learning examples. One of the major problems within this method consists in the high cognitive burden associated the construction of the learning examples. To overcome this problem, we developed an interactive active learning process, based on MR-Sort method, allowing to choose the alternatives for which the assignment will be informative. Within this approach the decision maker is asked, interactively, about the assignment of well-chosen alternatives in order to infer her model parameters while limiting the cognitive burden.

The proposed interactive approach presents the first contribution of this thesis in which we will assist the decision maker to better understand her preferences and ensure the classification of alternatives to ordered categories. The second contribution is to derive an ordinal classification for group decision markers. The next chapter will be dedicated to present the literature review associated to this aim.

Chapter 2

Group Decision Making and Automated Negotiation

Contents

2.1	Introduction	65
2.2	Group decision making	66
2.2.1	Group decision making for ranking problems	67
2.2.2	Group decision making for ordinal classification problems	71
2.3	Basics of Automated negotiation	74
2.3.1	Negotiation object	75
2.3.2	Participants	75
2.3.3	Negotiation protocol	77
2.3.4	Negotiation strategies	82
2.4	Synthesis	85

2.1 Introduction

The main goal of this thesis consists in deriving a group classification from individual classifications based on a negotiation process. In this chapter, we present the literature review related to this topic as follows:

- In Section 2.2, the group decision making literature in the context of ordinal classification.
- In Section 2.3, the basic concepts of automated negotiation.

2.2 Group decision making

The decision making in ordinal classification problems is not an easy task when it involves several stakeholders with divergent classifications. This can be illustrated by the following example.

Example 2.2.1 *Beatrice decided to travel to one of the three destinations New York, Bali and Paris with her friends Nicolas and Raja. Each of them classifies each destination in the 3 categories “Bad”, “Medium” and “Good”. In fact, Beatrice classifies New York in “Good”, Bali in “Medium” and Paris in “Bad”. Nicolas classifies New York and Paris in “Good” and Bali in “Bad”. Raja classifies the three destinations in “Medium”. Consequently, the three friends have different opinions and cannot agree on a common classification.*

In this section we detail how to derive a common ordinal classification for a group of decision makers.

Group decision making [6, 14, 16, 17, 27, 57, 74, 84] is a process in which multiple stakeholders, with individual preferences, act collectively to make a common decision. For instance, the executive committee members need to agree on several joint decisions in a variety of situations, such as setting a company strategy, sharing a budget, prioritizing projects, to name but a few.

Several researches were conducted in different sub-domains of decision science aiming at deriving a common decision. From these decision science domains we can mention *social choice theory* [84], *group recommender system* [57, 78, 98] *consensus reaching process* [33, 134]. Most of social science works, consider the group decision

making problems as voting problems, where different rules are used to derive a common decision such as Borda [11] and Condorcet [28]. In group recommender systems, similarities are assessed among the agents' preferences or the alternatives in order to rank alternatives. Most of the methods are extensions of the basic recommender systems such as collaborative filtering methods or its variant matrix factorization, or content based filtering. Consensus reaching process works are based on an iterative process where the decision makers revise their preferences in such a way to reach a consensus.

Although the notable contribution of these domains to the group decision making theory, most of these works are focused on ranking problems. In this section, we present the literature review associated to the contribution of the three domains in the area of group decision making theory. Then, we present the major contributions of group decision making theory in ordinal classification problems, which is the most important issue in this thesis.

2.2.1 Group decision making for ranking problems

In *social choice theory* [6], several methods have been designed to deal with group decision making problems. Social choice combines theories from social welfare and voting systems which are applied in several contexts. These contexts may raise important issues such as political elections (election of a president or of an assembly, etc.) or less important issues such as a group of friends that use a vote to decide in which restaurant they have lunch or a group of co-workers that use online voting to choose a meeting date.

In order to derive a common ranking, most of social choice approaches are inspired by voting systems. They consider group members as “voters”, and the alternatives as “candidates”. Thus, generating a ranking will depend on the chosen *voting rule*. For several voting approaches, the group decision process is centralized. In fact, a central agent collects the preferences of all the agents and aggregates them (without involving the agents) according to a voting rule to return, at the

end, a ranking that represents the group decision. Several voting rules exist in the literature, we shall mention two general rules: *scoring rules* and *pairwise comparison rules*.

- Scoring rules [13] assign a score to the candidates, based on their ranking in the votes. A score is associated to each ranking made by a voter based on a voting rule. The final score of a candidate is the sum of her scores overall the votes. Hence, the winner is the candidate with the highest total score. Among the most common scoring rules, we mention for instance: *plurality rule*, in which the score of the first place is 1, all other scores are 0, so the winner is the candidate who is most often ranked first; *borda rule* [11], in which the score decreases progressively from $m - 1$ for the first place to 0 for the last place, when there are m candidates; *k-approval rule* [14], which gives a point for k preferred candidates.
- Pairwise comparison rules, are based on comparing each pair of alternatives in order to rank them. To generate such ranking, several methods exist in the literature such as the Condorcet method or the Copeland method. An alternative is called a *Condorcet winner* [28], if she wins all her duels against the other candidates. A ranking might be generated by iteratively deleting the Condorcet winner and seek for a new winner. *Copland's rule* [26], counts the number of times an alternative beats the others, based on the majority rule, minus the number of times it loses; *the maximin rule* [87, 128], which selects alternatives for which the pairwise minimum majority is maximized.

Computational social choice tackles social choice problems with the tools of computer science (with an emphasis on algorithmic aspects) [14]. The computation of a solution is a major issue to be considered when solving concrete problems, especially those that include a high number of candidates and alternatives. Consequently, computational social choice consists in assessing and optimizing the computational complexity of social choice procedures. In particular, it is used in group recommender systems to derive group recommendation based on agents' preferences.

Group recommender system has become an active area of research dealing with the group decision issues. It aims at seeking items' recommendation that should satisfy as much as possible the individual preferences of all group members. Among the group recommendation systems, we mention for instance the collaborative travel advice system [101] which assists the users in choosing a common vacation or MusicFX [100] which chooses a radio station for background music taking into consideration tastes of people training at the same time in the same fitness center. Most of group recommender methods use aggregation strategies [19, 98] to derive recommendations based on agents' preferences. Such strategies are generally used to aggregate individual ratings into a group rating based on social choice theory. Aggregation methods, in particular voting rules aim at deriving efficient and fair recommendations. However, they reveal the followings problems:

- 1) *Condorcet's paradox* [97], states that the pairwise majority relation can be cyclic and violate transitivity¹;
- 2) *Arrow's impossibility theorem* [5], establishes that there is no voting system satisfying together several basic axioms;
- 3) *Gibbard-Satterthwaite's theorem* [58, 123], states that no voting rule that satisfies basic axioms is immune to manipulation (when agents decide to misreport their preferences to the central agents in order to make a specific candidate win or lose).

Since we are living in a dynamic environment, agents' preferences are not fixed, they can change over time for many reasons. In our work, we are interested by a particular case where a group of decision makers are involved in an interactive process allowing to generate a common ordinal classification of several alternatives. This type of problems rise a number of reasons leading to a change over the decision makers' behaviours and preferences. For instance, imagine that for a given decision maker, all the past common classifications were different from her own

¹given three alternatives a, b and c, suppose that a majority of voters prefers a to b, an other majority prefers b to c and a final majority prefers c to a. Such situation lead to a cycle and a violation of transitivity.

preferred classifications. This might lead to an inflexibility of the decision maker in the future. Thus, the number of concessions can be considered in our process. When a decision maker makes multiple concessions by accepting classifications different from her own, she may tend to become less cooperative, and thus make less concessions. Time also can represent an important factor influencing the decision makers' behavior. As we get close to a deadline where a decision should be taken, decision makers might tend to become more cooperative, and thus, tend to make more compromises. Hence, aggregation methods do not reflect truly the decision makers' preferences as their behaviors might change during the decision making process. To take into consideration the different changes over the decision makers' behaviors, group recommender system methods apply distributed approaches such as negotiation [56, 85, 144]. The purpose of negotiation is to manage the process of reaching agreement on given objects by a group of agents.

Another widely used process to derive a collective decision is consensus reaching process [34, 110, 141]. Such process is multi-round, and in each round, agents are able to interact, negotiate and modify their preferences to reach a collective agreement over a decision. The main steps of the consensus reaching process are as follows: first agents are informed about the parameters of the decision model and the alternatives from which we want to select the best ones. Then, each agent expresses her preferences on the alternatives generally in a fuzzy and imprecise way [18, 140, 142]. Once agents express their preferences, consensus measure will be established to determine whether or not a consensus level has been reached. In case the consensus level is satisfactory for the group, the process ends with success and the phase of selecting the best alternatives for the group is initiated. Otherwise, in case the consensus level is not satisfactory, agents are asked to review and modify their preferences through a feedback mechanism. Hence, for consensus reaching process, the final collective decision is generally about all the available alternatives. In fact, if agents do not reach an agreement on this final decision and the termination condition of the process is verified (such as the negotiation deadline reached), the model may not lead to a collective decision and the process fails.

Moreover, to the best of our knowledge, the different approaches that we pre-

sented previously were generally proposed with the goal of generating a ranking of all the alternatives and are not often used to deal with group ordinal classification problems. The process we developed in this thesis aims at deriving a group ordinal classification by using negotiation approach that we present with more details in the following sections.

2.2.2 Group decision making for ordinal classification problems

This section is dedicated to present a literature review associated to classification problems involving multiple decision makers. One way to solve such problems is to classify the alternatives based on a common preference relation system derived from individual agents' preferences [74]. A second way, consists in engaging the agents into an interactive process aiming at generating a common leaning set [27]. The generated learning set will be then used in an aggregation/disaggregation process allowing to learn the group preferences, and to classify the remaining alternatives. A third approach, consists in using Mixed Integer Linear Programming (MILP) in order to infer group category limits and vetoes from assignment examples [16, 17]. The generated shared parameters (category limits and vetoes) as well as each decision maker's assignment examples, will be used to restrict the weights and the majority thresholds for each decision maker. All these works considered the use of the ELECTRE TRI or MR-Sort methods by a group of decision makers.

Jabeur and Martel [74] have proposed an ordinal classification method aiming at deriving a collective assignment of alternatives to pre-defined categories. In this process, group members agree on some characteristics of the decision problem, such as the number of categories, the criteria and the reference profiles. Each group member, has a weight representing her importance, and builds her own *preference relation system* (p.r.s.) representing $\langle P, I, J \rangle$ structure by making pairwise comparisons between alternatives and the reference profiles. For each pair alternative and reference profile, a collective relation set is assessed. The elements in this set are collective preference relations minimizing the distance of Jabeur et al [75] from each agent's p.r.s with respect to her weight. These collective relations are used to

derive assignments of alternatives to pre-defined categories. In case the process provides more than a single collective assignment, an interactive procedure is engaged. This last stops under one of the following two circumstances:

- the success of the process when a consensus is reached over a collective assignment;
- the failure of the process when a consensus is not reached over a collective assignment. In fact, group members may not be able to agree on a collective assignment and the stopping rule occurs (for instance, the maximum number of iterations is reached).

Damart et al. [27] proposed a methodology assisting the decision makers in their interactive process aiming at sorting some exemplary alternatives. These assignment examples are then used, through an aggregation/disaggregation process to derive an ELECTRE TRI based group multicriteria aggregation model. Two main issues are addressed by the developed methodology: the way to make a whole group achieve a consensus on a set of robust assignment examples and a common vector of inferred weights; the way to maintain both individual and collective consistencies while sorting the alternatives. To achieve such aim, each decision maker provides a set of consistent² sorting examples and the corresponding robust sorting interval. Based on these intervals, the decision makers, supported by a decision analyst, try to agree on at least one assignment, otherwise the process ends. In case an agreement is reached, the corresponding examples are incorporated in both the collective model and the decision makers' models. The process stops if all the agents feel that the collective model is satisfactory.

Cailloux et al. [16, 17] have proposed an elicitation procedure to infer group category limits and/or vetoes from assignment examples provided by multiple decision makers. Based on these assignment examples of individual assignments, the developed procedure generates common category limits (and/or vetoes), and looks

²The decision maker can be supported by IRIS to solve the inconsistencies.

for individual preference (weights and majority threshold) with respect to the generated shared category limits. These inferred parameters are generated based on three mixed integer linear programming (MILP):

- The first MILP is called Infer Category Limits (ICL). It infers from the assignment examples the group category limits while keeping the variables weights and the majority thresholds at the individual level.
- The second MILP, called Infer Category Limits with Vetoes (ICLV), is proposed as an extension of ICL to infer, at the group level, the vetoes in addition to the category limits. As for the ICL, ICLV is based on the assignment examples, taken with respect to the criteria weights in the individual level.
- The last MILP, called Compute Weights Restrictions (CWR), this MILP generates weights restrictions given the shared model parameters and the assignment examples.

It is possible to encounter the situation where it is difficult to create a preference model satisfying each decision maker's assignment examples given the shared category limits. Thus, some of the involved decision makers might need to change their preferences regarding certain assignment examples in order to achieve a consensus group preference model. In case they are not flexible to revise their preferences, the process may end without reaching a preference model.

In the processes mentioned above, it is possible to not reach any agreement in case agents did not find a consensus and the stopping rule reached. Moreover, the process does not consider dynamic external factors that may influence the agent's behavior such as the agent's flexibility regarding time progress or the number of concessions already made in previous rounds of the process.

In this thesis, we propose a process where the agents are not supposed to validate the final collective decision, but they try to reach an agreement for each alternative and in each round. In addition, the emphasis on modelling the flexibility of the

agents with respect to time and concessions constitutes the originality of our multi-agent approach. At the end of the process, if there are still alternatives that were not collectively approved by the agents, a third-party makes a collective decision over these alternatives to guarantee the success of the process. This process will be presented and detailed in Chapter 4.

2.3 Basics of Automated negotiation

Negotiation remains an omnipresent act in everyone's daily life. We negotiate the price of a house with a seller, the salary for a new employment opportunity with a recruiter, the choice of travel destination with our family members, etc. The negotiation process is complex when it involves a high number of agents who have conflicting preferences, a considerable number of negotiation objects or when each agent defends her own interests. Automated negotiation [45, 79] occurs in such cases in order to improve the quality of negotiated outcomes and reduce the cost compared with human negotiators [45]. It is a blossoming research area at the heart of multiagent system [147] involving a set of autonomous agents acting faithfully on behalf of human agents. Such agents ensure a negotiation according to the rules dictated by the protocol governing their communications. Automated negotiation occurs in several situations such as *auctions* [1, 71, 145], *resource allocation* [24, 40, 41] or *collective decision making* [72, 86, 89, 115] which is the target of this thesis. The major challenge of the negotiation process for collective decision making is to design autonomous agents capable of making decisions reconciling between individual and group's objective. In the literature, various approaches have been proposed for automated negotiation. These include game theory approaches [25, 39, 88, 116, 149], heuristic approaches [42, 43, 47, 124] and argumentation approaches [82, 91, 112, 125]. In this thesis, we adopt an heuristic approach. The distinguishing feature of this approach is its adaptation to human agents and its suitability for taking into account several aspects in group decision making.

In order to design a negotiation process, we identify its main components: i)

negotiation objects which define the issues to be negotiated; ii) the participants which define the agents with their own objectives and preferences; iii) negotiation protocol which defines the rules that govern the negotiation; iv) individual strategies defines the internal reasoning of the involved agents.

2.3.1 Negotiation object

This component represents the objects on which agents negotiate. It may be, for instance, resources in order to determine the best way for distributing them to the members of a population and thus optimize a given objective [24, 40, 41], or a negotiation on the ranking or classification of alternatives in a collective decision making context [72, 86, 115]. It represents the space of possible proposals (also called deals) made by the agents during the negotiation process. Each proposal is characterized by one or multiple attributes (also called issues). Hence, the proposal can be defined according to the number of issues characterizing it as follows:

- a single-issue: agents negotiate over the value of a single attribute [114]. For instance, this attribute could be the price of a negotiation' object [21]. A proposal can be defined by multiple attributes where one is negotiable and the rest are fixed and non-negotiable [96];
- a multiple-issue: agents negotiate over the value of more than a single attribute such as price, time, quality, etc. These attributes can be independent or interdependent. For independent attributes, a negotiation on a part of the attributes can influence a future negotiation on the rest attributes. Contrary to the interdependent attributes, the negotiation process must take into account the dependency between them [43, 47, 48, 72].

2.3.2 Participants

In automated negotiations, each participant is represented by an intelligent agent who is able to act autonomously and flexibly within an environment in order to

achieve her objectives.

Agent's properties

Wooldridge define agents as follows [147]:

"Agents are computer systems with two important capabilities. First, they are at least to some extent capable of autonomous action of deciding for themselves what they need to do in order to satisfy their design objectives. Second, they are capable of interacting with other agents not simply by exchanging data, but by engaging in analogues of the kind of social activity that we all engage in every day of our lives: cooperation, coordination, negotiation, and the like".

In fact, the properties of an agent can be presented as follows :

- 1) Autonomy: the capacity of an agent to perceive the environment and to respond to the changes that occur in order to achieve her objectives without the need for an external controller to guide her;
- 2) Proactiveness: the ability of taking initiatives to achieve her goals. For this purpose, she is not subject to the invocation of another entity to act but may act on her own initiative;
- 3) Flexibility: the ability of adapting her behavior to her perception of the environment and participating in organizations (group) in order to better satisfy her goal;
- 4) Sociability: the capacity to interact with other agents in order to achieve her goals or to assist other agents in their activities.

Thus, an agent does not only interact with the environment but can also develop social skills. She may be aware of the existence of other agents and interact with them.

Negotiation's cardinality

Negotiation's cardinality represents the interactions between participants according to their number. It is presented by two main types:

- bilateral negotiation: it is a one-to-one negotiation where one participant negotiates with only one other participant [48, 115, 120, 149];
- multilateral negotiation: it involves one-to-many and many-to-many negotiation. For one-to-many negotiation, one participant negotiates with more than one participant. The agent who negotiates with several other agents has significant power and influence on them. This is determined by the fact that this agent is the only one who manages the negotiation process [39, 72, 94, 111, 124]. For many-to-many negotiation, many participants negotiate with many other participants at the same time [7].

Identifying the number of participants is important in a negotiation process. This dimension influences the number of computations or interactions and thus influences the complexity of the process.

2.3.3 Negotiation protocol

An automated negotiation follows a protocol that sets the communication rules to be respected by the involved stakeholders. This protocol defines :

- the type of participants allowed in the negotiation;
- the different possible states during a negotiation;
- the events causing the states;
- the valid actions corresponding to each state.

The rules dictated by the negotiation protocol define the interactions between the agents, impose constraints on the proposals made by the agents, specify when a deal might be proposed and when an agreement might be reached over a deal. They also define situations under which negotiation fails. These rules may guarantee certain properties which are desirable for any negotiation protocol.

Negotiation protocol proprieties

A negotiation protocol may be evaluated according to the following properties [39, 45]:

- the guarantee of success: ensuring that an agreement is reached at the end of the negotiation;
- Pareto efficiency: ensuring that no other solution improves the outcome of one agent without decreasing the outcome of another;
- individual rationality: ensuring that all agents have an interest in participating in the negotiation process;
- stability: allowing all agents to participate in the negotiation and act in a way that achieves their goals;
- simplicity: allowing the agent to identify her optimal strategy easily;
- distribution: minimizing communication between agents. The protocol must be distributed (without mediator);
- comprehensibility: allowing humans to understand decisions made by automated negotiators acting on their behalf;
- timeliness: ensuring that between two negotiation processes leading to the same result, we would prefer the one that ended earlier.

Types of negotiation protocols

Several types of negotiation protocol have been proposed in the literature. Among these types we present the followings:

Auctions protocols: Such protocols are a form of negotiation involving, on the one hand, a seller (auctioneer) making a request and, on the other hand, competing buyers (bidders) making offers that aim at reaching an agreement [122]. The principle is to award the product to the highest bidder. An auction mechanism allocates resources to buyers and sellers based on predefined rules. These rules define the process of exchanging proposals, determining the winner and the final agreement. Among the well-known auction protocols, we mention *the dutch auctions*, *the first-price sealed-bid auctions* and *english auctions* [1, 69, 145]. For the dutch auctions, they are launched by a seller with a relatively high price. In case there is no buyer who expresses her interest, the seller continues to lower the price to a certain reserve price. Otherwise the object is sold to the buyer at the offered price. For first-price sealed-bid auctions, no visibility is given to buyers. The auctions take place in two successive stages: a stage of collecting secret offers submitted by the bidders in sealed envelopes and a stage of bids' evaluation. The buyer with the best offer is selected and she paid the price she offered; for the english auctions, they are launched by the seller with a base price. At each stage, a buyer makes a proposal, it is then put on hold and the other buyers are called to propose better. Auctions end when all bidders (except the best one) give up or when the time allowed for these auctions is over.

Contract Net Protocol (CNP): Such protocol is applied for the task allocation area [129]. According to this protocol, negotiation involves two agents: a manager, who manages the execution of a task and a contractor, who is the responsible for its execution. The protocol is initiated by the manager who sends a task announcement to all agents. Each agent verifies whether the announcement is suitable to her capacities and her current charges. If she approves the announcement, she sends a bid message to the manager. The manager allocates the contract to the contractor having made the best offer. This organisation is established in a context where

all agents are cooperative, i.e. they cooperate together to achieve a common goal. Otherwise, if agents are competitive, they accept or refuse a contract according to the marginal cost calculations [121].

Monotonic concession protocols: Such protocols were proposed by Zeuthen [149] in the context of a bilateral negotiation. The protocol is conducted by rounds and for each round both agents submit proposals simultaneously. Two possible cases arise during a given round: an agreement is reached when one agent accepts the proposal of her opponent, i.e. she rates her opponent's proposal at least as high as her own current proposal; conflict arises when both agents maintain their offer and refuse to make concession. This protocol has been generalized by Endriss [39] in the context of multilateral negotiation. He introduced seven different types of concessions that may be made by an agent favouring the acceptance of her next proposal by her opponent. This protocol requires a precise knowledge of the utility function of the agents involved, which is different from our context whereby we aim to keep each agent's utility functions private throughout the negotiation process.

Mediated and decentralized negotiation protocols: The interaction between agents can be conducted with or without a third-party. A negotiation that involves a third-party is a mediated negotiation [72, 73, 86]. This intervenor is called a *mediator* whose role changes according to the conception of negotiation process. Such a mediator has the authority over the interactions in the system. She works with agents who are in a conflictual situation to help them exchange better with each other and develop by themselves solutions for resolving the conflict. In some situations where agents fail to resolve the conflict, the mediator can act as an arbitrator and elaborate a solution that satisfies as much as possible all the agents. The mediator may be responsible for keeping a record of the exchanges that have taken place between negotiators, supervising the respect of the rules conducted by the negotiation protocol, initiating negotiations and not disclosing the private information of each agent to the others.

A decentralized negotiation is a negotiation that does not include third-party [39, 46, 81, 113, 115]. Each party of the negotiation evaluates the information from

her own perspective in order to reach an agreement that is acceptable by the whole group. In fact, an agent has the ability to perform her tasks and control them locally. In addition an agent is aware of her environment and information available on her system so she can decide when she interacts or cooperates with other agents in the system. Decentralized negotiation is desirable to avoid possible breakdowns of the central entity and bottlenecks.

Cooperative and Competitive Negotiation: A negotiation can be cooperative [91, 113, 129] or competitive [39, 46, 121]. It is called cooperative when the involved agents cooperate together to reach a common goal. It is possible to achieve a solution that satisfies all the agents when they acquire more and more knowledge about their interests and the interests of the others, learn to overcome their rigid and emotional attitude and adjust their proposals taking into account the needs of the others. Consequently, these approaches aim at maximizing the global satisfaction. A negotiation is called competitive when the involved agents are self-interested. They focus only on their own interests and try to maximize their local utility.

Interactions in Multi-Agent Systems

Negotiation is a form of interaction between agents in order to collaborate and cooperate with each other. Consequently, the negotiation protocol requires a language which supports such communication.

In multi-agent systems two main agent communication languages have been proposed: the *Knowledge Query and Manipulation Language (KQML)* [52] and the *Foundation for Intelligent Physical Agents, Agent Communication Language (ACL)* [53]. KQML and FIPA-ACL are very similar in their basic concepts and principles. In this thesis we used FIPA-ACL. An ACL message includes the following attributes:

- *sender*: represents the sender of the message;
- *receiver*: represents the receiver of the message;
- *content*: contains the information which the sender intends to send;

- *language*: highlights the language in which the content is represented
- *ontology*: represents the name of the ontology used to give a meaning to the terms used in the content.
- *performative* (also called communicative act): involves 22 locutions and represents the intention of the message's sender.

In relation to the last point, the 22 locutions are divided into 5 groups:

- information exchange such as *inform* (informing another agent about a fact of the world);
- request for information such as *query-if* (asking another agent whether a given proposal is valid);
- negotiation such as *propose* (making a proposal to realize a certain action);
- action execution such as *agree* (agreeing to perform some action);
- error handling such as *failure* (telling another agent that an action was attempted but failed).

Example 2.3.1 *Agent A wants to inform agent B that she has classified New York in the category good: (inform :sender A :receiver B :content (New York, Good) :ontology classification)*

2.3.4 Negotiation strategies

Negotiation strategies represent the internal reasoning of the agents. In other words, how the agents define their proposals or accept/reject the proposals of other agents. Two main negotiation strategies could be adopted by agents to decide which proposal to choose in the next round: Concession and trade-off. With concession strategies, each agent gradually reduces the utility of her proposal until there is no more conflict [150]. With tradeoff strategies, instead, agent is seeking to find a

proposal that has the same utility as her previous proposal and increase the utility of her opponent's proposal [44].

Moreover, time can be a determining factor in deciding the next proposal to be made by influencing the agent's behavior. In fact, negotiation processes are constrained by a deadline to be respected by all participants. The acceptance of a proposal by an agent may be influenced by the deadline. This can be defined based on several factors such as the discounting factor [23], time-dependant threshold [43], or anytime strategy [15]. For the discounting factor, the acceptance of the proposal by an agent depends on its utility function deducted from the discounting factor. For the time-dependant threshold, the acceptance of the proposal depends on the evaluation of the proposal compared to this threshold. For anytime strategy, an acceptable proposal should be provided rapidly but continues to improve progressively over time.

To study negotiation strategies, we distinguish three approaches: game theory, heuristics and argumentation approaches.

- **Game Theory:** game theory is a branch of micro-economics [106, 108] aiming at analyzing the strategic interaction between rational agents. The interaction between the agents is represented as a game where each player (agent) performs actions that maximize her gain considering the actions of all the other players. A crucial concept in game theory is the *Nach equilibrium* where each agent relies on her expectations about the actions of other agents to make the best decision.

Game theory is widely used in automated negotiation [25, 39, 48, 88, 116, 149]. It allows to analyze the rational actions carried out by self-interested agents during a negotiation in order to provide optimal and stable solutions.

Application of game theory in different automated negotiation contexts may raise the following problems. On the one hand, each agent's decision depends on their beliefs about the possible actions of other agents. The evaluation of all these actions may require more time and computation than can be afforded.

On the other hand, participants are supposed to be also fully informed about the preferences of others [79]. However, one relevant limitation of game theory is the unbounded rationality of agents. Thus, they always make decisions that optimize their utility according to their belief about their opponent's "optimal" strategies. However, in practice, the assumption that agents must always make rational choices may not be applicable to practical situations where agents have limited computing resources at their disposal or may behave in an altruistic manner.

- **Heuristics:** such approaches solve the problems raised by game theory approach. They mainly handle the unbounded rationality of agents. Hence, agents make "good" but not necessarily "optimal" choices [79]. Each agent makes a decision according to her own strategy without needing to consider all the actions of other agents. Consequently, these approaches aim at deriving a sub-optimal solution and the evaluation of these solutions is based on experimentation through simulations. They are less constraining than game theory approaches. They provide more flexibility to agents and also allow the consideration of more aspects involved in decision making. As part of the heuristic approach, the decision functions introduced by Faratin et al. [43] are fundamental. In fact, the generation of the offer or counter-offer is linear combination of simple decision functions called tactic. Such a tactic calculates the amount of the concession for a single attribute of an offer according to a given parameter such as time, remaining resources or the behavior of the opponent. Therefore, three types of tactics are suggested for each parameter:
 - Time-dependant tactic: identifies when an agent should concede;
 - Behavior-dependant tactic: determines the behaviour of an agent by imitating the behaviour of her opponent;
 - Ressource-dependant tactic: generates an offer according to the available resources.

According to the concession function of time-dependant tactic, [43] which can be polynomial or exponential, an agent concedes more or less quickly when

the deadline approaches. Thus, if the concession function is polynomial, the agent concedes more quickly at the beginning of the negotiation than when the concession function is exponential. As a result, we can distinguish two agent behaviors. *Boulware* agent who makes small concessions at the beginning of the negotiation and large concessions when negotiation deadline approaches and *conceder* agent who concedes rapidly from the beginning.

A concession considering remaining resources is similar to a concession considering time when its quantity decreases along the process. Hence, agents are expected to gradually make concessions to reduce the remaining resources.

- **Argumentation:** such approach allows agents to exchange proposals while providing additional information justifying their proposals or supporting an offer [90, 91]. When an agent rejects a proposal, she can send a criticism explaining the reason for her rejection. By exchanging arguments, agents can influence each other and revise their opinion about their decisions. Interaction based on arguments leads to a consistent conclusion [99] but generates a considerable communication cost. Therefore, the negotiator has to choose the best argument in response to the opponent's proposal.

As in this thesis we aim to derive a group classification while taking into consideration dynamic external factors that may influence the agent's behavior, we adopt multilateral negotiation based on heuristic approach.

2.4 Synthesis

Aggregation methods of agents' preferences are widely studied in computational social choice theory in order to reach a collective decision. They are as well used in group recommender systems to derive group recommendation. Within these methods, there are several paradoxes such as Arrow's impossibility theorem stating that there is no voting system that satisfies several basic axioms at once. Moreover, these methods are applied only for the purpose of deriving a ranking of all alternatives.

In the literature, few works deal with group ordinal classification problems. Such works classify alternatives either based on a common preference relation system derived from agent's preferences or by using an aggregation/disaggregation approach that allows to learn the group preference and to classify the remaining alternatives. However, the proposed approaches do not deal with the influence of the environment's dynamicity on the agent's behavior such as the influence of time pressure on the group decision.

In this thesis, we adopt a negotiation process to derive a group classification. We use multi-agent theory to address the issues of communication and collaboration that are fundamental to any decision making system. Agents are also able to learn from their past experience and thus improve their proposal by revising their strategies. Thus, the behavior of the agents can change according to the environment in which they are involved.

The proposed negotiation process is based on a heuristic and cooperative approach. The heuristic approach was chosen for its adaptability to a dynamic and distributed environment. We considered that all agents are cooperative and share the common goal of reaching an agreement. Moreover, we consider that the space of possible solutions is not known by all the agents. Each agent has a knowledge of her own proposals and does not perceive those of the others. For this aim, we have favored a multilateral negotiation (one-to-many) conducted by a mediator. The choice of a mediator allows on the one hand to limit the scope of interaction between agents in order to reduce the complexity of their reasoning, to facilitate the search for agreement and to avoid the disclosure of their private information, on the other hand to ensure that an agreement is reached at the end of the negotiation process in order to guarantee its success.

Chapter 3

Pre-negotiation process: Interactive Elicitation of an ordinal classification model

Contents

3.1	Introduction	88
3.2	MR-Sort example	89
3.3	Learning a MR-Sort model from assignments	90
3.3.1	Possible Categories	91
3.3.2	Maximum margin optimization	92
3.4	Incremental MR-Sort	94
3.4.1	Main Framework	94
3.4.2	Question selection strategies	96
3.5	Experiments	99
3.6	Conclusions	104

This chapter was published in the International Conference on Algorithmic Decision Theory (ADT 2019) [105]

3.1 Introduction

As we presented in the previous chapter, there are several situations where it is necessary to classify objects, defined on several criteria, into ordered classes (for example, credit ratings, evaluating students, hotel categorization, etc). In this chapter, we consider the problem of eliciting a model for ordered classification for a single decision maker. In particular, we consider Majority Rule Sorting (MR-Sort).

As presented in the section 1.2.3, the remarkable characteristic of the MR-Sort procedure is that its classifications are readily explainable to the users. In MR-Sort an alternative is assigned to a category when it is “at least as good as” an idealized profile describing the category’s lower “limit” and “not better” than the category’s upper limit. The limit profiles encode the characteristics that an alternative should have in order to be assigned to a particular class according to the decision maker (for example: a 4-stars hotel should have rooms of at least a certain size, it should have a swimming pool, a 5-star hotel should have a classy restaurant, etc). The relation comparing alternatives and limit profiles is based on a weighted majority rule (in Sobrie et al. [131] a non linear model for combining the weights is considered).

Recall that in order to use MR-Sort in practice, it is necessary to undertake a preference elicitation phase in order to assess the parameters of the model. The “classical” approach to elicitation consists in a fine grained assessment of the model parameters [30, 31, 92, 130]; however it is often not reasonable to directly ask the decision maker about the values of the parameters of the model. The novelty of this work is adopting an incremental elicitation approach in order to learn the parameters of MR-Sort. For this aim, we assume that the decision maker can easily classify one given alternative into one of the categories; the information provided by the user can be used to update the model and provide a better result. The intuition is that, by selectively choosing which items the decision maker could classify, we could retrieve a good approximation of the “correct” model of the decision maker (DM) even with a small number of examples. We also stress that the information provided by the user may be noisy, as she may have made a mistake in assessing

the right category for the alternative.

To do so, we adopt max-margin optimization, whose main principle is the following. The currently known assignments of items to categories are encoded by a set of inequalities on the feasible parameters; a shared non-negative margin is introduced as a decision variable that is maximized in the objective function. Noisy feedback is addressed by relaxing the constraints using slack variables and adding a penalty term in the objective function for violated constraints.

In this chapter, we propose an incremental approach for eliciting the parameters of a MR-Sort model; at each step of the elicitation procedure, the system asks a question to the user and the question is specifically chosen in order to be as “informative” as possible. Our main algorithm repeatedly uses the max-margin optimization routine with two goals 1) to make an estimation of the model parameters given the current information 2) to determine a measure of uncertainty associated with a potential question. For the second goal, we consider different strategies based on computing the maximum-margin a posteriori (considering the different answers that the decision maker could give).

This chapter is structured as follows. After providing an example of MR-Sort background in Section 3.2, we present MR-Sort model in Section 3.3 and the interactive elicitation in Section 3.4. We then discuss our experimental results in Section 3.5 and provide some final comments and directions for future work in Section 4.9.

3.2 MR-Sort example

We refer to section 1.2.3 for more details about MR-Sort. We now provide a small example illustrating the method.

Example 3.2.1 *Assume that we have 5 hotels (x, y, z, t, u) defined on $n = 3$ criteria (cost, comfort, quality of the restaurant). Our aim is to assign our hotels into $p = 3$ categories (C_1 : 1 star, C_2 : 2 stars, C_3 : 3 stars). The MR-Sort model parameters*

	<i>criterion</i> ₁	<i>criterion</i> ₂	<i>criterion</i> ₃
b^1	0	0	0
b^2	2	2	4
b^3	4	4	8
b^4	5	5	10
x	1	2	3
y	3	3	8
z	1	5	8
t	1	3	10
u	1	2	1

Table 3.1 – Performance of limit profiles b^1, b^2, b^3, b^4 and alternatives x, y, z, t, u

$b^1 b^2 b^3$	b^4	Assignment
$x \succeq \not\succeq \not\succeq$	$\not\succeq$	C_1
$y \succeq \succeq \not\succeq$	$\not\succeq$	C_2
$z \succeq \succeq \succeq$	$\not\succeq$	C_3
$t \succeq \succeq \not\succeq$	$\not\succeq$	C_2
$u \succeq \not\succeq \not\succeq$	$\not\succeq$	C_1

Table 3.2 – Comparison between alternatives and profiles and final assignments.

are: the majority threshold $\lambda = 0.55$, a vector of weights $w = (0.2, 0.5, 0.3)$. The criteria evaluation scale is between 0 and 5 (5 being the best score) for *criterion*₁ and *criterion*₂; the scale for *criterion*₃ is between 0 and 10. Table 3.1 presents the performances of the limit profiles and the alternatives; Table 3.2 their comparisons using $\succeq$ and the assignments of alternatives.

Note that the MR-Sort method allows the use of heterogeneous scales since the only information being useful for the relation $\succeq$ is an ordinal one. Hence, a problem with only ordinal scales or with different types of scales (interval one, ratio one and ordinal one) can be handled without difficulty by an MR-Sort model.

3.3 Learning a MR-Sort model from assignments

We assume that we are given some assignments of alternatives to categories. In this work we assume that limit profiles are given. In the following, we first introduce the

notion of possible and necessary categories, and then we present a maximum-margin optimization for learning the parameters of a MR-Sort model.

3.3.1 Possible Categories

We recall that each alternative is assessed under a set of n criteria $\mathcal{C} = \{cr_1, \dots, cr_n\}$, the evaluation of x is a vector $(x_1, x_2, \dots, x_n)$ and the set of categories is denoted by $\mathcal{C} = \{C_1, C_2, \dots, C_p\}$.

Let $\mathcal{A}$ be the set of alternatives whose category is known and LS the “learning set” (pairs of alternatives and categories): $LS = \{(x, C_h), x \in \mathcal{A}, x \rightarrow C_h\}$. From LS, assuming an underlying MR-Sort model, we can reason about the weights that are consistent with the current knowledge. According to Equation (1.1) and (1.2), an assignment of the type $x \xrightarrow{DM} C_h$ (made by the decision maker) corresponds to imposing the linear constraints:

$$x \succeq b^h \iff \sum_{i: x_i \geq b_i^h} w_i \geq \lambda \quad (3.1)$$

$$x \not\succeq b^{h+1} \iff \sum_{i: x_i \geq b_i^{h+1}} w_i < \lambda \quad (3.2)$$

Let $\Theta(LS)$ be set of parameters that are compatible with the learning set LS, satisfying constraints (3.1) and (3.2) for all pairs in LS, and the requirements that the weights are non negative and normalized. When dealing with partially specified preference models, it is typical to reason about possible and necessary preference information [66]. In our context, given an alternative and its performance evaluation we can reason about the parameters that are consistent with the current knowledge. Now, given an item $x \notin \mathcal{A}$, we define the set of possible categories $PC(x; LS)$ as the set of categories such that there is an instantiation of the parameters $\theta = \{w_1, \dots, w_n, \lambda\}$ consistent with the assignment *given the previously known assignments* LS.

$$PC(x; LS) = \{C_i \in \mathcal{C} \mid \exists \theta \in \Theta(LS) : x \xrightarrow{\theta} C_i\}$$

where we write $x \xrightarrow{\theta} C_i$ to emphasize the dependency between the parameters θ and the assignment. If there is only one possible category, i.e. $|\text{PC}(x)| = 1$, then it means that the alternative has to be *necessarily* assigned to the only category in PC (assuming that the model is consistent with the learning set; in Section 3.3.2 we discuss how to handle inconsistencies). In practice, it is possible that a partial knowledge about the parameters allows us to determine in which categories we have to place several alternatives. We anticipate that the concept of possible categories plays an important role in our elicitation strategies (see Section 3.4.2).

3.3.2 Maximum margin optimization

We now address the problem of learning the parameters $\theta = \{w_1, \dots, w_n, \lambda\}$ of a MR-Sort model (given a set of learning assignments) with a linear program. We assume that we are given as input the following values: the number of criteria n , the number of categories p , and the limit profiles b^h for each $h \in [p]$. Our goal is to assess the weights w and the majority threshold λ , that are decision variables for the optimization problem. In order to discriminate between different choices for the parameters, the first step is to introduce a shared margin μ as an additional decision variable. The first linear program (LP) assumes that the data in LS is perfectly consistent with the MR-Sort model.

$$\mu^* = \max \quad \mu \tag{3.3}$$

$$\text{s.t.} \quad \sum_{i: x_i \geq b_i^h} w_i \geq \lambda + \mu \quad \forall (x, C_h) \in \text{LS} \tag{3.4}$$

$$\sum_{i: x_i \geq b_i^{h+1}} w_i \leq \lambda - \mu \quad \forall (x, C_h) \in \text{LS} \tag{3.5}$$

$$\sum_{i=1}^n w_i = 1 \tag{3.6}$$

$$\mu \in \mathbb{R} \tag{3.7}$$

$$w_i \geq 0 \quad \forall i \in [n] \tag{3.8}$$

Constraints (3.4) and (3.5) correspond to Equations (3.1) and (3.2) with the inclusion of the shared margin μ . Note that later in Section 3.4.2 we will use $\mu^*(\text{LS})$

to denote the application of the LP above to find the value of the best compatible margin.

We observe that in general there may be occasional inconsistencies in user feedback, or simply the data may not be compatible with a MR-Sort model. The “typical” way to handle this is to introduce slack variables (whose sum we aim at minimizing); this idea goes back to the UTA approach [77] and several later models [2, 55]. We formalize the problem of finding the weights of a MR-Sort model (given a set of learning assignments) with the following linear program.

$$m^* = \max \quad \mu - \alpha \sum_{(x, C_h) \in \text{LS}} \xi_{i,j} \quad (3.9)$$

$$\text{s.t.} \quad \sum_{i: x_i \geq b_i^h} w_i + \xi_{x,h} \geq \lambda + \mu \quad \forall (x, C_h) \in \text{LS} \quad (3.10)$$

$$\sum_{i: x_i \geq b_i^{h+1}} w_i - \xi_{x,h} \leq \lambda - \mu \quad \forall (x, C_h) \in \text{LS} \quad (3.11)$$

$$\sum_{i=1}^n w_i = 1 \quad (3.12)$$

$$\mu \in \mathbb{R} \quad (3.13)$$

$$\xi_{x,h} \geq 0 \quad \forall (x, C_h) \in \text{LS} \quad (3.14)$$

$$w_i \geq 0 \quad \forall i \in [n] \quad (3.15)$$

The variables are the following: $\lambda \in [0, 1]$, $w_i \in [0, 1] \forall i \in [n]$, $\mu \in [0, 1]$ and $\xi_{x,h} \in [0, 1] \forall x \in \text{LS}$ such that x assigned to C_h . We use a parameter α to express the “cost” of violating a constraint representing assignments from the learning set. The objective (Equation 3.9) is to maximize the shared margin μ (that we want to maximize) minus the sum of the constraints violations $\xi_{i,j}$ (that we want to minimize). In order to be able to handle the inconsistency in the user’s answers we add a slack variable $(\xi_{x,h})$ for each $x \in \mathcal{A}$ and for each $h \in \{1, \dots, p\}$. Constraint 3.10 and constraint 3.11 handle the conditions related to the assignments of the alternatives in the learning set LS. Constraint 3.12 enforces that the weights are normalized.

3.4 Incremental MR-Sort

Algorithm 2 : The MR-Sort-Inc algorithm. K is the number of questions, X the set of alternatives, C the set of categories, LS ($LS = \{(x, C_h), x \in \mathcal{A}, x \rightarrow C_h\}$) learning assignments, s the strategy.

```

1 Procedure MR-Sort-Inc( $K, X, C, LS, s$ ):
2   for  $j \leftarrow 1, \dots, K$  do
3     Compute  $m^*(LS)$  and associated  $w^*$  and  $\lambda^*$  with linear program
4     Classify items in  $X \setminus \mathcal{A}$  using  $w^*$  and  $\lambda^*$ 
5     Compute classification error          /* This is only possible in
      simulations */
6     for  $x \in X \setminus \mathcal{A}$  do
7       for  $C_i \in C$  do
8          $LS' \leftarrow LS \cup \{x \rightarrow C_i\}$  /* Include additional assignment
          */
9          $v_i(x) \leftarrow m^*(LS')$  /* Compute margin a posteriori */
10      end
11       $S_s(x) \leftarrow$  aggregate  $v(x)$  according to strategy  $s$ 
12    end
13     $x^* \leftarrow \arg \max_x (S_s(x))$  /* Decide what to ask next */
14     $C_{x^*} \leftarrow \text{Answer}(x^* \rightarrow ?)$  /* Ask question to DM */
15     $LS \leftarrow LS \cup \{(x^*, C_{x^*})\}$ 
16  end
17  return  $w^*, \lambda^*$ 

```

In this section, we provide an elicitation method for an ordinal classification problem assuming that the preferences can be modeled by MR-Sort.

3.4.1 Main Framework

Our elicitation procedure starts with a small learning set LS to which we add one-by-one new assignment examples. We ask questions of the type “*In which category should x be assigned to?*”; it is crucial to select “informative” items to ask about, in order to quickly converge (in few interaction cycles) to a good classification model.

We remind that a MR-Sort model is defined by the following parameters: the weights $(w_1, \dots, w_n)$, the limit profiles $(b^1, \dots, b^{p+1})$ and majority threshold λ . We fix the limit profiles before the beginning of the elicitation. In the experiments below the limit profiles are chosen in a way to evenly partition the criteria scale.

The main steps of our approach are presented in Algorithm 2. Briefly, we start with some data from the learning set LS (learning alternatives $\mathcal{A}$ and their classification) and we include them as constraints in our linear program (LP). Given the learning set, we can estimate the weights w^* and λ^* using the linear program; in simulations we can also evaluate the classification error. After that, we test the assignment of each unassigned item to each category and retrieve the margin values obtained when using the optimization routine with each of the additional assignments. This gives us a vector of margins “a posteriori”, that are aggregated differently depending on the “strategy” (different strategies will be discussed in the following section), giving a score for each item. The item having the highest score is chosen: the user is asked about the assignment of that item and the learning set is augmented. This procedure continues until a stopping condition (in our experiments when we reach a fixed number of questions, but in real applications termination may be left to the user). The last linear program contains all the assignments of the computed learning set, hence it provides the weights that we are looking for.

Example 3.4.1 *We apply our incremental elicitation algorithm to the running example. Suppose that we only know the assignment of alternative x ($\text{LS} = \{(x, C_1)\}$) and we want to ask just one question to the user. We first set the limit profiles: $b^1 = (0, 0, 0)$, $b^2 = (\frac{5}{3}, \frac{5}{3}, \frac{10}{3})$, $b^3 = (\frac{10}{3}, \frac{10}{3}, \frac{20}{3})$, $b^4 = (5, 5, 10)$. Our goal is to learn λ and a weight vector w' which will assign the remaining alternatives to categories as close as possible to the ones presented in Example 3.2.1. For this, according to our algorithm, we compute the score of alternatives y, z, t, u ($S_s(y), S_s(z), S_s(t), S_s(u)$) and ask the assignment of the alternative having the highest score. We add this new constraint to LS and find the weight vector w^* corresponding to the largest margin. Using w^* , we can find the current best assignment of the remaining alternatives and compute ¹ an error measure using the assignments of Example 1 and the one that we find using w^* .*

¹This step is of course only to be performed in simulations, in real use of the procedure the classification error will not be known

3.4.2 Question selection strategies

In the following we present several strategies to select the next question to ask to the user. These strategies are used within our interactive elicitation paradigm described in Algorithm 2. The proposed strategies makes use of the max-margin optimization programs (discussed above in Subsection 3.3.2) to identify informative questions.

Most Uncertain

With this strategy, we aim to ask the DM to classify the *most uncertain* item in $X \setminus \mathcal{A}$, that is the item that is compatible (according to the constraints derived from the known assignments of the learning set) with the highest number of categories. The score that this strategy assigns to an alternative x is given by the cardinality of the set of possible categories: $S_{\text{MU}}(x) = |\text{PC}(x; \text{LS})|$.

The computation of the set $\text{PC}(x)$ for an item x makes use of the linear program for learning the parameters of a MR-Sort without the slack variables (Equations 3.3-3.8). For each category C_i , we then constrain the item x to be assigned to C_i and simply check whether the margin is non negative; indeed $\mu^*(\text{LS} \cup (x, C_i)) > 0$ if and only if C_i is a possible category ($C_i \in \text{PC}(x; \text{LS})$). The score of this strategy can be compactly written as:

$$S_{\text{MU}}(x) = |\text{PC}(x; \text{LS})| = \sum_{i=1}^p H[\mu^*(\text{LS} \cup (x, C_i))]$$

where $H(\cdot)$ is the step function.

Example 3.4.2 Consider the problem presented in Example 3.2.1. We suppose that we know the performance of each alternative, and the assignment of x , while the other assignments are unknown. Assume that we want to ask just one question. As before, we fix $b^1 = (0, 0, 0)$, $b^2 = (\frac{5}{3}, \frac{5}{3}, \frac{10}{3})$, $b^3 = (\frac{10}{3}, \frac{10}{3}, \frac{20}{3})$, $b^4 = (5, 5, 10)$. Table 3.3 presents the S_{MU} of all the remaining alternatives (step 3).

Alternatives	PC(x ; LS)	S _{MU}
y	$\{C_2, C_3\}$	2
z	$\{C_1, C_3\}$	2
t	$\{C_1, C_2, C_3\}$	3
u	$\{C_1\}$	1

Table 3.3 – Number of possible categories for the example of hotel categorization.

As a result we ask the assignment of alternative t to the user. The user will answer C_2 to this question (coherent with Example 3.2.1) and the constraint related to $t \rightarrow C_2$ will be added to the LP. After the inclusion of the constraint $t \rightarrow C_2$ in LS, we find our final weight vector w^* , such as $w^* = (0.266, 0.366, 0.366)$. Even if w' is different from the weight vector of Example 3.2.1 ($w = (0.2, 0.5, 0.3)$), after our incremental elicitation we find the same assignments for y, z, u (i.e. $y \rightarrow C_2, z \rightarrow C_3$ and $u \rightarrow C_1$).

Sum-of-margin strategy

The problem of the most-uncertain strategy is that, roughly speaking, it is agnostic to whether a potential assignment is consistent with large portions of the parameter space or just with small area. The intuition of sum-of-margin is to use the value of the objective function of the LP as a surrogate measure of the “degree” of satisfaction of an assignment $x \rightarrow C_h$. Intuitively, we should ask about items that may fit well into several different categories. We also include the possibility that some constraints can be violated, therefore make use of the second LP (Equations 3.9-3.15), with the penalty variables ξ for violated assignments of the learning set.

Considering an alternative $x \in X \setminus \mathcal{A}$, we evaluate the penalized margin m^* adding the assignment $x \rightarrow C_i$ for all $i \in [p]$ and construct the following vector:

$$v(x) = (m^*(\text{LS} \cup (x, C_1)), \dots, m^*(\text{LS} \cup (x, C_p))).$$

In order to aggregate this vector into a single numerical measure, we adopt the sum.

Assignment	$v_i(\cdot)$	S_Σ	Assignment	v'	S_E
$y \rightarrow C_1$	-0.45	} 0.45	$y \rightarrow C_1$	0	} 0.30
$y \rightarrow C_2$	0.45		$y \rightarrow C_2$	0.5	
$y \rightarrow C_3$	0.45		$y \rightarrow C_3$	0.5	
$z \rightarrow C_1$	0.45	} 0.90	$z \rightarrow C_1$	0.5	} 0.30
$z \rightarrow C_2$	0		$z \rightarrow C_2$	0	
$z \rightarrow C_3$	0.45		$z \rightarrow C_3$	0.5	
$t \rightarrow C_1$	0.45	} 1.08	$t \rightarrow C_1$	0.43	} 0.44
$t \rightarrow C_2$	0.18		$t \rightarrow C_2$	0.15	
$t \rightarrow C_3$	0.45		$t \rightarrow C_3$	0.42	
$u \rightarrow C_1$	0.49	} -0.10	$u \rightarrow C_1$	1	} 0.00
$u \rightarrow C_2$	0		$u \rightarrow C_2$	0	
$u \rightarrow C_3$	-0.55		$u \rightarrow C_3$	0	

Table 3.4 – The score obtained by the different items in the hotel categorization example using the heuristics sum-of-margin (left) and entropy (right)

Hence, the score of the alternative x is computed as

$$S_\Sigma(x) = \sum_{i=1}^p v_i(x) = \sum_{i=1}^p m^*(LS \cup (x, C_i)).$$

Example 3.4.3 Consider again the running example. Table 3.4 (in the left) presents the S_Σ of alternatives $\{y, z, t, u\}$. As the example shows, the best question is to ask about t , the second best question is to ask about z , then y , and finally u . We note a disagreement of MU and sum-of-margin about the ranking of z and y .

Entropy

This strategy adopts the notion of entropy to assess the uncertainty for a given alternative. As the strategy sum-of-margin, we calculate m^* which represents the maximum penalized margin if we assign the alternative x to the category C_i , for all $i \in [p]$. We then combine these values in an evaluation vector v' that filters out negative values (assigning them a zero value) and we normalize so that the values sum up to one:

$$v'_i(x) = \frac{R(m^*(LS \cup (x, C_i)))}{\sum_{j=1}^p R(m^*(LS \cup (x, C_j)))}$$

where $R(\cdot)$ is the ramp function. We then apply the entropy to calculate the final score of the alternative.

$$S_E(x) = - \sum_{i \in [p]: v'_i(x) > 0} v'_i(x) \log(v'_i(x))$$

Intuitively this method should favour asking about items whose uncertainty is the greatest in an information-theoretic sense, considering the vector v' (composed of normalized non-negative margin values) as a surrogate measure for the probability distribution of the category of x (i.e. how likely is x to be in each of the categories).

Example 3.4.4 *We consider again our example on hotel categorization. We now apply the entropy strategy; Table 3.4 (on the right) presents the score S_E of all the alternatives. Item t has the highest score, then z and y are tied in the second position; u is last.*

Random strategy

As a baseline, we consider the random selection of an object x from $X \setminus \mathcal{A}$. Utilizing this strategy is equivalent to use a non incremental elicitation procedure with $|LS| + K$ (LS being the initial learning set of Algorithm 2 and K being the number of questions) alternatives in the learning set.

Note that all our strategies have the same computational complexity : $O(Kmp)$ (resp. number of questions, alternatives, categories).

3.5 Experiments

We implemented our incremental MR-Sort algorithm using Java API of CPLEX for solving the LP.² We performed simulations aimed at evaluating the effectiveness

²All experiments were run on a 2.9 GHz Intel, Core i7 and 16 Giga of RAM.

of the proposed elicitation strategies with two data-sets: a synthetic data-set fully represented by a MR-Sort model, and a data-set from the UCI machine learning repository³.

Synthetic data-set For the first part of our experiments, the input data was generated randomly using a uniform distribution on the space of evaluations.

We include only one assignment, chosen at random in LS. The steps of the simulation are as follows:

- i. Using a uniform distribution for each of the parameters, we generate the performance table of alternatives, a vector of weight w , the majority threshold λ . The limit profiles are chosen to evenly partition the range of evaluation values. These parameters fully specify the decision maker; we call this model $\overline{M}$. We then apply the MR-Sort method in order to find the assignments of the generated alternatives; these assignments constitute the “ground truth”.
- ii. We apply our incremental elicitation Algorithm 2 using the assignments of some alternatives of $\overline{M}$ as learning assignments. We apply incremental MR-Sort by asking questions to the user based on the proposed strategies, simulating a decision maker who answers according to the results of the $\overline{M}$. We generate as output a vector of weight w^* representing the DM’s preferences.
- iii. Using the assumed limit profiles and the learned weight vector w^* and the majority threshold λ^* , we obtain our learned model M that provides the assignments of the remaining alternatives.
- iv. At the end we calculate an error rate based on the difference between the assignments of the true model $\overline{M}$ and the learned model M .

We iterate these steps 50 times and we evaluate the classifications obtained with the different methods according to the average classification error $AE_k = \sum_{x \in X} \text{err}_k(x)/m$ where m is the number of alternatives and $k \in \{1, 2\}$. The value

³These data sets are available at <https://archive.ics.uci.edu/ml/datasets.html>

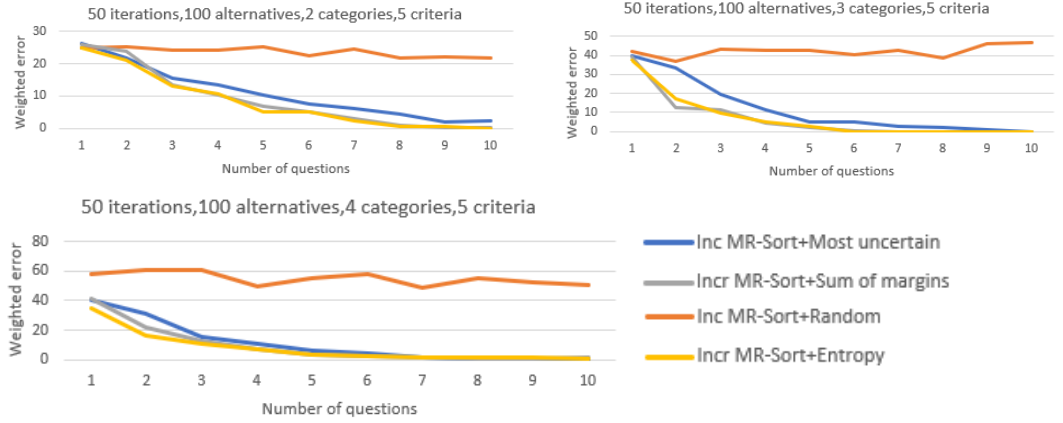


Figure 3.1 – AE_2 based on number of questions with $m = 100$, $n = 5$, $p = 2$, $p = 3$, $p = 4$, $|LS| = 1$ and 50 iterations, where theoretical bounds are $\min(\text{weighted error}) = 0$ and $\max(\text{weighted error}) = 100$

$\text{err}_k(x) = d(C_i, C_j)$ is the magnitude of error when $x \xrightarrow{\overline{M}} C_i$ and $x \xrightarrow{M} C_j$ (the true model assigns x to C_i while the learned model M assigns it to C_j). err_1 adopts a 0/1 loss as distance, while err_2 considers the displacement between the assignments, that is $d(C_i, C_j) = |i - j|$.

Figures 3.1, 3.4, 3.5 and 3.6 present the results of our simulations. Note that the random strategy performs very poorly, while all three strategies (maximum uncertainty, sum of margin, entropy) have reasonably good performance. Not surprisingly, the higher the number of categories, the quicker we converge, since there is more uncertainty with more categories. Conversely, the higher the number of criteria the less quickly the interaction converges. The results with respect to AE_2 are quite satisfying since they show that, after few questions have been answered, our procedure makes few assignment mistakes and these mistakes concern consecutive categories most of the times.

UCI Car We performed simulations based on Car data-set⁴. There are 1728 items defined on 6 categorical attributes (that we interpret as ordinal); the items are partitioned in 4 categories: unacceptable (65 items), acceptable (1210), good (384) and very good (69). We converted the qualitative (ordinal) attributes to

⁴<https://archive.ics.uci.edu/ml/datasets/car+evaluation>

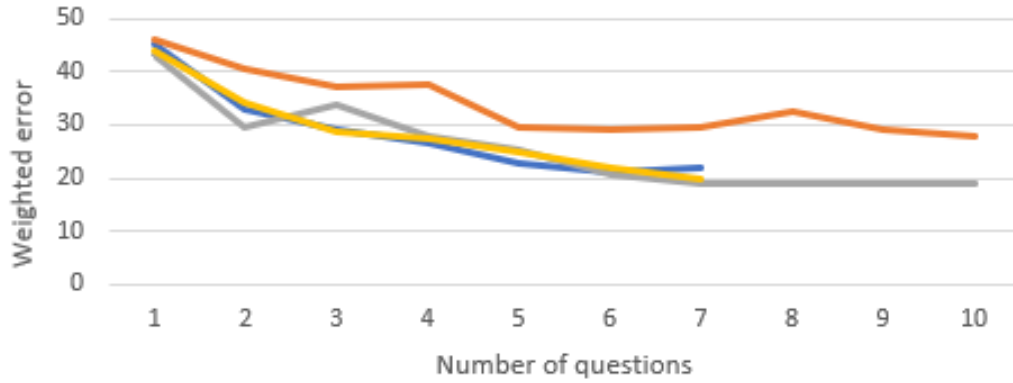


Figure 3.2 – AE_1 based on number of questions with $m = 300$, $n = 6$, $p = 4$, $|LS|=1$, $\lambda = 0.7$ and 50 iterations, where theoretical bounds are $\min(\text{weighted error}) = 0$ and $\max(\text{weighted error}) = 100$

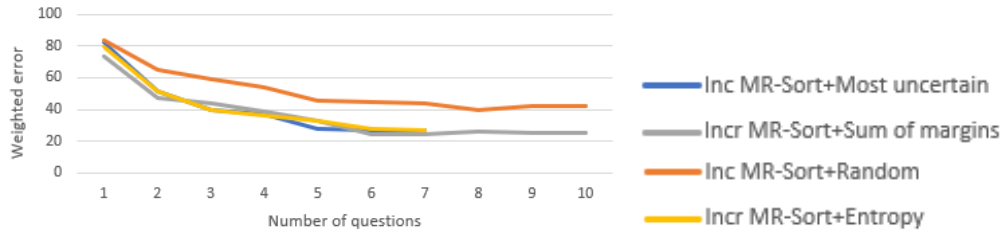


Figure 3.3 – AE_2 based on number of questions with $m = 300$, $n = 6$, $p = 4$, $|LS|=1$, $\lambda = 0.7$ and 50 iterations, where theoretical bounds are $\min(\text{weighted error}) = 0$ and $\max(\text{weighted error}) = 100$

numeric attributes and we set the criteria rating scale between 1 and 4. The steps of the simulation are : i.) We first compute the best error rate that we could have if all the data was included in LS . ii.) We ask questions until we reach this error percentage. We started to test on 300 alternatives chosen randomly and we start by including in the learning set only one assignment taken randomly.

In order to find the best error percentage, we call it P^* , we include all the data in LS and we vary the value of λ . We obtain P^* when λ is between 0.7 and 0.75. In this case the $err_1 = 18.86\%$ (err_2 being 0.24). So we aim to reach this percentage by asking the minimum number of questions. Our simulations showed that fixing the value of λ makes faster the convergence, we show in Figures 3.2- 3.3 the comparison between the 4 strategies where λ is fixed to 0.7.

With the random strategy we observe that the variation of error according to number of questions is not monotonous and we don't approach to P^* by asking

10 questions. On the other hand, the strategies maximum uncertainty and sum of margin are closer to P^* starting from 6th question. The strategy maximum uncertainty stops after the 7th question because it hangs in case of inconsistency. We notice that the assignments errors are almost always between consecutive categories; Figure 3.3 shows the performance with respect to the second metric.

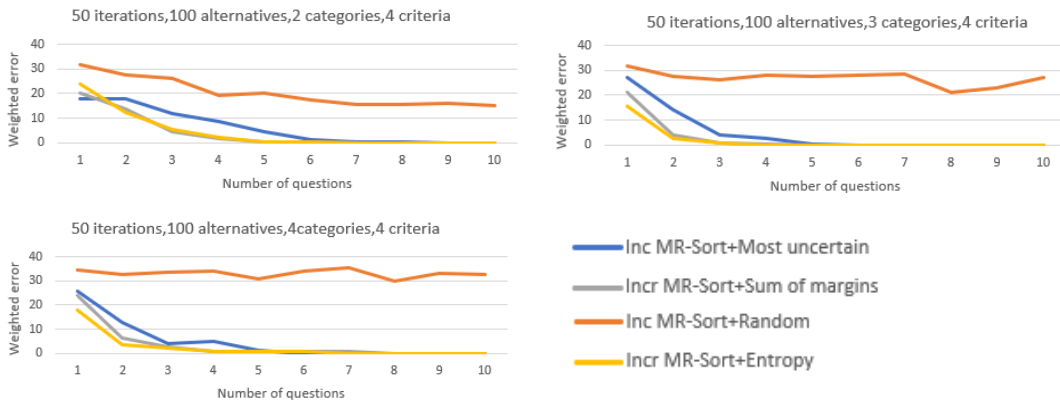


Figure 3.4 – Classification error AE_1 based on number of questions with $m = 100$, $n = 4$, $p = 2$, $p = 3$, $p = 4$, $|LS| = 1$ and 50 iterations, where theoretical bounds are $\min(\text{weighted error}) = 0$ and $\max(\text{weighted error}) = 100$



Figure 3.5 – AE_2 based on number of questions with $m = 100$, $n = 4$, $p = 2$, $p = 3$, $p = 4$, $|LS| = 1$ and 50 iterations, where theoretical bounds are $\min(\text{weighted error}) = 0$ and $\max(\text{weighted error}) = 100$

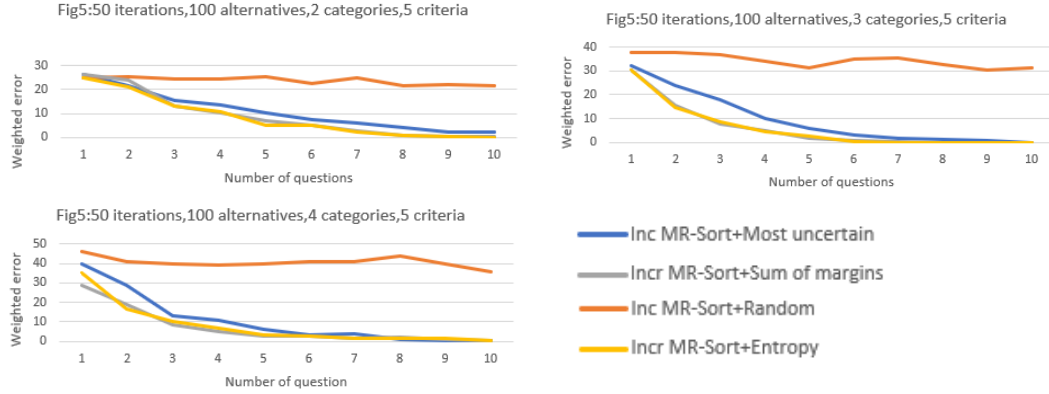


Figure 3.6 – AE_1 based on number of questions with $m = 100$, $n = 5$, $p = 2$, $p = 3$, $p = 4$, $|LS| = 1$ and 50 iterations, where theoretical bounds are $\min(\text{weighted error}) = 0$ and $\max(\text{weighted error}) = 100$

3.6 Conclusions

MR-Sort is an attractive method for ordinal classification that offers the advantage of allowing the use of heterogeneous scales (mixing ordinal and cardinal scales). In this chapter we presented an incremental elicitation procedure for the parameters (weights and threshold) of MR-Sort. Our approach relies on a maximum-margin optimization that aims at satisfy as well as possible the known assignments, following approaches proposed in the literature for non-interactive learning of MR-Sort and variations [93, 131, 132]. The novelty of this method consists in showing how the optimization can be used to evaluate the uncertainty associated to the items whose category is not known yet; the choice of the next question is based on evaluating the margin a posteriori (the value of maximum-margin optimization when adding a possible response to a question). Based on this intuition we proposed several strategies for selecting the next question to ask. We evaluated the proposed interactive elicitation procedure comparing the performance of the different strategies showing that the procedure quickly converges to the real optimal classification in very few interaction cycles.

We highlight that one important limitation of our framework is that we expect the decision maker to provide us with the limit profiles of the different categories. In

future works we will relax this assumption considering techniques to elicit as well the limit profiles (either in a preliminary step or in an integrated approach), therefore providing a complete method for interactive elicitation of a MR-Sort model. This task will be challenging, since previous works on (non incremental) elicitation of MR-Sort have shown that, while it is possible to introduce integer variables [93] the resulting optimization is computationally very demanding and not scalable; therefore randomized heuristics [130,133] have been proposed.

We are also interested in performing simulation to compare our approach to other elicitation frameworks, as the recent work of Olteanu [107], and the approaches based on minimax regret [8,9], in realistic settings. We plan to investigate the connection between our approach and the field of machine learning. First, maximum-margin bears much similarity to Support Vector Machines (SVM). Second, ordinal classification has also been studied in machine learning. Third, there is strong similarity between incremental elicitation and active learning.

Finally, another important direction is to elicit the preferences of several users, providing methods that can exploit the similarity between users, as in Teso et al. [136].

Chapter 4

Group-based Ordinal Classification based on a Negotiation Process

Contents

4.1	Introduction	108
4.2	Multi-agent classification architecture	109
4.3	Classification process	110
4.4	Agents' description	112
4.4.1	User agents	114
4.4.2	Mediator agent	116
4.5	Illustration	119
4.5.1	Context	119
4.5.2	Application of the developed negotiation process	120
4.6	Experimental Approach	123
4.6.1	Evaluation criteria	125
4.6.2	Experimental results	128
4.7	Sensitivity analysis	132
4.7.1	Experimental process	132

4.7.2	Experimental settings	133
4.7.3	Observed indicators	135
4.7.4	Experimental results	136
4.8	Behavioural analysis	141
4.9	Conclusion	144

This chapter is published in the International Joint Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT'20)

4.1 Introduction

The previous chapter presented the first contribution of this thesis which represents the pre-negotiation phase. Such a phase aims to answer the question: *how can we elicit the preferences of a single decision maker in a context of multiple criteria ordinal classification while asking her a limited number of questions?* In fact, after helping a decision maker to understand her preferences and to generate the classification of alternatives to categories, we proceed to the negotiation phase which completes the objective of this thesis and answers the major question *how to make a group of decision makers converge to a group classification?* In this chapter, we propose a multi-agent group decision making process based on a multilateral negotiation to reach common ordinal classifications from individual preferences. Each user involved in the classification process is represented by a user agent acting on behalf of her preferences, constraints and goals. We assume that all agents are cooperative and share the common goal of reaching an agreement. However, local preferences could be different or even contradictory and common classifications become hard to get to. In order to avoid such conflictual situations, a mediator agent is given the task of initiating and conducting the negotiation process. A key role of the mediator agent is to make an objective decision when the negotiation doesn't lead to an agreement.

Each user agent behaves dynamically all along the process; she makes her decisions based on an aggregation of several parameters representing her profile such as

flexibility over time. The communication protocol is characterized by a high degree of privacy, hence, agents decide whether to reveal entirely or partially their preferences. The chapter is organized as follows. In the next section we give an insight about the global process architecture proposed in this thesis. Then, we focus on the negotiation phase by introducing our classification process. Section 4.1 presents an overview of multi-agent classification architecture. Section 4.3 is dedicated to detail agent's algorithms and negotiation strategies. We present in Section 4.6 our experimental approach and results. In Section 4.8 and 4.7, we present respectively the performed sensitivity and behavioral analysis and their results. Finally we provide some concluding remarks.

In the literature, several works have addressed the problem of group decision making in order to make a single choice for the group [14, 56, 57, 84]. The main novelty of our work is therefore to tackle classification problems. Some works [17, 27, 74] deal with ordinal classification problems involving multiple decision makers. However, they do not consider any dynamic external factor and agents' privacy is not respected as agents need to share their preferences in order to reach a group classification. The focus of our work is to derive a group classification while taking into account the change of agents' behavior and optimizing their private information disclosure.

4.2 Multi-agent classification architecture

To derive a group classification, we have used multi-agent theory to cope with communication and collaboration issues that are crucial in any decision making system. Agents are able to learn about their past experience and consequently to improve their decision-making process by adjusting their strategies. Thus, the agents' behaviour may change depending on the environment they are deployed on.

In addition, there are common agreed properties of agents, which include autonomy, proactiveness, responsivity and adaptability. Hence, agents are also able to know and learn human users' preferences when they interact with them [80].

Agents typically have a partial control over their environment. Each agent has a limited viewpoint and incomplete information. Figure 4.1 represents the distributed architecture of our multi-agent system to derive the group classification. It displays on the one hand the pre-negotiation phase (detailed in the previous chapter) and on the other hand the negotiation phase.

In the pre-negotiation phase, each user agent communicates with a human user and elicits her preferences in order to derive her ordinal classification. Therefore, she has a communication module which is implemented to communicate either with human user or with the mediator agent with respect to the communication protocol. Such a communication protocol specifies the types of allowed interactions and the actions that she is authorized to make during the elicitation process. Moreover, the user agent has a decision module which allows her to choose the "best" strategy for selecting the "best" question since she wish to minimize elicitation effort by asking the human user the most potentially informative question. Then, she constructs the question and sends it to the human user. The latter answers the question and the response will be sent and incorporated in the elicitation module that will update the elicitation process (explained in the previous chapter). The procedure stops after k questions. At the end of the process, the user agent derives the ordinal classification of the human user.

Once each user agent generates the ordinal classification of all the alternatives, the negotiation phase starts. Such a phase is guided by the mediator agent that owns as well a communication module in order to communicate with the user agents according to a communication protocol. The mediator agent has a decision module that allows her to make a decision in case user agents are not able to reach an agreement about a classification of a given alternative. The role of the mediator agent is detailed in the section 4.

4.3 Classification process

The classification process is based on a negotiation between a mediator agent and several user agents. The user agents share a common goal that is to incrementally

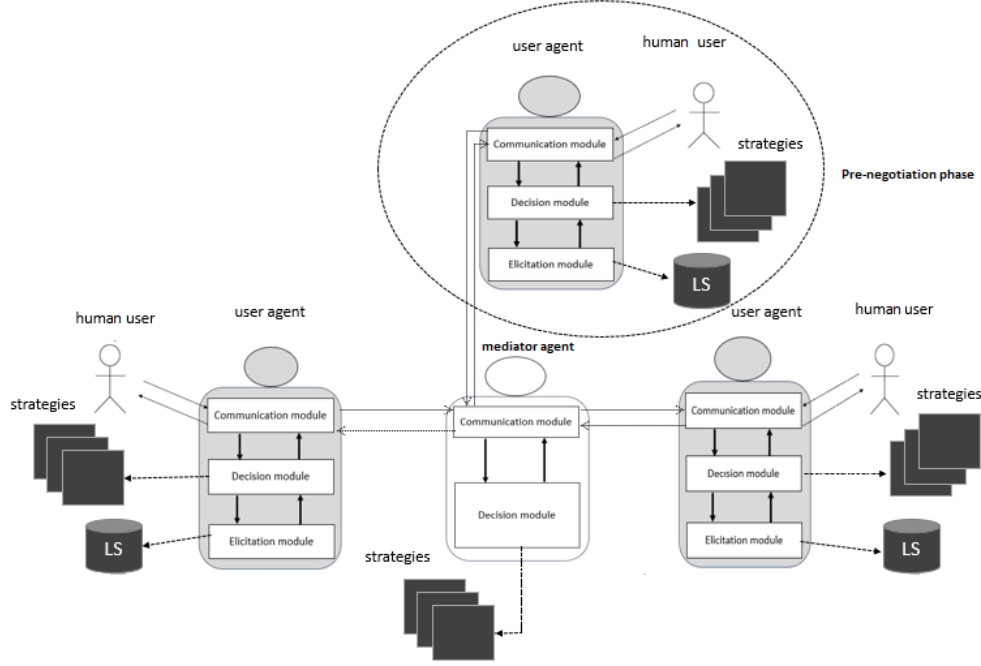


Figure 4.1 – Architecture de classification multi-agents

construct common classifications. Since agents could have different preferences, the mediator agent insures the convergence of the process.

Before detailing the process, we present some useful preliminary notations :

- $\mathcal{A} = \{a_1, \dots, a_p\}$, with $p \geq 2$: the set of user agents. Each user agent represents a decision maker.
- m : the mediator agent.
- $\mathcal{X} = \{x_1, x_2, \dots, x_n\}$: the set of alternatives considered for classification.
- $\mathcal{C} = \{c_1, \dots, c_q\}$, $q \geq 2$: the set of ordered categories to which alternatives will be classified. We assume that, $\forall k \in \{1, \dots, q-1\} : c_{k+1}$ is better than c_k .
- $cl : \mathcal{A} \times \mathcal{X} \longrightarrow \mathcal{C}$: agents' classifications. $cl(a_i, x_j) = c_k$ means that for agent a_i , the preferred classification of the object x_j is the category c_k .
- $c_* : \mathcal{X} \longrightarrow \mathcal{C}$: the common/final classifications ; $c_*(x_j) = c_k$ means that the object x_j is classified in the category c_k at the end of the process.

- α , with $\alpha \in]0.5, 1]$: the majority threshold enabling a coalition of agents to be decisive.
- $\hat{t}$: the negotiation deadline.

The classification process is a negotiation initiated by the mediator agent. At each round t , the mediator agent asks all user agents about how they would classify a given alternative. Each user agent has her own preference-based behavior; one agent may just communicate her preferred category while another user agent communicates a set of “possible categories” (depending on whether or not she is flexible or selfish, and on her sensitivity to the passing of time, etc.)

Once all user agents have sent their possible categories, the mediator agent makes one of the 2 following decisions :

- 1) she classifies the alternative: when a qualified majority of user agents agreed with the classification (to break ties, the mediator uses the plurality rule, i.e she chooses the one supported by the highest number of agents).
- 2) she postpones the classification: when there is no sufficient majority supporting the classification of the alternative to any category.

After proposing to the user agents all the alternatives for classification and while the process deadline is not reached, the mediator agent applies the whole process to the alternatives for which the classification has been postponed. Some of the user agents may change their set of possible categories during the process (for instance they can be more flexible when the deadline is close). When the negotiation time is over, the mediator agent uses the plurality strategy to classify the remaining alternatives. Figure 4.2 gives an overview of our classification process.

4.4 Agents' description

As explained in section 4.3, our multi-agent system is formed by user agents representing human decision makers and a mediator agent responsible of the conducting

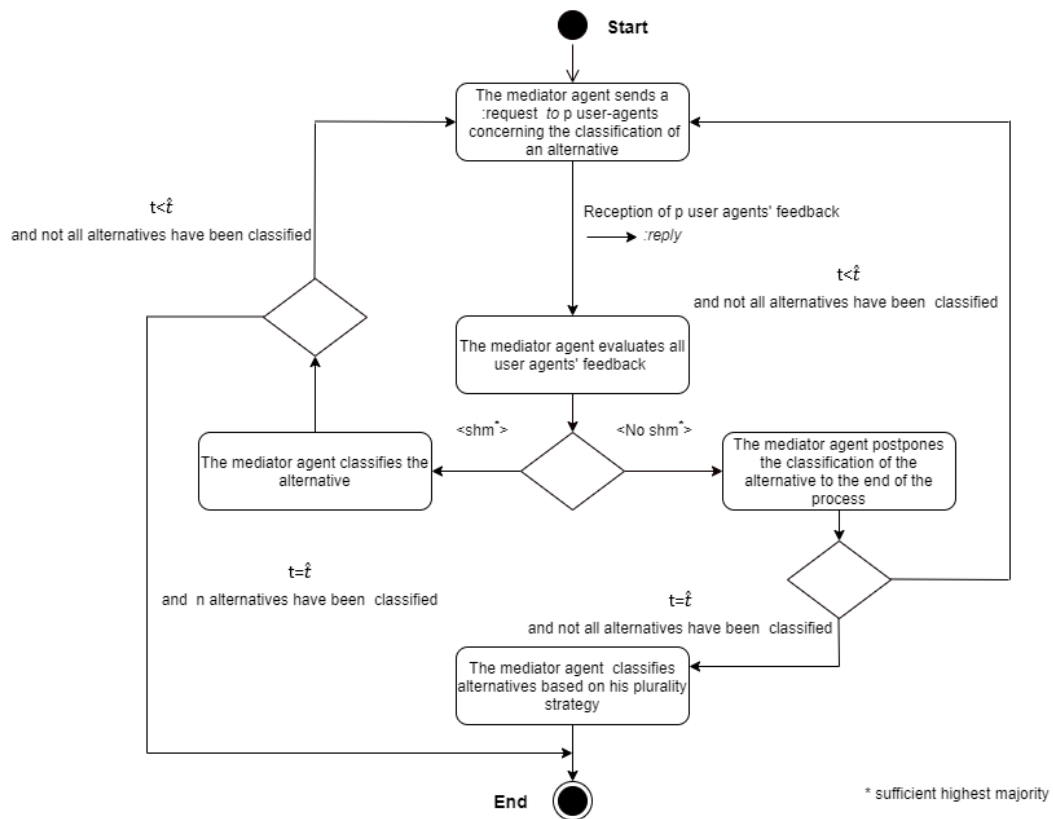


Figure 4.2 – Group-classification process

process.

The process follows a communication protocol that specifies the valid actions of each agent. Recall that agent communication was discussed in Chapter 2 Section 2.3.3. All the communication performatives are presented in Table 4.1.

Action	Semantic	Context
$\text{request}(\mathcal{A}, x)$	m requests all $a_i \in \mathcal{A}$ about the set of possible categories for classifying the alternative x	as long as it is still time and there are still alternatives to classify
$\text{reply}(m, C)$	the user agent replies m by a set of possible categories	in response to a request action
$\text{inform}(\mathcal{A}, x, c)$	m informs all $a_i \in \mathcal{A}$ about the classification of alternative x in the category c	if an agreement is reached about a proposed classification

Table 4.1 – Agent performatives

4.4.1 User agents

Each user agent is unique and has a different preference-based behavior. A global utility function, kept private during the whole process, represents its behavior. When the mediator agent asks about the classification of an object x , the user agent calculates for each category the utility of classifying x in it. If she judges the utility sufficient then this category is possible for her. The user agent may change her opinion (thus the value of the utility function) about the possible classification of x during the process (because the deadline is close or because she has already done a lot of concessions, etc.).

The utility function is defined as $U : \mathcal{X} \times \mathcal{C} \times \mathbb{N} \longrightarrow [0, 1]$; $U(x, c_k, t)$ represents the utility of classifying x in c_k when $t < \hat{t}$. The threshold λ is used in order to decide if the classification is possible or not: $v : \mathcal{X} \times \mathcal{C} \times \mathbb{N} \longrightarrow \{0, 1\}$:

$$v(x, c_k, t) = \begin{cases} 1 & \text{If } U(x, c_k, t) \geq \lambda \\ 0 & \text{Otherwise} \end{cases}$$

There are different criteria that have an impact on the utility function U , namely: the difference between the decision maker's preferred classification and the proposition of the mediator, the number of concessions already done and the remaining time. These three criteria are evaluated with respect to three subutility functions u , fl , and f , that we now discuss:

- the subutility $u : \mathbb{N} \longrightarrow [0, 1]$ is associated to the deviation of c_k from the agent's preferred classification cl ; the deviation is taken as the displacement, i.e. $d(c_i, c_j) = |i - j|$. u is used to model the flexibility/tolerance in accepting a different classification. Naturally, $u(0) = 1$ (this happens when the proposition of the mediator is the user's preferred category) and u is not increasing.
- the subutility $fl : \mathbb{N} \longrightarrow [0, 1]$ is associated to the current time (measured in number of rounds), $t \in \mathbb{N}$, where $t \leq \hat{t}$. fl is used to model the flexibility of the agent with respect to time preferences; supposing that the aim of the

group of decision makers is to reach a common classification, this function will be not decreasing (the more we get close to the deadline, the more the agent is leaning towards accepting to collaborate).

- the subutility $f : \mathbb{N} \rightarrow [0, 1]$ is associated to the number of concessions $r \in \mathbb{N}$ previously made by a user agent. f models the agent's disposition towards making a new concession, when r concessions have been made before. This function will be decreasing (the more the agent has already done concessions, the more she gets reluctant toward accepting to collaborate). A “concession” occurs when the preferred classification of the user agent is different from the group classification.

Each user agent can aggregate these three subutilities in different ways. The choice of the aggregation operator depends on the decision maker's behavior. She can use a weighted sum which is very compensatory or multiplicative form which is less compensatory, she can also apply more sophisticated rules (for instance integrating vetos or logical rules, etc.). For the rest of our article we present three possibilities :

- A *flexible* strategy makes use of a compensatory operator Δ^1 ; the aggregation is as follows:

$$\Delta(A, B, C) \rightarrow \frac{u(A) + u(B) + u(C)}{3}.$$

An agent adopting this strategy is quite flexible over the acceptance of a classification: a low performance on one criteria A , B or C , may be compensated by high performances on other criteria.

- An *inflexible* strategy makes use of a less-compensatory operator ∇^2 . The aggregation is represented as follows:

$$\nabla(A, B, C) \rightarrow \sqrt[3]{u(A) \times u(B) \times u(C)}.$$

¹Within this work, Δ is defined as follows: $\Delta(A_1, \dots, A_n) = \frac{\sum_{i=1}^n u(A_i)}{n}$

²Within this work, ∇ is defined as follows: $\nabla(A_1, \dots, A_n) = \sqrt[n]{\prod_{i=1}^n u(A_i)}$

An agent following this strategy is less flexible over the acceptance of a classification. Roughly speaking, all the performances on criteria A , B and C should be good in order to get a good value in output.

- *Hybrid* strategies make use of combinations of the operators Δ and ∇ . The aggregation is performed in one of the following ways:

$$- \nabla(A, \Delta(B, C)) \longrightarrow \sqrt{u(A) \times \left(\frac{u(B)+u(C)}{2}\right)}.$$

An agent following this strategy allows compensation between B and C . The result of such compensation is aggregated with A following the inflexible strategy.

$$- \Delta(A, \nabla(B, C)) \longrightarrow \frac{u(A) + \sqrt{u(B) \times u(C)}}{2}$$

An agent following this strategy combines the $u(B)$ and $u(C)$ in a less-compensatory way, but allows compensation between A the pair B and C .

Algorithm 3 describes the negotiation process from the point of view of the user agent. The process starts by the request of the mediator. Once the user agent receives the **request** (line 10), she computes the U function in order to define the set of possible classification categories for x and to communicate them to the mediator. When the user agent receives the **inform** message (line 20) of the mediator agent precising the final classification of x , she updates consequently the number of concessions.

4.4.2 Mediator agent

The role of mediator agent is to coordinate the classification process in order to reach common classifications. The mediator agent has no information about the user agents' strategies.

Algorithm 3 : The UserAgent class

```

1 agent UserAgent:
2   properties
3      $r \leftarrow 0$                                 /* n. of concessions */
4      $\lambda$                                        /* acceptance threshold */
5   end
6   function main():
7      $t \leftarrow 1$ 
8     while true do
9       msg  $\leftarrow$  listenMessages()
10      if msg.getPerformative() == request then
11         $t \leftarrow t + 1$ 
12         $x \leftarrow$  msg.getContent()
13         $C \leftarrow \emptyset$ 
14        for  $c \in \mathcal{C}$  do
15          if  $U(x, c, t) > \lambda$  then
16             $C \leftarrow C \cup c$ 
17          end
18        end
19        reply (m, C)                                /* send reply */
20      else if msg.getPerformative() == inform then
21         $(x, c_*) \leftarrow$  msg.getContent()
22        if  $c_* \neq cl(x)$  then
23           $r \leftarrow r + 1$                                 /* concession */
24        end
25      end
26    end
27  end
28 end

```

In Algorithm 4, the mediator agent maintains a queue of alternatives that need to be classified, initialized as the set of alternatives. She starts by picking the alternative x at the top of the queue and sends a common request to all user agents concerning the classification of the alternative x (line 10). Then, she collects the replies of all user agents (line 11), and verifies if there is a category supported by a qualified majority. If this is the case, then the mediator agent adopts the category as the classification of x (if more categories are acceptable to a majority, she chooses the one with the highest number) and informs user agents. Otherwise, alternative x is added at the end of the queue of alternatives. The process repeats until all alternatives are classified or until we reach the end of the negotiation time $\hat{t}$. When

Algorithm 4 : The MediatorAgent class

```

1  agent MediatorAgent:
2    properties
3       $\alpha$                                 /* majority threshold */
4       $\hat{t}$                                 /* negotiation deadline */
5    end
6    function main():
7       $Q \leftarrow \text{Queue}(\mathcal{X}), t \leftarrow 1$ 
8      while  $\neg \text{isEmpty}(Q) \wedge t \leq \hat{t}$  do
9         $x \leftarrow Q.\text{dequeue}()$ 
10       request( $\mathcal{A}, x$ )
11        $R_1, \dots, R_p \leftarrow \text{receiveAllReplies}()$ 
12        $c \leftarrow \text{highestMajority}(\alpha, R_1, \dots, R_p)$ 
13       if  $c == \text{null}$  then
14          $Q.\text{enqueue}(x)$ 
15       else
16         Classify  $x$  in  $c$ 
17         inform( $\mathcal{A}, x, c$ )
18       end
19        $t \leftarrow t + 1$ 
20     end
21     if  $\neg \text{isEmpty}(Q)$  then
22       pluralityClassification( $Q$ )
23     end
24   end
25 end

```

the negotiation time is over (line 21) and some alternatives are still unclassified, the mediator agent makes a classification based on the `pluralityClassification` method and the process ends.

The plurality method consists in classifying an alternative to a category supported by the highest number of user agents. In case only one category is supported by the highest number of user agents, x will be classified in this category. Otherwise, the mediator agent considers the user agent's feedback for k nearest neighbor (k -NN) categories. This way of proceeding allows to choose the category that reduces the mean absolute deviation of the user agents' classifications. x will be classified in the category for which the k -NN categories are supported by the highest number of agents.

4.5 Illustration

In this section, we present an example of group classification problems.

4.5.1 Context

This case shows an example of the whole negotiation process as displayed in Figure 4.2.

Assume that we have 4 hotels $X = \{x_1, x_2, x_3, x_4\}$, considered as the alternatives in our process, and 4 friends $\mathcal{A} = \{a_1, a_2, a_3, a_4\}$, represented by 4 user agents. Their goal is to derive a common classification that can be seen as a classification of the hotels to 4 categories (C_1 : 1 star, C_2 : 2 stars, C_3 : 3 stars, C_4 : 4 stars). The common classification will be generated based on individual classifications. Table 4.2 displays the hotels' classifications for each of the group members.

x	$cl(a_1, x)$	$cl(a_2, x)$	$cl(a_3, x)$	$cl(a_4, x)$
x_1	C_1	C_4	C_3	C_4
x_2	C_2	C_4	C_1	C_3
x_3	C_3	C_3	C_2	C_4
x_4	C_4	C_3	C_4	C_4

Table 4.2 – The classification of the alternatives $x \in X$ by agents $a_i \in \mathcal{A}$ to the category $cl(a_i, x)$.

In order to derive a common classification through our negotiation process, we set the user agents' negotiation strategies, as well as the subutility functions. Table 4.3 displays negotiation strategies for user-agents a_1, a_2, a_3 and a_4 .

a_1	a_2	a_3	a_4
$\Delta(A, \nabla(B, C))$	$\Delta(A, B, C)$	$\Delta(\nabla(A, B), C)$	$\nabla(A, B, C)$

Table 4.3 – Negotiation strategies of user-agents a_1, a_2, a_3, a_4

Table 4.4 displays the subutility functions associating each user agent to three criteria: the user-agent's deviation, the user-agent's flexibility with respect the negotiation deadline $\hat{t}$, and the user-agent's flexibility regarding the number of concessions made all along the process.

$d(c_i, c_j) \setminus a_i$	a_1	a_2	a_3	a_4
$u_i(1)$	0.5	0.6	0.6	0.6
$u_i(2)$	0.5	0.5	0.3	0.5
$u_i(3)$	0.2	0.4	0.1	0.4
$t \setminus a_i$	a_1	a_2	a_3	a_4
$fl_i(1)$	0.2	0.3	0.3	0.3
$fl_i(2)$	0.2	0.3	0.4	0.4
$fl_i(3)$	0.4	0.4	0.4	0.5
$fl_i(4)$	0.6	0.7	0.5	0.7
$fl_i(5)$	1	1	1	1
$r \setminus a_i$	a_1	a_2	a_3	a_4
$f_i(1)$	0.5	0.6	0.5	0.7
$f_i(2)$	0.4	0.6	0.4	0.5
$f_i(3)$	0.3	0.5	0.4	0.3
$f_i(4)$	0.3	0.4	0.2	0.3

Table 4.4 – The subutility values $u_i(d(c_i, c_j))$, $fl_i(t)$, and $f_i(r)$ associating user-agent a_i to respectively the deviation $d(c_i, c_j)$, the current time t and number of concessions made previously r .

We assume that in case the proposed classification matches with user-agent a_i classification, $u_i(0)$ will take the value 1. Similarly, in case an agent a_i did not make concessions, $f_i(0)$ will take the value 1. We fix the acceptance threshold $\lambda = 0.5$ and majority threshold $\alpha = 0.6$.

4.5.2 Application of the developed negotiation process

Our negotiation process is organised in several rounds. In each round, the mediator agent will derive a common classification based on the user-agents' subutility functions and negotiation strategies. This section is dedicated to detail the different rounds in our process.

At the beginning, $c = \{c_1, c_2, c_3, c_4\}$. The mediator agent chooses hotel x_1 randomly. Then, she makes a common request about the classification of this alternative to all agent. Each agent replies the mediator agent by sending to her the set of accepted categories based on her own strategy.

Based on the the user-agents' feedback, the mediator agent computes the proportion of user agents who accept each classification.

Tables 4.6, 4.7, 4.8, 4.9 display the values of the function $(v_i)_{i \in \{1,2,3,4\}}$ associated to each agent $(a_i)_{i \in \{1,2,3,4\}}$ concerning the classification of respectively x_1 , x_2 , x_3 , and x_4 to the proposed categories.

At the end of round 1, $c = \{C_1, C_4\}$. $|c(x_1)| > 1$ so the mediator agent should classify x_1 to the category supported by the highest majority. In such case we have the equivalent majorities supporting both c_1 and c_4 . Thus, the mediator agent chooses randomly one of the two categories.

At the end of round 2, $c = \emptyset$. Thus, x_2 is added at the end of the queue of alternatives Q in order to postpone its classification.

In round 5, the mediator agent requests again the alternative x_2 to all user-agents. There is no agreement again and the negotiation deadline $\hat{t}$ is reached. In fact, Q contains two remaining alternatives x_2 and x_3 . In this case, the mediator agent makes an objective decision based in the plurality strategy. In case we have the equivalent plurality, The mediator agent considers the user agents' feedback for k -nearest neighbors ($k - NN$) categories. Thus, she chooses to classify x_2 in c_4 and x_3 in c_4 .

In order to illustrate the way v_i is computed for an agent a_i , we consider the example of a_1 in round 2. Let's see how she calculated $v_1(x_2, c_4, t)$. The negotiation strategy associated to agent a_1 is $\Delta(A, \nabla(B, C))$. For $v_1(x_2, c_4, 2)$, the agent has to deviate by two categories from her initial classification. The utility associated to such deviation for a_1 is 0.2. The flexibility of the agent with respect to negotiation time in the second round is 0.2. The agent a_1 has already made concessions in the first round. So, her flexibility to make concessions is 0.5. Hence, the computation of $v_1(x_2, c_4, 2)$ will be:

$$U_1(x_2, c_4, 2) = \frac{u_1(2) + \sqrt{(f_1(2) \times f_1(1))}}{2} = 0.408 < \lambda$$

Thus, $v_1(x_2, c_4, 2) = 0$.

At the end of the round 5, the mediator agent obtains a common classification

satisfying the higher majority of the agents and taking into account their behaviours.

	c_1	c_2	c_3	c_4	c	Q	Common classification
$v_1(x_1, c_j, 1)$	1	0	0	0	$\{c_1, c_4\}$	$\{x_2, x_3, x_4\}$	$x_1 \longrightarrow c_4$
$v_2(x_1, c_j, 1)$	1	1	1	1			
$v_3(x_1, c_j, 1)$	1	1	1	1			
$v_4(x_1, c_j, 1)$	0	0	0	1			

Table 4.5 – The user agents' feedback and the common classification of x_1 after round 1

	c_1	c_2	c_3	c_4	c	Q	Common classification
$v_1(x_2, c_j, 2)$	0	1	0	0	$\emptyset$	$\{x_3, x_4\}$	$x_1 \longrightarrow c_4$
$v_2(x_2, c_j, 2)$	1	1	1	1			
$v_3(x_2, c_j, 2)$	1	0	0	0			
$v_4(x_2, c_j, 2)$	0	0	1	0			

Table 4.6 – The user-agents' feedback and the common classification of x_2 after round 2

	c_1	c_2	c_3	c_4	c	Q	Common classification
$v_1(x_3, c_j, 3)$	0	0	1	0	$\{c_2, c_3\}$	$\{x_4, x_2\}$	$x_1 \longrightarrow c_4$
$v_2(x_3, c_j, 3)$	1	1	1	1			$x_3 \longrightarrow c_3$
$v_3(x_3, c_j, 3)$	0	1	0	0			
$v_4(x_3, c_j, 3)$	0	1	1	1			

Table 4.7 – The user-agents' feedback and the common classification of x_3 after round 3

	c_1	c_2	c_3	c_4	c	Q	Common classification
$v_1(x_4, c_j, 4)$	0	1	1	1	$\{c_3, c_4\}$	$\{x_2\}$	$x_1 \longrightarrow c_4$
$v_2(x_4, c_j, 4)$	1	1	1	1			$x_3 \longrightarrow c_3$
$v_3(x_4, c_j, 4)$	0	0	0	1			$x_4 \longrightarrow c_4$
$v_4(x_4, c_j, 4)$	0	0	1	1			

Table 4.8 – The user-agents' feedback and the common classification of x_4 after round 4

	c_1	c_2	c_3	c_4	c	Q	Common classification
$v_1(x_1, c_j, 5)$	1	1	1	1	$\{c_1, c_2, c_3, c_4\}$	$\emptyset$	$x_1 \longrightarrow c_4$
$v_2(x_1, c_j, 5)$	1	1	1	1			$x_3 \longrightarrow c_3$
$v_3(x_1, c_j, 5)$	1	1	0	0			$x_4 \longrightarrow c_4$
$v_4(x_1, c_j, 5)$	1	1	1	1			$x_2 \longrightarrow c_2$

Table 4.9 – The user agents' feedback and the common classification of x_2 after round 5

4.6 Experimental Approach

In this section we present an application inspired from a real case. We consider that a group of persons (playing the role of experts) have to decide in which category they classify different touristic destinations for a travel magazine. Their aim is to get a *collective* classification even if their personal preferences differ and may be contradictory. Our case study uses real data on the classifications and simulated data on the behaviors of experts.

We collect the classification of each expert to whom we assign a user agent a_i . We have 38 experts in total ($\mathcal{A} = \{a_1, \dots, a_{38}\}$), 50 destinations ($X = \{x_1, \dots, x_{50}\}$) and five categories (c_1, c_2, c_3, c_4 and c_5 representing “very bad”, “bad”, “medium”, “good”, “very good”). Table 4.10 illustrates an extract of the collected classifications; notice that while for some destinations there is a form of consensus (NewYork, Bali), for others a “common” classification seems to be hard to obtain (Theran, Ibiza). Interestingly, for a given destination, the distribution of experts over categories is almost single-peaked (with Theran being the exception).

	c_1	c_2	c_3	c_4	c_5
destination	very bad	bad	medium	good	very good
New-York	1	2	3	12	20
Tehran	10	6	8	7	7
Bali	2	2	3	12	19
Dubrovnik	3	2	19	11	3
Dubai	5	7	7	12	7
Brussels	4	10	17	7	0
Oslo	2	7	13	15	1
Tanzania	4	6	13	7	8
Istanbul	3	3	9	15	8
Ibiza	3	6	9	9	11

Table 4.10 – Distribution of expert classifications on a sample of 10 travel destinations (the total number of experts is 38).

We sample utility functions representing the behavior of the experts according to uniform distributions. We test our approach on three different groups, each follows a different negotiation strategy:

destination	median	plurality (mode)
New-York	c_5	c_5
Tehran	c_3	c_1
Bali	c_4	c_5
Dubrovnik	c_3	c_3
Dubai	c_3	c_4
Brussels	c_3	c_3
Oslo	c_3	c_4
Tanzania	c_3	c_3
Istanbul	c_4	c_4
Ibiza	c_4	c_5

Table 4.11 – Median and mode values for a sample of travel destinations according to the distribution of expert classification. The mode can be seen as the classification picked with a voting procedure when adopting plurality as a social rule.

- 1) Group 1 (the “flexible” agents): all agents are flexible and have the flexible strategy $\Delta(A, B, C)$.
- 2) Group 2 (the “inflexible” agents) : all agents have the inflexible strategy $\nabla(A, B, C)$.
- 3) *Group3* (“hybrid”): 10 agents with the flexible strategy, 10 agents with the non flexible strategy and 18 agents with the hybrid strategies; 9 agents have $\nabla(A, \Delta(B, C))$ strategy and 9 agents have $\Delta(A, \nabla(B, C))$ strategy. The distribution of these strategies to each agent changes from a simulation to another.

Our process was designed and developed using Java Agent Development Environment (JADE)³ for implementing the multi-agent system.

For each group, we run 30 simulations using different utility functions of the user agents. We test our results for different values of the majority threshold α . To ensure that the different results are comparable, in all simulations the alternatives are placed in the queue in the same order. The acceptance threshold is fixed to $\lambda = 0.5$ for all the user agents.

In this experiment below, we compare our multi-agent approach to two simple centralized approaches (that are not based on negotiation) assuming that the ex-

³<https://jade.tilab.com/>

perts give all their preferred classifications in the beginning of the process; thus with full revelation. We choose two intuitive centralized methods as baselines:

- the **median** method selects as a classification the category that is the statistical median of the distribution of expert classifications;
- the **plurality** method selects as a classification the category supported by the highest number of agents; i.e. the mode of the distribution of expert classifications.

In Table 4.11 we show how these two methods classify the alternatives in the travel domain; notice that, for instance, Tehran is classified *medium* by the median, while it is *very bad* according to plurality.

We evaluate the performance of our process using two types of indicators. The indicators of the first type focus on the individual satisfaction of the agents, while those of the second type measure the amount of information revealed by each agent along the process. These indicators are discussed next.

4.6.1 Evaluation criteria

Average Dissatisfaction (AvD) We compute the deviation between the final classification and the experts classifications. The results are averaged with respect to the experts and the travel destinations. The AvD from a group classification will be defined as:

$$AvD = \sum_{i=1}^p \sum_{j=1}^n \frac{|cl(i, x_j) - c_*(x_j)|}{n \times p}$$

where $c_*(x_j)$ is the final classification of destination x_j .

Maximum Dissatisfaction ($MaxD$) We compute the highest value of dissatisfaction (averaged over alternatives) among the experts:

$$MaxD = \max_{i=1, \dots, p} \frac{1}{n} \sum_{j=1}^n |cl(i, x_j) - c_*(x_j)|.$$

Dispersion of Dissatisfaction Index (DD) We measure the inequality with respect to the total dissatisfaction among experts; DD is inspired from the GINI index [59] which is used to measure inequality of income or wealth distribution of a population. In our domain, the fairest classification consists in the situation where all agents have the same value of average deviation from the final classification. The DD is based on the comparison of cumulative proportions of the experts against cumulative proportions of dissatisfaction they have, and it ranges between 0 in the case of perfect equality and 1 in the case of perfect inequality. For instance if we have 3 experts such that expert 1 has in average 3 deviations from the final classification, while experts 1 and 2 has no deviation, the DD will be 1 since a fair distribution of total dissatisfaction is 1 deviation for each expert and the difference between cumulative proportions is: $\|(0, 0, 3) - (1, 2, 3)\| = 1 + 2 + 0 = 3$ for 3 experts, hence $DD = 3/3 = 1$. In more details, the DD corresponds to the difference between two Lorenz curves:

- the first one is represented by the identity function, displaying the theoretical fair classification.
- the second is represented by a normalized cumulative function displaying the ratio between the deviations⁴ from a group classification, and the global deviation⁵.

Thus, the DD can be defined as follow:

- 1) We define the agent's deviation D as a mapping assessing the average deviation for each agent. D is defined as:

$$\begin{aligned} D : \{1, \dots, p\} &\longrightarrow [0, q - 1] \\ i &\longmapsto \sum_{j=1}^n \frac{|cl(i, x_j) - c_*(x_j)|}{n}. \end{aligned}$$

- 2) We define the global deviation GD as:

$$GD = \sum_{i=1}^p D(i).$$

⁴Sorted from the less deviating to the most deviating user agent.

⁵The sum over all user agents' deviations

- 3) To sort the deviation values, we use a permutation function σ , defined as on $[1, p]$, such that: $\forall i, j \in [1, p]$

$$D(i) < D(j) \Rightarrow \sigma(i) < \sigma(j)$$

Finally, the function s is a mapping defined as follow:

$$\begin{aligned} s &: \{1, \dots, p\} \longrightarrow [0, 1] \\ i &\longmapsto \sum_{j=1}^i \frac{D(\sigma^{-1}(j))}{GD} \end{aligned}$$

where σ^{-1} is the inverse of σ .

Hence, the DD is defined as follows:

$$DD = 2 * \sum_{i=1}^p \frac{\frac{i}{p} - s(i)}{p}.$$

The term 2 is used in order to normalize the index between zero and one.

Information Privacy Index (IPI) It is related to the amount of information shared between user agents and the mediator agent. When we use centralized methods (median and plurality), experts disclose all their classifications. When we use our approach, the preference based behavior of experts (utility functions, aggregations, thresholds) are only known by their corresponding user agents. Moreover, a user agent does not communicate about her preferred classifications. She only communicates about “possible” classifications (calculated using the preferred classification of the expert and her utility function, etc.). For instance, it may happen that the user agent says that all the categories are possible, which means that the mediator did not obtain any clear information about the preferred classification of the expert.

Thus, the basic idea of our indicator is the following: the more “possible categories” are communicated to the mediator, the less information is shared. The privacy index $IPI(i, x_j)$ measures the information shared with the mediator (if the

user agent reports a single category, this has to be the preferred category, that is revealed to the mediator). We define $IPI(i, x_j)$ as following:

$$IPI = \frac{1}{(q-1)} \left(\sum_{C \in \mathcal{C}} v_i(x_j, C) - 1 \right)$$

4.6.2 Experimental results

Our approach is based on a majoritarian rule (the default value of majority threshold α is 0.5). However one can use greater values of α in order to be more exigent about final classifications (for instance $\alpha = 0.7$ means that the category c_k can be chosen by the mediator if and only if 70% of user agents define it as possible). Naturally when α gets greater, a common classification becomes difficult to be reached and there are more and more postponed classifications (that will be decided using plurality). Our simulations show us that for $\alpha = 0.5$, in *Group1* (resp. *Group2* and *Group3*) 100% (resp. 98% and 45%) of alternatives are classified before the deadline, as the agents reach an agreement (hence without the intervention of the mediator agent). Recall that *Group1*, *Group2* and *Group3* represent respectively the flexible agents, the inflexible agents and the hybrid agents. This proportions get smaller when we increase α ; for $\alpha = 0.7$, in *Group1* (resp. *Group2* and *Group3*) 87% (resp. 19% and 8%) of alternatives are classified before the deadline.

Average Dissatisfaction(AvD) Figure 4.3 shows average dissatisfaction values depending on the value of majority threshold α . Five methods are tested: our approach (on *Group1*, *Group2* and *Group3*), the plurality, and the median methods. As expected, when we use our negotiation-based approach the average dissatisfaction of *Group1* is greater than the one of *Group2*, which is greater than the one of *Group3*.

In general, we notice that the results are convincing for all the methods, considering 5 categories, the classification of a travel destination given by an expert has

an average deviation around 0.7-0.8 which is less than one category (note that, as there are 5 categories, the theoretical maximum is 4).

In addition, we observe that the plurality and median methods have very similar values of AvD and our approach gets closer to them when experts become less flexible (*Group3*).

Not surprisingly, the bigger the value of α , the closer our approach to the plurality method. This is due to the increase of postponed classifications which may not be treated in time $\hat{t}$. Thus, the mediator agent makes an objective decision based on plurality over the user agents' feedback.

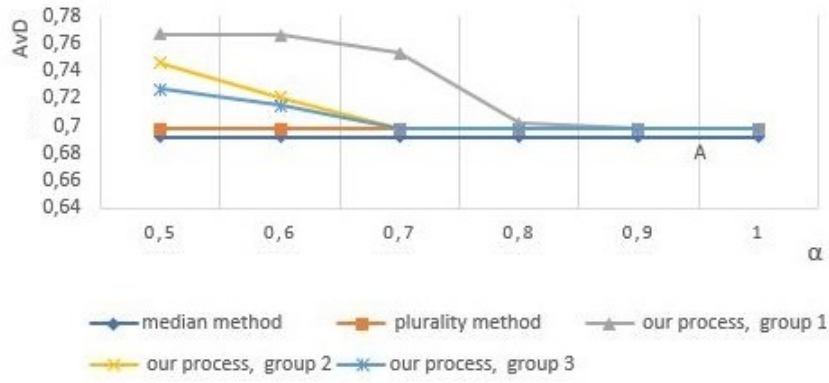


Figure 4.3 – Average Dissatisfaction (AvD) with $\hat{t} = 50$, $n = 50$, $p = 38$, $\lambda = 0.5$ and 30 simulations, where theoretical bounds are $\min(AvD) = 0$ and $\max(AvD) = 4$

Maximum Dissatisfaction Index ($MaxD$) Figure 4.4 presents the maximum dissatisfaction values with respect to different values of the majority threshold α . $MaxD$ values of our simulations are between 2.5 and 2.9 (by definition the range of the index is obviously between 0 and 4). It means that in our group of experts there exists at least one expert having contradictory preferences with the majority of experts and her average deviation from the final classifications is between 2 and 3 categories. A quick look to our data shows that there are two experts who were very strict and classified a big number of destinations in the *very bad* category. These two individuals, whose preferences set them apart from the rest of the group, may have more influence towards the final result when the other experts are flexible

or the median method is used. This observation is coherent with the fact that the plurality approach “protects” less the minorities.

Unlike the average dissatisfaction, the maximum dissatisfaction of *Group1* is smaller than the one of *Group2*, which is smaller than the one of *Group3*. It shows that when the experts have flexible preferences, even if the average dissatisfaction of the group is higher then the one of a inflexible group, the unhappiest expert of a flexible group is more satisfied then the one of a mixed or inflexible group.

The values of $MaxD$ are bounded by the values obtained with the plurality (worst case) and the median (best case). Like with the AvD , not surprisingly, the bigger the value of α , the closer our approach to the plurality method.

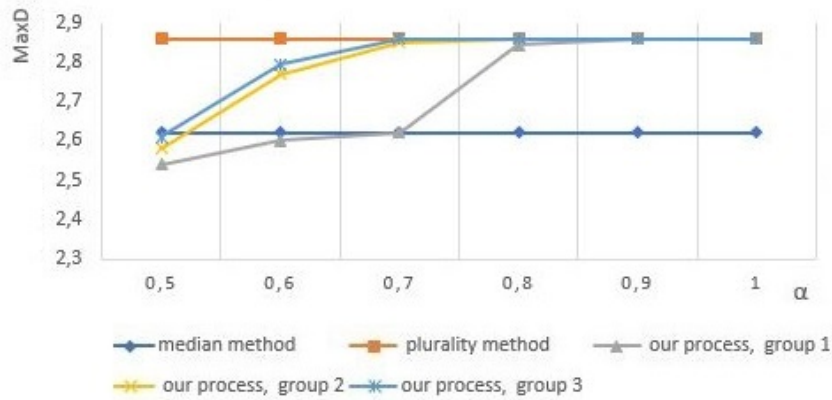


Figure 4.4 – Maximum Dissatisfaction ($MaxD$) as function of α with $\hat{t} = 50$, $n = 50$, $p = 38$, $\lambda = 0.5$ and 30 simulations, where theoretical bounds are: $\min(MaxD) = 0$ and $\max(MaxD) = 4$

Dispersion of Dissatisfaction Index (DD) Figure 4.5 shows the dispersion of dissatisfaction values with respect to different values of α . While the theoretical range of the index is between 0 and 1, in our simulations, we observe DD values between 0.16 and 0.25. This shows that all the methods perform quite fairly (as they distribute the global dissatisfaction in a somewhat “egalitarian” way). One explanation of this may be related to the nature of our case study. Even though our experts have different preferences over the destinations, our classification problem

is not that controversial.⁶

Notice that for α between 0.5 and 0.8, our approach provides more fair results than centralized methods and as in the case of *MaxD* and *AvD*, the greater α , the closer we are to the plurality. With respect to *DD*, our method performs better when agents are of *Group1* than with *Group2*, and even worse with agents of *Group3*.

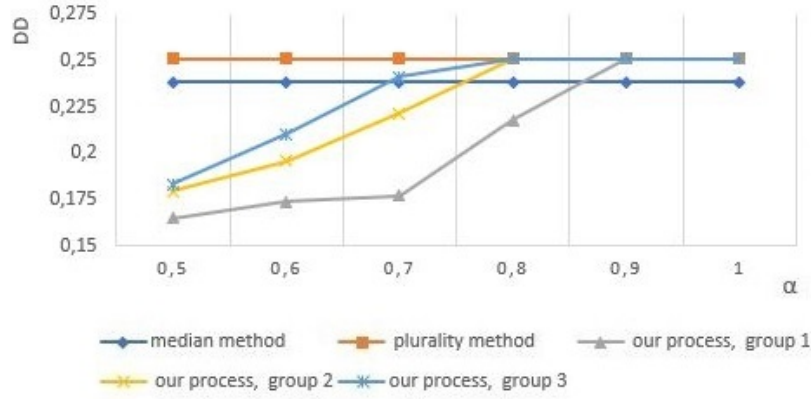


Figure 4.5 – Dispersion of Dissatisfaction Index (*DD*) with $\hat{t} = 50$, $n = 50$, $p = 38$, $\lambda = 0.5$ and 30 simulations, where theoretical bounds are $\min(DD) = 0$ and $\max(DD) = 1$.

Information Privacy Index (*IPI*) Figure 4.6 illustrates information privacy index with respect to different values of α . As explained in the previous section, *IPI* is zero for centralized methods since the preferred classifications of all experts are known by the one who controls the process. Using our negotiation-based approach we observe values between 0.23 and 0.612 (the theoretical maximum is 1).

Not surprisingly, the more flexible are experts' preferences, the less information is shared (*Group1* is “better” than *Group2*, which is better than *Group3*). Contrary to the previous indicators, the difference between the values of our three groups are quite different. Remark that when our experts are all flexible, only less than 30% of classification information is shared with the mediator.

Unlike the other indicators, the values of *IPI* do not depend on the values of α .

⁶As an example of a domain where the classification is probably more controversial, consider the case of classifying nuclear energies as either *acceptable*, *neutral* or *not acceptable*.

For that reason, we remark that even for very big α , the *IPI* values of our approach do not get closer to the plurality and we perform better than centralized methods.

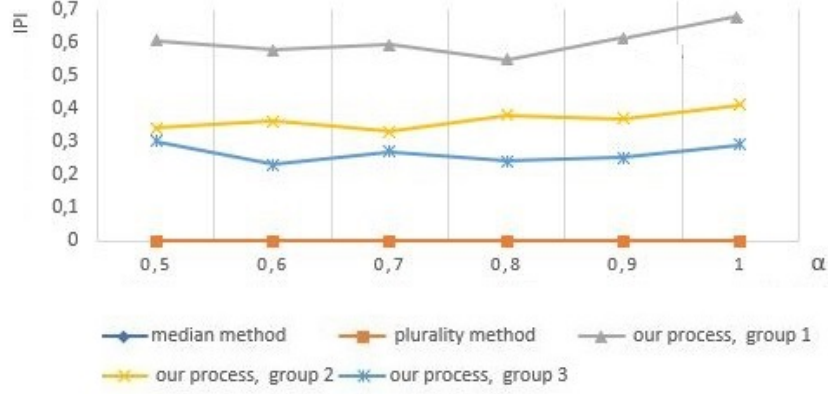


Figure 4.6 – Information Privacy Index (*IPI*) with $\hat{t} = 50$, $n = 50$, $p = 38$, $\lambda = 0.5$ and 30 simulations; the range of the index is between 0 and 1. For plurality and median, *IPI* is zero.

4.7 Sensitivity analysis

In this section, we present the results of the conducted sensitivity analysis over the parameters of our process. For this aim, we detail the experimental process, we highlight the experimental settings, we present the observed indicators to evaluate the results of the process and then we analyze the experimental results.

4.7.1 Experimental process

The experimental process is conducted as follows:

We use the same real data used in the experiments presented in the previous section. We recall that these data correspond to the classification of 50 travel destinations by 38 experts into 5 ordered categories. The data generated randomly representing the utility functions (following a monotonous uniform distribution) and negotiation strategies of each expert are also fixed for a given series of simulations. Note that we improved the utilities for the sensitivity analysis since we do not need

to accentuate the difference in the agents' behaviours as in the previous section. More specifically, we considered that sub-utility values $fl_i(t)$ associating the user-agent a_i to respectively the current time t follows a monotonous uniform distribution ranging from 0.35^7 to 1.

4.7.2 Experimental settings

The experimental settings is performed as follows:

We keep fixed the following parameters :

- 1) The number of alternatives: the group classifications do not depend on the number of alternatives but on their order;
- 2) The number of categories: varying this parameter will change the problem we are dealing with (the result of increasing the number of categories by one will differ based on whether it is a new added category or a split of a former one into two categories);
- 3) The number of agents: varying the number of agents is similar to varying the majority threshold;
- 4) The agents classifications;
- 5) The utilities: varying the utilities is similar to varying the acceptance threshold.
- 6) The maximum time allocated for the negotiation process $\hat{t}$.

Since the parameters on which we are performing a sensitivity analysis are independent, we use the One-at-a-time (OAT) [70] sensitivity approach over the following parameters:

⁷To improve the acceptability of the alternative's classifications by inflexible agents, we set the minimum value that might be taken by fl_i to 0.35.

- 1) Majority threshold α : This parameter represents the qualified majority allowing to reach a group classification. We set all the parameters of the process and we run series of simulations with different values of α ranging from a given α_{min} to an α_{max} with a step ϵ such that:

$$\alpha = \alpha_{min}, \alpha_{min} + \epsilon, \dots, \alpha_{max}$$

In our case, we consider $\alpha_{min} = 0.5$, $\alpha_{max} = 1$ and $\epsilon = 0.01$.

- 2) Acceptance threshold λ : This parameter represents the tolerance of each agent to accept a classification. We set all the parameters of the process and we run series of simulations with randomly generated values of λ for each agent and in each series. Such way of processing allows to avoid increasing the acceptance threshold for all the agents simultaneously, since this last will lead to less flexible agents. Thus, this corresponds to generating randomly for an agent j and a simulation i , an $\epsilon_i \in [\epsilon_{i,min}, \epsilon_{i,max}]$

$$\lambda_{i,j} \pm \epsilon_i$$

In our case, we consider $\epsilon_{i,min} = 0$, $\epsilon_{i,max} = 0.15$.

- 3) Order of requests: As presented in agents' description of section 4.4, for each step the mediator agent sends a request to all the agents to classify an alternative randomly chosen. Therefore, there is no precise order in which she sends her requests throughout the negotiation process. However, the group classification depends on the order of questions since the agents' behavior vary based on the number of concessions she made and the negotiation time.

As a result, we analyze the sensitivity based on changing the order of requests chosen by the mediator from one simulation to another while keeping all the process parameters fixed. We conduct a series of q simulations with different order of requests *order* such that:

$$order = \{(request_{\sigma_1(1)}, \dots, request_{\sigma_1(n)}), \dots, (request_{\sigma_q(1)}, \dots, request_{\sigma_q(n)})\}$$

Where $request_{\sigma_i(j)}$ represents the request sent in the simulation i , by the mediator over the alternative $x_{\sigma_i(j)}$ such that $\forall i \in \{1, \dots, q\}$, σ_i is a permutation function (a bijection defined from $\{1, \dots, n\}$ to $\{1, \dots, n\}$).

4.7.3 Observed indicators

The sensitivity analysis was conducted over the group classifications. In order to synthesize these results we use the following metrics:

- 1) Mean distance between two successive simulations MS : computes the average distance of the process results between each two successive simulations i and $i + 1$ for a series of simulations where $i \in [1, q - 1]$.

$$MS = \sum_{i=1}^{q-1} \frac{dist(R_i, R_{i+1})}{q - 1}$$

Where R_i and R_{i+1} represents the group classification derived respectively during the simulations i and $i + 1$. We consider $dist$ as the euclidean distance between the group classifications.

Such a metric is mainly used to evaluate the sensitivity of the majority threshold parameter since its variation from one simulation to another is monotonous.

- 2) Mean distance between a group classification and the median group classification computed over all the simulations MM : computes the average distance of the current simulation's group classifications and the median group classifications of all simulations that we note R_M .

$$MM = \sum_{i=1}^q \frac{dist(R_i, R_M)}{q}$$

This metric is mainly used to evaluate the sensitivity of the acceptance threshold and order of requests parameters since their variation from one simulation to another is performed randomly.

4.7.4 Experimental results

We conducted 15 simulation series. Each series is characterized by varying a single parameter: either the majority threshold, or the acceptance threshold, or the order of questions.

Majority threshold Figures 4.7 shows the performed sensitivity analysis results over the group classification based on *MS* metric and the variation of the different evaluation indicators (*MaxD*, *DD*, *AvD*, *IPI* and mediator intervention) for 15 simulations' series. These results depend on the variation of the majority threshold α from 0.5 to 1 with a step of 0.01. Note that the mediator intervention is an indicator computing the amount of alternatives classified by the mediator agent (averaged over alternatives).

We notice that the results of the performed sensitivity analysis are stable when $\alpha \in [0.5, 0.79]$. This is due to a relatively low acceptance threshold (ranging from $[0.5, 0.65]$ for 70% of the agents), and the high proportion of agents with flexible and hybrid strategies leading to a stability of the group classifications within this interval. Not surprisingly, when the final classifications of the agents do not change from one simulation to another, the different evaluation indicators remain unchanged. This explains the stability of *MaxD*, *DD*, *AvD*, *IPI* and mediator intervention indicators when $\alpha \in [0.5, 0.79]$.

Moreover, we notice small variations of our indicators when $\alpha \in [0.8, 1]$. These variations are detailed above:

For *MS* metric, we notice that the group classification is slightly sensitive according to the variation of the majority threshold. We observe values between 0.0374 and 0.0866 (note that when the 50 alternatives take different group classifications in two adjacent simulations, *MS* takes its maximum value: 1). This means that among the 50 alternatives, 5 have different group classifications. These results are correlated with the increase of the mediator intervention. In fact, we notice that the values representing group classification using *MS* metric increases

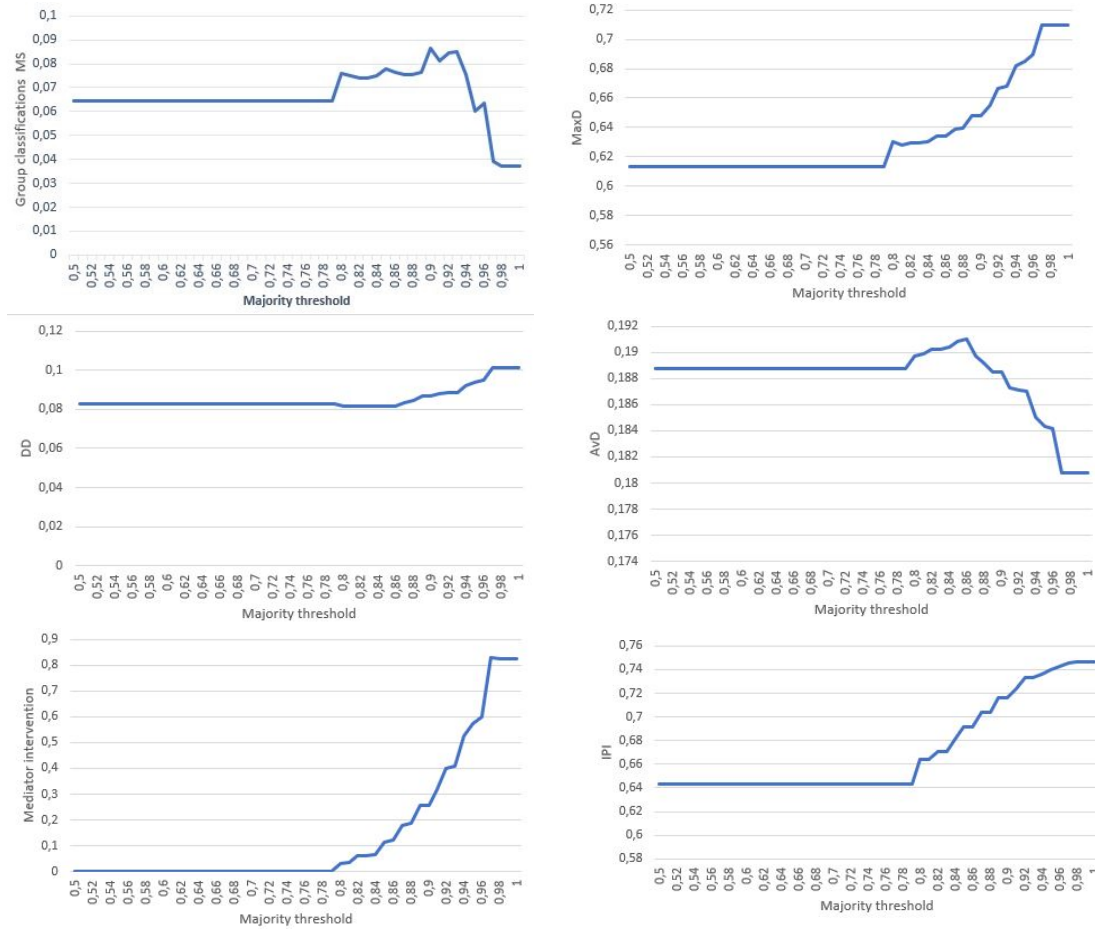


Figure 4.7 – Sensitivity analysis of group classifications based on MS metric and the variation of evaluation indicators: $MaxD$; DD ; AvD and mediator intervention (to be minimized); IPI (to be maximized), with respect to α for 15 simulation series, where the theoretical minimum is 0 and the theoretical maximum is 1

gradually until reaching the maximum value 0.0866 when $\alpha = 0.9$. At this level, the value representing the mediator's intervention is equal to 0.25 (note that the value 1, corresponds to the situation where the mediator agent classifies all the alternatives). This explains that 25% of alternatives are classified by the mediator agent based on her plurality strategy. The mediator intervention does not necessarily concern the same alternatives from one simulation to another, which increases the difference regarding the group classifications. Then, we observe that the values representing agent's final classification decreases until reaching the minimum value 0.037417 when $\alpha = 1$. At this level, the value representing the mediator intervention is equal to 0.9 which represents 90% of alternatives classified in the same way by

the mediator agent. This decreases the difference of the final agents classifications from one simulation to another.

We observe a very weak sensitivity over the evaluation indicators *MaxD*, *DD*, *AvD* and *IPI*, since there is a weak sensitivity of group classifications. We notice that the evolution of the results is monotonous. For *MaxD*, *DD* and *AvD* indicators, this is correlated with the increase of the unclassified alternatives and thus the increase of the mediator agent intervention. Hence, their values converges gradually to those obtained by applying the plurality method. We remark an increase of *IPI* values associated to the increase of α . This was not clear enough considering the *IPI* results obtained in section 4.6.2. We explain this tendency by the improvements of the utilities for the sensitivity analysis. Table 4.12 illustrates the average, min, max, standard deviation and Confidence interval of the performed sensitivity analysis regarding the process indicators as function of the majority threshold.

	Average	Min	Max	Standard deviation	Confidence interval
Group classifications	0.0687	0.0374	0.0866	0.0164	[0.0523,0.0851]
<i>MaxD</i>	0.6607	0.628	0.7099	0.0309	[0.6298,0.6916]
<i>DD</i>	0.0886	0.0813	0.1012	0.0074	[0.0812,0.096]
<i>AvD</i>	0.1869	0.1807	0.191	0.0036	[0.1833,0.1905]
Mediator intervention	0.3572	0.0333	0.8253	0.2899	[0.0673,0.6471]
<i>IPI</i>	0.7129	0.6641	0.747	0.0299	[0.683,0.7428]

Table 4.12 – The average, min, max, standard deviation and confidence interval of the performed sensitivity analysis regarding the process results are functions of the majority threshold

We notice that the fluctuation of the different process results remains weakly sensitive. This can be seen through the small confidence interval⁸, the small the distance between the maximum *Max* and the minimum *Min* values of the process indicators and the standard deviation.

The highest confidence interval is the one corresponding to the indicator *mediator intervention*, since the number of alternatives not supported by the majority of agents varies significantly from $\lambda = 0.8$ to $\lambda = 1$. For instance, the mediator's

⁸the confidence interval=[average variation-standard deviation, average variation+ standard variation]

intervention over the classification of 50 alternatives moves from the classification of 2 alternatives when $\lambda = 0.8$ to 43 alternatives when $\lambda = 1$.

Acceptance threshold & order of requests Figures 4.8 and 4.9 show respectively on the acceptance threshold and the order of request, the performed sensitivity analysis results regarding the group classification based on *MM* metric and the variation of the different evaluation indicators for 15 series of simulations. These results depend on the variation of the acceptance threshold and the order of requests.

- The variation of the acceptance threshold is generated randomly: for each agent j and a simulation i , we vary the acceptance threshold $\lambda_{i,j}$ for the agent j with a step $\pm\epsilon_i$, where $\epsilon_i \in [\epsilon_{i,min}, \epsilon_{i,max}]$, $\epsilon_{i,min} = 0$ and $\epsilon_{i,max} = 0.15$. Note that the distribution of agents' acceptance thresholds is generated as follows: 35% of agents have $\lambda \in [0.5, 0.55]$, 35% of agents have $\lambda \in]0.55, 0.65]$, 25% of agents have $\lambda \in]0.65, 0.75]$ and 5% of agents have $\lambda \in]0.75, 1]$.
- The order of requests is generated randomly for the first 14 series (using the same permutation σ described in the third point of the experiment process description). The order of requests for the last series (the 15th), is the reverse order of the 14th. This is done in order to examine the sensitivity in the extreme case where we take reverse orders.

Back to the Figures 4.8 and 4.9, it is noticeable that the variation of the acceptance threshold or the order of questions does not have any impact on the oscillation of the different process indicators. This is also highlighted through the Tables 4.13 and 4.14 displaying the average, min, max, standard deviation and confidence interval of the performed sensitivity analysis regarding the process indicators as function of acceptance threshold and order of requests.

For the performed sensitivity analysis over the group classifications using *MM* metric, we observe values between 0.0433 and 0.0573 for the variation of acceptance

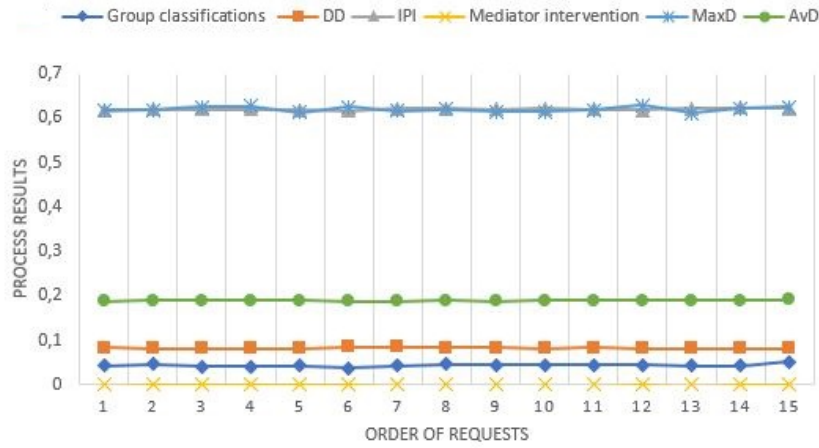


Figure 4.8 – The performed sensitivity analysis of group classifications (based on *MM* metric) and the variation of evaluation indicators: *MaxD*; *DD*; *AvD*; mediator intervention (to be maximized) and *IPI* (to be minimized) , regarding order of requests' variation for 15 simulation series, where the theoretical minimum is 0 and the theoretical maximum is 1

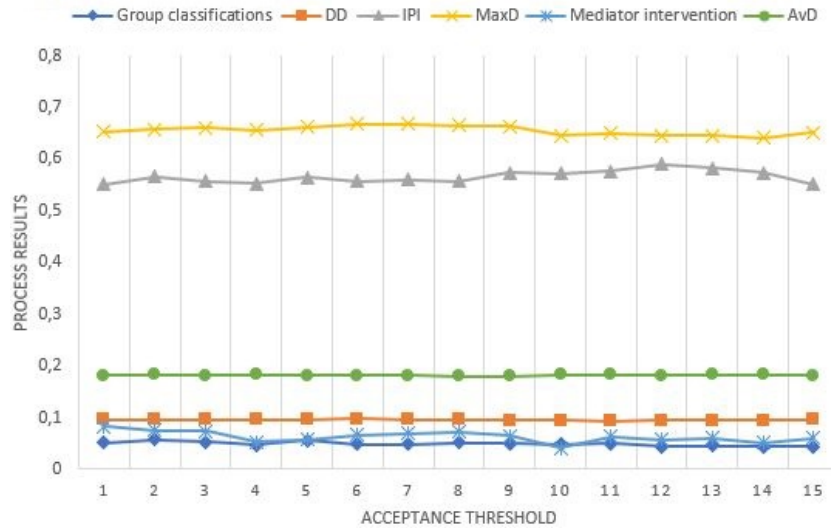


Figure 4.9 – The performed sensitivity analysis of group classifications (based on *MM* metric) and the variation of evaluation indicators: *MaxD*; *DD*; *AvD* and mediator intervention (to be minimized); *IPI* (to be maximized), regarding acceptance threshold's variation for 15 simulation series, where the theoretical minimum is 0 and the theoretical maximum is 1

threshold and between 0.038 and 0.0516 for the variation of order of requests (note that when the 50 alternatives take different group classifications in two adjacent simulations, *MM* takes its maximum value: 1). This means that among the 50 alternatives, 3 have different group classifications. Hence, a weak difference in the

	Average	Min	Max	Standard deviation	Confidence interval
Group classifications	0.0492	0.0433	0.0573	0.004	[0.0452,0.0532]
<i>MaxD</i>	0.6556	0.6413	0.6682	0.0087	[0.6469,0.6643]
<i>DD</i>	0.0956	0.0936	0.0971	0.0011	[0.0945,0.0967]
<i>AvD</i>	0.1819	0.1803	0.1834	0.0008	[0.1811,0.1827]
Mediator intervention	0.0632	0.0413	0.0826	0.0104	[0.0528,0.0736]
<i>IPi</i>	0.5658	0.5516	0.5904	0.0119	[0.5539,0.5777]

Table 4.13 – The average, min, max, standard deviation and confidence interval of the performed sensitivity analysis regarding the process results as function of acceptance threshold's

	Average	Min	Max	Standard deviation	Confidence interval
Group classifications	0.0441	0.038	0.0516	0.0031	[0.041,0.0472]
<i>MaxD</i>	0.6198	0.611	0.6283	0.0055	[0.6143,0.6253]
<i>DD</i>	0.0832	0.0818	0.0855	0.0011	[0.0821,0.0843]
<i>AvD</i>	0.1895	0.1876	0.191	0.001	[0.1885,0.1905]
Mediator intervention	0.0007	0	0.0013	0.0006	[0,0001,0.0013]
<i>IPi</i>	0.6203	0.6163	0.6234	0.0019	[0.6184,0.6222]

Table 4.14 – The average, min, max, standard deviation and Confidence interval of the performed sensitivity analysis regarding the process results regarding order of requests

final classification of agents leads to a weak variation in the evaluation indicators. This can be seen through their tight confidence interval, the small distance between max and min and the small standard deviation.

The conducted sensitivity analysis shows that the process is very slightly sensitive considering the variation of the majority threshold, order of request and acceptance threshold.

4.8 Behavioural analysis

To conduct the behavioural analysis, we focus on the agents' homogeneity. This homogeneity represents the similarity of the agents' negotiation strategies.

In fact, we aim at assessing the impact of increasing agents' inflexibility on the results of the process. To do so, we proceed as follows:

- For the first simulation series, we suppose that all agents are flexible and have a flexible strategy.
- Then, in each simulation series, we increase the percentage of inflexible agents by 10% until all the agents become inflexible. Moreover, for each simulation series, we perform 15 simulations in which we randomly generate the process parameters.

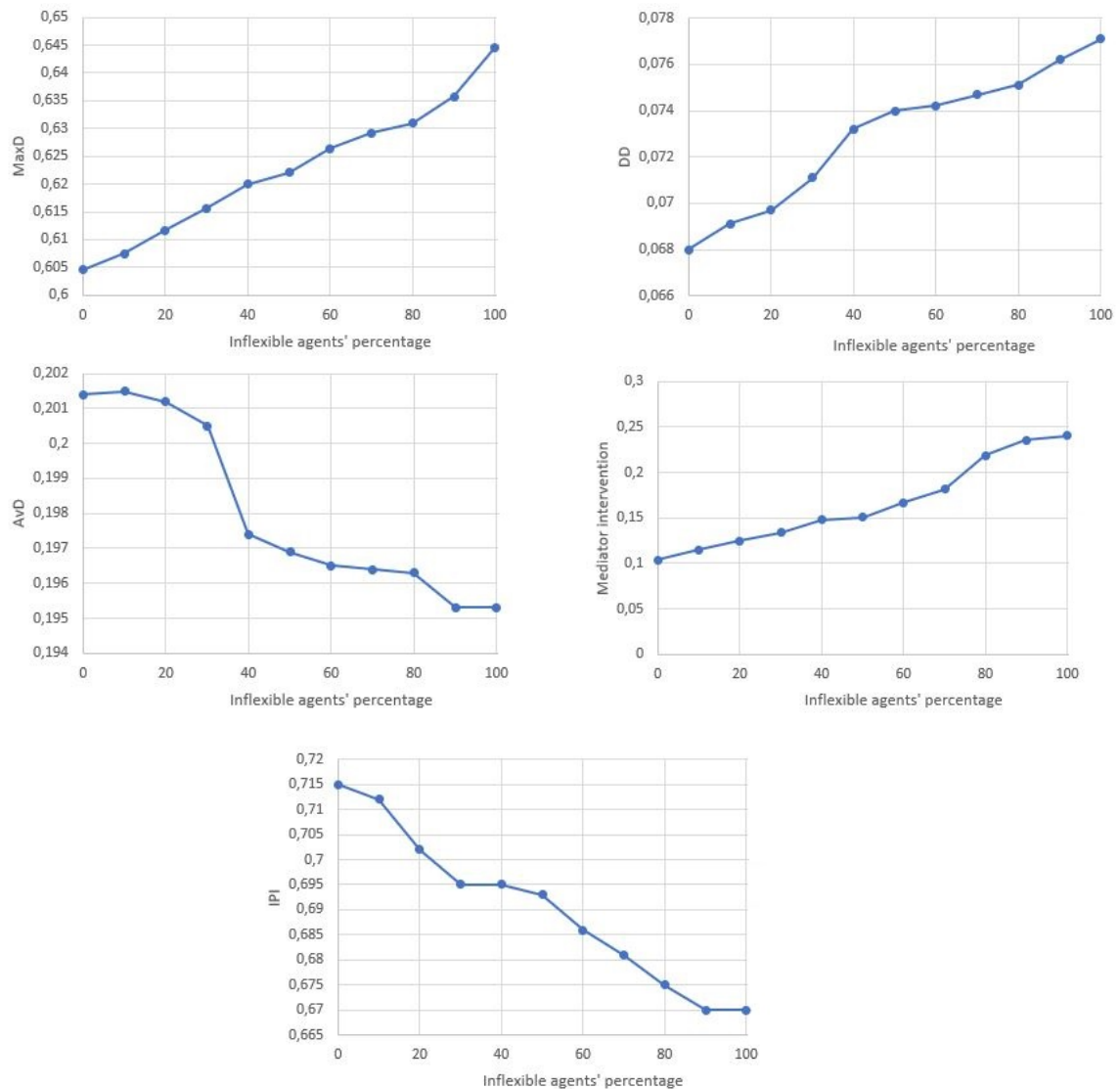


Figure 4.10 – Behavioural analysis of evaluation indicators: *MaxD*; *DD*; *AvD* and mediator intervention(to be minimized); *IPI* (to be maximized), with respect to inflexible agents' percentage for 15 simulation series, where the theoretical minimum is 0 and the theoretical maximum is 1

Figure 4.10 shows the behavioural analysis results over the different evaluation indicators ($MaxD$, AvD , IPI , DD and mediator intervention) for 15 simulations' series as function of inflexible agents' percentage.

We notice that the evolution of the results is monotonous. Not surprisingly, the higher the percentage of inflexible agents, the higher the number of alternatives not supported by the majority. This is due to the increase of mediator intervention values which leads to the convergence of the DD , $MaxD$ and AvD values towards those obtained by applying the plurality method. The decrease of IPI values is related to the acceptance strategy of non flexible agents which are less cooperative to accept alternative classifications.

Figure 4.11 shows the performed behavioural analysis results over the process' number of rounds for 15 simulations' series as function of inflexible agents' percentage. Note that the minimum number of rounds is 50 when the agents manage to rank the 50 alternatives. In case, there are still unclassified alternatives after 50 rounds, the mediator agent gives for each alternative at least a second chance to be reconsidered. Thus, the maximum number of rounds is 100.

An increase of inflexible agents leads gradually to an increase of rounds' number. Hence, the process lasts less when the process involves flexible agents.

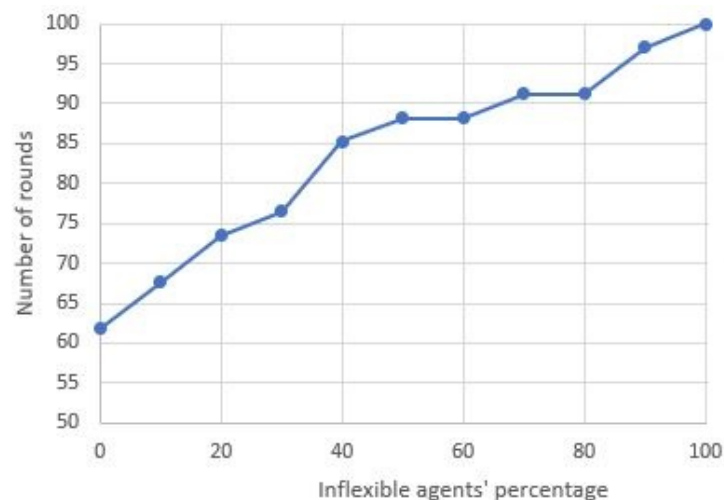


Figure 4.11 – Behavioural analysis results over the process' number of rounds for 15 simulations' series as function of inflexible agents' percentage, where the theoretical minimum is 50 and the theoretical maximum is 100

4.9 Conclusion

We proposed a multi-agent negotiation process aiming at deriving a common ordinal classification for a group of agents having individual classifications. Our approach is based on a negotiation process where each user agent makes use of her utility function. Our experiments showed that our process, when compared to centralized approaches, makes more satisfied the least satisfied agent (compared to plurality), distributes the global dissatisfaction in a fairer way and better protects the privacy of the decision makers. A sensitivity analysis was also conducted. The performed experiments show that the process is very weakly sensitive to the variation of influential parameters in the process. In future works, we plan to test our approach in larger datasets (increasing the number of alternatives and the decision makers) and with more sophisticated decision strategies (for instance, including the possibility for agents to vetoing a proposed classification). We would also like to consider incomplete or imprecise classification data, such as interval of categories, that can facilitate the task of expressing preference information. Moreover, motivated by real applications, we could relax the assumption that alternatives have the same importance. Similarly, we could model the different degree of expertise of the experts in classifying the alternatives; in particular an expert may have a high confidence in her preferred classification for an alternative, while she may have low confidence in other alternatives.

Conclusion & Perspectives

This thesis focuses on the development of an ordinal group classification process. To achieve this goal, two main phases are considered. The first phase consists in developing a methodology to interactively learn the preferences of each group member with a minimal cognitive burden and to assist her in deriving her individual classification. The second phase consists in developing an ordinal group classification from individual classifications based on a negotiation process.

We start this thesis with an introduction that highlights the objectives of our work regarding the development of such tools. Let us now outline our main contributions to achieve these objectives.

Pre-negotiation process: interactive elicitation of an ordinal classification model

The first contribution is the development of an interactive elicitation process for the parameters (weights and threshold) of MR-sort. Such process asks, at each step, the decision maker to classify an alternative and her answers (assignments) are integrated as training set for learning the model. To limit the cognitive burden of elicitation, we aim at asking informative questions in order to find a good approximation of the optimal classification in a limited number of elicitation steps. Efficient strategies are proposed to compute the next question. Such a computation can be formulated as a linear program. The proposed strategies rely on a maximum-margin optimization to evaluate the uncertainty associated to the items whose assignment is not known yet; the choice of the next question is based on evaluating the margin

a posteriori (the value of maximum-margin optimization when adding a possible response to a question). We evaluate the proposed interactive elicitation process by comparing the performance of the different strategies. Experiments show that the procedure quickly converges to the real optimal classification in very few interaction cycles.

Group-based Ordinal Classification based on a Negotiation Process

The second contribution is a negotiation-based ordinal classification process characterized by: ordinal agents' preferences, evolving preferences with respect to external factors (such as remaining time) and the possibility of not revealing full preferences, therefore providing a certain degree of privacy. Such process is based on a multilateral negotiation to reach common ordinal classifications from individual preferences. Each decision maker involved in the classification process is represented by a user agent and the process is guided by a mediator agent. Such mediator facilitates the process by avoiding interaction between agents and guarantees its success in case agents do not agree on a classification. Each user agent has a personalized preference-based behavior defined by a utility function. The emphasis on modeling the behavior of agents constitutes the originality of our multi-agent approach.

We present experiments with both real and simulated data in order to illustrate our approach and assess its performance (with respect to both user satisfaction and privacy) comparing it with two centralized methods (plurality and median methods). Our experiments showed that our process, when compared to centralized approaches, is fairer (the worst-off agent is more satisfied) and better protects user privacy. Sensitivity analysis are also conducted. The performed experiments show that the process is not sensitive considering the variation of influential parameters in the process.

Future works

The research conducted in this thesis can be further developed. Let us discuss specific adaptations of each contribution for future research.

In Chapter 3, we developed incremental preference elicitation process for MR-Sort model, assuming that category profiles have been defined a priori. Additional research could be conducted to relax this assumption. In fact, learning all the parameters of MR-Sort will be challenging, since introducing integer variables in an optimization problem requires high computing time. We also aim to carry out simulations comparing our approach to other elicitation frameworks, such as Olteanu's recent work [107], and minimax regret-based approaches [8, 9], based on real data. We intend to study the connection between our approach and the field of machine learning for many reasons such as the similarities between maximum margin and support vector machines or the similarity between incremental elicitation and active learning. Moreover, we plan to expand our process to elicitate simultaneously the preferences of several decision maker as in Teso et al. [136].

In chapter 4, we developed a negotiation process aiming at deriving group classification. The strategies reflecting agents behaviour are conducted based on defined utility function. We aim to test our approach with more sophisticated decision strategies as including the possibility for agents to veto a proposed classification. Moreover we assume that alternatives have the same importance. Further research could be performed to relax this assumption. Similarly, we could model the different degree of expertise reflecting the confidence of a decision maker over alternatives' classification.

This thesis is in the crossroads of several disciplines: multicriteria decision making; social choice; multi-agent system and automated negotiation.

Bibliography

- [1] S. Adhau, M. L. Mittal, and A. Mittal. A multi-agent system for distributed multi-project scheduling: An auction-based negotiation approach. *Engineering Applications of Artificial Intelligence*, 25(8):1738–1751, 2012.
- [2] J. Ah-Pine, B. Mayag, and A. Rolland. Identification of a 2-additive bi-capacity by using mathematical programming. In *International Conference on Algorithmic Decision Theory*, pages 15–29. Springer, 2013.
- [3] J. Almeida-Dias, J. R. Figueira, and B. Roy. Electre tri-c: A multiple criteria sorting method based on characteristic reference actions. *European Journal of Operational Research*, 204(3):565–580, 2010.
- [4] J. Almeida-Dias, J. R. Figueira, and B. Roy. A multiple criteria sorting method where each category is characterized by several reference actions: The electre tri-nc method. *European Journal of Operational Research*, 217(3):567–579, 2012.
- [5] K. J. Arrow. A difficulty in the concept of social welfare. *Journal of political economy*, 58(4):328–346, 1950.
- [6] K. J. Arrow. *Social choice and individual values*, volume 12. Yale university press, 2012.
- [7] C. Bartolini, C. Preist, and N. R. Jennings. A generic software framework for automated negotiation. In *First international conference on autonomous agent and multi-agent systems*. Citeseer, 2002.

- [8] N. Benabbou, P. Perny, and P. Viappiani. A regret-based preference elicitation approach for sorting with multicriteria reference profiles. In *From Multicriteria Decision Making to Preference Learning (DA2PL'16)*, 2016.
- [9] N. Benabbou, P. Perny, and P. Viappiani. Incremental elicitation of choquet capacities for multicriteria choice, ranking and sorting problems. *Artif. Intell.*, 246:152–180, 2017.
- [10] U. Böckenholt. Thresholds and intransitivities in pairwise judgments: A multilevel analysis. *Journal of Educational and Behavioral Statistics*, 26(3):269–282, 2001.
- [11] J. d. Borda. Mémoire sur les élections au scrutin. *Histoire de l'Académie Royale des Sciences pour 1781 (Paris, 1784)*, 1784.
- [12] D. Bouyssou and P. Vincke. Relations binaires et modélisation des préférences. *Concepts et méthodes pour l'aide à la décision*, 1, 2003.
- [13] S. Brams and P. C. Fishburn. *Approval voting*. Springer Science & Business Media, 2007.
- [14] F. Brandt, V. Conitzer, U. Endriss, J. Lang, and A. D. Procaccia. *Handbook of computational social choice*. Cambridge University Press, 2016.
- [15] I. Brigui-Chtioui, P. Caillou, and S. Pinson. Approche anytime pour l'ajustement de l'incrément dans les enchères multicritères automatisées. In *23es Journées Francophones sur les Systèmes Multi-Agents (JFSMA'15)*, pages 153–162. Cépaduès, 2015.
- [16] O. Cailloux, B. Mayag, P. Meyer, and V. Mousseau. Operational tools to build a multicriteria territorial risk scale with multiple stakeholders. *Reliability Engineering & System Safety*, 120:88–97, 2013.
- [17] O. Cailloux, P. Meyer, and V. Mousseau. Eliciting electre tri category limits for a group of decision makers. *European Journal of Operational Research*, 223(1):133–140, 2012.

- [18] N. Capuano, F. Chiclana, H. Fujita, E. Herrera-Viedma, and V. Loia. Fuzzy group decision making with incomplete information guided by social influence. *IEEE Transactions on Fuzzy Systems*, 26(3):1704–1718, 2017.
- [19] J. Castro, J. Lu, G. Zhang, Y. Dong, and L. Martínez. Opinion dynamics-based group recommender systems. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 48(12):2394–2406, 2017.
- [20] U. Chajewska, D. Koller, and R. Parr. Making rational decisions using adaptive utility elicitation. In *AAAI/IAAI*, pages 363–369, 2000.
- [21] A. Chavez and P. Maes. *Kasbah: An agent marketplace for buying and selling goods*. Practical Application, 1996.
- [22] C.-T. Chen. Extensions of the topsis for group decision-making under fuzzy environment. *Fuzzy sets and systems*, 114(1):1–9, 2000.
- [23] S. Chen and G. Weiss. An efficient automated negotiation strategy for complex environments. *Engineering Applications of Artificial Intelligence*, 26(10):2613–2623, 2013.
- [24] Y. Chevaleyre, P. E. Dunne, U. Endriss, J. Lang, M. Lemaitre, N. Maudet, J. Padget, S. Phelps, J. A. Rodríguez-Aguilar, and P. Sousa. Issues in multi-agent resource allocation. 2005.
- [25] V. Conitzer and T. Sandholm. Complexity of mechanism design. *arXiv preprint cs/0205075*, 2002.
- [26] A. H. Copeland. A reasonable social welfare function. Technical report, mimeo, 1951. University of Michigan, 1951.
- [27] S. Damart, L. C. Dias, and V. Mousseau. Supporting groups in sorting decisions: Methodology and use of a multi-criteria aggregation/disaggregation dss. *Decision Support Systems*, 43(4):1464–1475, 2007.
- [28] N. De Condorcet et al. *Essai sur l’application de l’analyse à la probabilité des décisions rendues à la pluralité des voix*. Cambridge University Press, 2014.

- [29] W. De Keyser and P. Peeters. A note on the use of promethee multicriteria methods. *European journal of operational research*, 89(3):457–461, 1996.
- [30] T. Denat and M. Öztürk. Dominance based monte carlo algorithm for preference learning in the multi-criteria sorting problem: Theoretical properties. In "*From Multiple Criteria Decision Aid to Preference Learning*" (DA2PL 2016), pages 47–52, 2016.
- [31] T. Denat and M. Öztürk. Dominance based monte carlo algorithm for preference elicitation in the multi-criteria sorting problem: Some performance tests. In *International Conference on Algorithmic Decision Theory*, pages 50–64. Springer, 2017.
- [32] J. Devaud, G. Groussaud, and E. Jacquet-Lagrece. Utadis: Une méthode de construction de fonctions d'utilité additives rendant compte de jugements globaux. *European Working Group on Multicriteria Decision Aid, Bochum*, page 94, 1980.
- [33] Y. Dong, X. Chen, and F. Herrera. Minimizing adjusted simple terms in the consensus reaching process with hesitant linguistic assessments in group decision making. *Information Sciences*, 297:95–117, 2015.
- [34] Y. Dong, Q. Zha, H. Zhang, G. Kou, H. Fujita, F. Chiclana, and E. Herrera-Viedma. Consensus reaching in social network group decision making: Research paradigms and challenges. *Knowledge-Based Systems*, 162:3–13, 2018.
- [35] M. Doumpos and C. Zopounidis. The use of the preference disaggregation analysis in the assessment of financial risks. *Fuzzy Economic Review*, 3(1):39–57, 1998.
- [36] M. Doumpos and C. Zopounidis. *Multicriteria decision aid classification methods*, volume 73. Springer Science & Business Media, 2002.
- [37] M. Doumpos and C. Zopounidis. A multicriteria decision support system for bank rating. *Decision support systems*, 50(1):55–63, 2010.

- [38] C. A. B. e Costa and J.-C. Vansnick. Macbeth—an interactive path towards the construction of cardinal value functions. *International transactions in operational Research*, 1(4):489–500, 1994.
- [39] U. Endriss. Monotonic concession protocols for multilateral negotiation. In *Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems*, pages 392–399, 2006.
- [40] U. Endriss, N. Maudet, F. Sadri, and F. Toni. On optimal outcomes of negotiations over resources. In *AAMAS*, volume 3, pages 177–184, 2003.
- [41] U. Endriss, N. Maudet, F. Sadri, and F. Toni. Negotiating socially optimal allocations of resources. *Journal of artificial intelligence research*, 25:315–348, 2006.
- [42] P. Faratin. *Automated service negotiation between autonomous computational agents*. PhD thesis, Queen Mary University of London, 2000.
- [43] P. Faratin, C. Sierra, and N. R. Jennings. Negotiation decision functions for autonomous agents. *Robotics and Autonomous Systems*, 24(3-4):159–182, 1998.
- [44] P. Faratin, C. Sierra, and N. R. Jennings. Using similarity criteria to make issue trade-offs in automated negotiations. *artificial Intelligence*, 142(2):205–237, 2002.
- [45] S. Fatima, S. Kraus, and M. Wooldridge. *Principles of automated negotiation*. Cambridge University Press, 2014.
- [46] S. S. Fatima, M. Wooldridge, and N. Jennings. An analysis of feasible solutions for multi-issue negotiation involving non-linear utility functions. 2009.
- [47] S. S. Fatima, M. Wooldridge, and N. R. Jennings. Multi-issue negotiation under time constraints. In *Proceedings of the first international joint conference on Autonomous agents and multiagent systems: part 1*, pages 143–150, 2002.
- [48] S. S. Fatima, M. J. Wooldridge, and N. R. Jennings. Multi-issue negotiation with deadlines. *Journal of Artificial Intelligence Research*, 27:381–417, 2006.

- [49] E. Fernández, J. R. Figueira, J. Navarro, and B. Roy. Electre tri-nb: A new multiple criteria ordinal classification method. *European Journal of Operational Research*, 263(1):214–224, 2017.
- [50] J. Figueira and B. Roy. Determining the weights of criteria in the electre type methods with a revised simos’ procedure. *European journal of operational research*, 139(2):317–326, 2002.
- [51] J. R. Figueira, S. Greco, B. Roy, and R. Słowiński. Electre methods: main features and recent developments. In *Handbook of multicriteria analysis*, pages 51–89. Springer, 2010.
- [52] T. Finin, Y. K. Labrou, J. Mayfield, et al. Evaluating kqml as an agent communication language. *Intelligent Agents II—Agent Theories, Architectures, and Languages*, 1037, 1996.
- [53] T. FIPA. Fipa communicative act library specification. *Change*, 2000(01/18), 2000.
- [54] P. C. Fishburn. Utility theory for decision making. Technical report, Research analysis corp McLean VA, 1970.
- [55] K. Gajos and D. S. Weld. Preference elicitation for interface optimization. In *Proceedings of the 18th annual ACM symposium on user interface software and technology*, pages 173–182, 2005.
- [56] I. Garcia and L. Sebastia. A negotiation framework for heterogeneous group recommendation. *Expert Systems with Applications*, 41(4):1245–1261, 2014.
- [57] I. Garcia, L. Sebastia, and E. Onaindia. On the design of individual and group recommender systems for tourism. *Expert systems with applications*, 38(6):7683–7692, 2011.
- [58] A. Gibbard. Manipulation of voting schemes: a general result. *Econometrica: journal of the Econometric Society*, pages 587–601, 1973.
- [59] C. Giugni. *Variabilità e mutabilità: Contributo allo studio delle distribuzioni e delle relazioni statistiche*. Cuppini, 1912.

- [60] G.-G. Granger. *mathématique sociale du marquis de Condorcet (La)*. Odile Jacob, 1989.
- [61] S. Greco, M. Ehrgott, and J. R. Figueira. *Trends in multiple criteria decision analysis*, volume 142. Springer Science & Business Media, 2010.
- [62] S. Greco, J. Figueira, and M. Ehrgott. *Multiple criteria decision analysis*. Springer, 2016.
- [63] S. Greco, B. Matarazzo, and R. Slowinski. A new rough set approach to multicriteria and multiattribute classification. In *International conference on rough sets and current trends in computing*, pages 60–67. Springer, 1998.
- [64] S. Greco, B. Matarazzo, and R. Slowinski. Rough approximation of a preference relation by dominance relations. *European Journal of operational research*, 117(1):63–83, 1999.
- [65] S. Greco, B. Matarazzo, and R. Slowinski. Rough sets theory for multicriteria decision analysis. *European journal of operational research*, 129(1):1–47, 2001.
- [66] S. Greco, V. Mousseau, and R. Słowiński. Ordinal regression revisited: multiple criteria ranking using a set of additive value functions. *European Journal of Operational Research*, 191(2):416–436, 2008.
- [67] S. Greco, R. Słowiński, and Y. Yao. Bayesian decision theory for dominance-based rough set approach. In *International Conference on Rough Sets and Knowledge Technology*, pages 134–141. Springer, 2007.
- [68] S. Guo and S. Sanner. Real-time multiattribute bayesian preference elicitation with pairwise comparison queries. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 289–296, 2010.
- [69] R. H. Guttman and P. Maes. Agent-mediated integrative negotiation for retail electronic commerce. In *International Workshop on Agent-Mediated Electronic Trading*, pages 70–90. Springer, 1998.

- [70] D. M. Hamby. A review of techniques for parameter sensitivity analysis of environmental models. *Environmental monitoring and assessment*.
- [71] M. He, H.-f. Leung, and N. R. Jennings. A fuzzy-logic based bidding strategy for autonomous agents in continuous double auctions. *IEEE Transactions on Knowledge and data Engineering*, 15(6):1345–1363, 2003.
- [72] M. Hemaissia, A. El Fallah Seghrouchni, C. Labreuche, and J. Mattioli. A multilateral multi-issue negotiation protocol. In *Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems*, pages 1–8, 2007.
- [73] T. Ito, H. Hattori, and M. Klein. Multi-issue negotiation protocol for agents: Exploring nonlinear utility spaces. In *IJCAI*, volume 7, pages 1347–1352, 2007.
- [74] K. Jabeur and J.-M. Martel. An ordinal sorting method for group decision-making. *European Journal of Operational Research*, 180(3):1272–1289, 2007.
- [75] K. Jabeur, J.-M. Martel, and S. B. Khélifa. A distance-based collective pre-order integrating the relative importance of the group’s members. *Group Decision and Negotiation*, 13(4):327–349, 2004.
- [76] E. Jacquet-Lagrèze. An application of the uta discriminant model for the evaluation of r & d projects. In *Advances in multicriteria analysis*, pages 203–211. Springer, 1995.
- [77] E. Jacquet-Lagrez and J. Siskos. Assessing a set of additive utility functions for multicriteria decision-making, the uta method. *European journal of operational research*, 10(2):151–164, 1982.
- [78] A. Jameson. More than the sum of its members: challenges for group recommender systems. In *Proceedings of the working conference on Advanced visual interfaces*, pages 48–54, 2004.
- [79] N. R. Jennings, P. Faratin, A. R. Lomuscio, S. Parsons, C. Sierra, and M. Wooldridge. Automated negotiation: prospects, methods and challenges.

- International Journal of Group Decision and Negotiation*, 10(2):199–215, 2001.
- [80] N. R. Jennings and M. Wooldridge. Applications of intelligent agents. In *Agent technology*, pages 3–28. Springer, 1998.
- [81] B. Johansson, A. Speranzon, M. Johansson, and K. H. Johansson. On decentralized negotiation of optimal consensus. *Automatica*, 44(4):1175–1179, 2008.
- [82] A. Kakas and P. Moraitis. Argumentation based decision making for autonomous agents. In *Proceedings of the second international joint conference on Autonomous agents and multiagent systems*, pages 883–890, 2003.
- [83] R. L. Keeney, H. Raiffa, and R. F. Meyer. *Decisions with multiple objectives: preferences and value trade-offs*. Cambridge university press, 1993.
- [84] J. S. Kelly. Social choice bibliography. *Social Choice and Welfare*, 8(2):97–169, 1991.
- [85] J. K. Kim, H. K. Kim, H. Y. Oh, and Y. U. Ryu. A group recommendation system for online communities. *International journal of information management*, 30(3):212–219, 2010.
- [86] M. Klein, P. Faratin, H. Sayama, and Y. Bar-Yam. Negotiating complex contracts. *Group Decision and Negotiation*, 12(2):111–125, 2003.
- [87] G. H. Kramer. A dynamical model of political equilibrium. *Journal of Economic Theory*, 16(2):310–334, 1977.
- [88] S. Kraus. Negotiation and cooperation in multi-agent environments. *Artificial intelligence*, 94(1-2):79–97, 1997.
- [89] S. Kraus. Automated negotiation and decision making in multiagent environments. In *ECCAI Advanced course on artificial intelligence*, pages 150–172. Springer, 2001.

- [90] S. Kraus, N. Nirke, and K. Sycara. Reaching agreements through argumentation. In *Proceedings of the Twelfth International Workshop on Distributed Artificial Intelligence*, pages 233–247, 1993.
- [91] S. Kraus, K. Sycara, and A. Evenchik. Reaching agreements through argumentation: a logical model and implementation. *Artificial Intelligence*, 104(1-2):1–69, 1998.
- [92] A. Leroy, V. Mousseau, and M. Pirlot. Learning the parameters of a multiple criteria sorting method. In *International Conference on Algorithmic Decision Theory*, pages 219–233. Springer, 2011.
- [93] A. Leroy, V. Mousseau, and M. Pirlot. Learning the parameters of a multiple criteria sorting method. In *Algorithmic Decision Theory - Second International Conference, ADT 2011, Piscataway, NJ, USA, October 26-28, 2011. Proceedings*, pages 219–233, 2011.
- [94] A. R. Lomuscio, M. Wooldridge, and N. R. Jennings. A classification scheme for negotiation in electronic commerce. In *Agent Mediated Electronic Commerce*, pages 19–33. Springer, 2001.
- [95] R. D. Luce. Semiorders and a theory of utility discrimination. *Econometrica, Journal of the Econometric Society*, pages 178–191, 1956.
- [96] R. Mailler, V. Lesser, and B. Horling. Cooperative negotiation for soft real-time distributed resource allocation. In *Proceedings of the second international joint conference on Autonomous agents and multiagent systems*, pages 576–583, 2003.
- [97] M. J. A. N. C. Marquis. *Essai sur l'application de l'analyse à la probabilité des décisions rendues à la pluralité des voix*. Chelsea, 1785.
- [98] J. Masthoff. Group recommender systems: Combining individual models. In *Recommender systems handbook*, pages 677–702. Springer, 2011.

- [99] N. Maudet, S. Parsons, and I. Rahwan. Argumentation in multi-agent systems: Context and recent developments. In *International Workshop on Argumentation in Multi-Agent Systems*, pages 1–16. Springer, 2006.
- [100] J. F. McCarthy and T. D. Anagnost. Musicfx: an arbiter of group preferences for computer supported collaborative workouts. In *Proceedings of the 1998 ACM conference on Computer supported cooperative work*, pages 363–372, 1998.
- [101] K. McCarthy, L. McGinty, B. Smyth, and M. Salamó. The needs of the many: a case-based group recommender system. In *European Conference on Case-Based Reasoning*, pages 196–210. Springer, 2006.
- [102] V. Mousseau. *Problèmes liés à l'évaluation de l'importance relative des critères en aide multicritère à la décision: Réflexions théoriques, expérimentations et implémentations informatiques*. PhD thesis, Paris 9, 1993.
- [103] V. Mousseau, J. Figueira, and J.-P. Naux. Using assignment examples to infer weights for electre tri method: Some experimental results. *European Journal of Operational Research*, 130(2):263–275, 2001.
- [104] V. Mousseau and R. Slowinski. Inferring an electre tri model from assignment examples. *Journal of global optimization*, 12(2):157–174, 1998.
- [105] O. Nefla, M. Öztürk, P. Viappiani, and I. Brigui-Chtioui. Interactive elicitation of a majority rule sorting model with maximum margin optimization. In *International Conference on Algorithmic Decision Theory*, pages 141–157. Springer, 2019.
- [106] M. Neuman . *Theory of games and economic behaviour*, 1944.
- [107] A. L. Olteanu. Strategies for the incremental inference of majority-rule sorting models. *Proceeding from multiple criteria Decision aid to Preference Learning (DA2PL)*, 2018.
- [108] M. J. Osborne and A. Rubinstein. *A course in game theory*. MIT press, 1994.

- [109] M. Öztürk, A. Tsoukiàs, and P. Vincke. Preference modelling. In *Multiple criteria decision analysis: State of the art surveys*, pages 27–59. Springer, 2005.
- [110] I. J. Pérez, F. J. Cabrerizo, S. Alonso, Y. Dong, F. Chiclana, and E. Herrera-Viedma. On dynamic consensus processes in group decision making problems. *Information Sciences*, 459:20–35, 2018.
- [111] I. Rahwan, R. Kowalczyk, and H. H. Pham. Intelligent agents for automated one-to-many e-commerce negotiation. *Computer Science Preprint Archive*, 2002(7):782–788, 2002.
- [112] I. Rahwan, S. D. Ramchurn, N. R. Jennings, P. McBurney, S. Parsons, and L. Sonenberg. Argumentation-based negotiation. *Knowledge engineering review*, 18(4):343–375, 2003.
- [113] T. Rahwan and N. R. Jennings. Distributing coalitional value calculations among cooperating agents. 2005.
- [114] F. Ren and M. Zhang. Bilateral single-issue negotiation model considering nonlinear utility and time constraint. *Decision Support Systems*, 60:29–38, 2014.
- [115] V. Robu, D. Somefun, and J. A. La Poutré. Modeling complex multi-issue negotiations using utility graphs. In *Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems*, pages 280–287, 2005.
- [116] J. S. Rosenschein and G. Zlotkin. Designing conventions for automated negotiation. *AI magazine*, 15(3):29–29, 1994.
- [117] B. Roy. *Méthodologie multicritère d’aide à la décision*. Economica, 1985.
- [118] B. Roy and D. Bouyssou. *Aide multicritère à la décision: méthodes et cas*. Economica Paris, 1993.

- [119] B. Roy and V. Mousseau. A theoretical framework for analysing the notion of relative importance of criteria. *Journal of Multi-Criteria Decision Analysis*, 5(2):145–159, 1996.
- [120] A. Rubinstein. Perfect equilibrium in a bargaining model. *Econometrica: Journal of the Econometric Society*, pages 97–109, 1982.
- [121] T. Sandholm. An implementation of the contract net protocol based on marginal cost calculations. In *AAAI*, volume 93, pages 256–262, 1993.
- [122] T. Sandholm. Distributed rational decision making, multiagent systems: A modern introduction to distributed artificial intelligence, weib, g, 1999.
- [123] M. A. Satterthwaite. Strategy-proofness and arrow’s conditions: Existence and correspondence theorems for voting procedures and social welfare functions. *Journal of economic theory*, 10(2):187–217, 1975.
- [124] C. Sierra, P. Faratin, and N. R. Jennings. A service-oriented negotiation model between autonomous agents. In *European Workshop on Modelling Autonomous Agents in a Multi-Agent World*, pages 17–35. Springer, 1997.
- [125] C. Sierra, N. R. Jennings, P. Noriega, and S. Parsons. A framework for argumentation-based negotiation. In *International Workshop on Agent Theories, Architectures, and Languages*, pages 177–192. Springer, 1997.
- [126] J. Simos. L’évaluation environnementale. Technical report, EPFL, 1989.
- [127] J. Simos. Évaluer l’impact sur l’environnement. une approche originale par l’analyse multicritère et la négociation [environmental impact assessment. an original approach for multi-criteria analysis and negociation]. *Géographie physique et Quaternaire*, 1990.
- [128] P. B. Simpson. On defining areas of voter choice: Professor tullock on stable voting. *The Quarterly Journal of Economics*, 83(3):478–490, 1969.
- [129] R. G. Smith. The contract net protocol: High-level communication and control in a distributed problem solver. In *Readings in distributed artificial intelligence*, pages 357–366. Elsevier, 1988.

- [130] O. Sobrie, V. Mousseau, and M. Pirlot. Learning a majority rule model from large sets of assignment examples. In *International Conference on Algorithmic Decision Theory*, pages 336–350. Springer, 2013.
- [131] O. Sobrie, V. Mousseau, and M. Pirlot. Learning the parameters of a non compensatory sorting model. In *International Conference on Algorithmic Decision Theory*, pages 153–170. Springer, 2015.
- [132] O. Sobrie, V. Mousseau, and M. Pirlot. A population-based algorithm for learning a majority rule sorting model with coalitional veto. In *International Conference on Evolutionary Multi-Criterion Optimization*, pages 575–589. Springer, 2017.
- [133] O. Sobrie, V. Mousseau, and M. Pirlot. Learning monotone preferences using a majority rule sorting model. *International Transactions in Operational Research*, 26(5):1786–1809, 2019.
- [134] E. Szmidt and J. Kacprzyk. A consensus-reaching process under intuitionistic fuzzy preference relations. *International Journal of Intelligent Systems*, 18(7):837–852, 2003.
- [135] S. Teso, A. Passerini, and P. Viappiani. Constructive preference elicitation by setwise max-margin learning. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*, pages 2067–2073, 2016.
- [136] S. Teso, A. Passerini, and P. Viappiani. Constructive preference elicitation for multiple users with setwise max-margin. In *International Conference on Algorithmic Decision Theory*, pages 3–17. Springer, 2017.
- [137] A. Tsoukiàs. On the concept of decision aiding process: an operational perspective. *Annals of Operations Research*, 154(1):3–27, 2007.
- [138] A. Tsoukiàs. From decision theory to decision aiding methodology. *European journal of operational research*, 187(1):138–161, 2008.

- [139] A. Tsoukias, P. Perny, and P. Vincke. From concordance/discordance to the modelling of positive and negative reasons in decision aiding. In *Aiding decisions with multiple criteria*, pages 147–174. Springer, 2002.
- [140] R. Ureña, F. J. Cabrerizo, J. A. Morente-Molinera, and E. Herrera-Viedma. Gdm-r: A new framework in r to support fuzzy group decision making processes. *Information Sciences*, 357:161–181, 2016.
- [141] R. Urena, F. Chiclana, G. Melancon, and E. Herrera Viedma. A social network based approach for consensus achievement in multi-person decision making. *Information Fusion*, 47:72–87, 2019.
- [142] R. Ureña, F. Chiclana, J. A. Morente-Molinera, and E. Herrera-Viedma. Managing incomplete preference relations in decision making: a re-view and future trends. *Information Sciences*, 302:14–32, 2015.
- [143] P. Viappiani and C. Boutilier. Optimal bayesian recommendation sets and myopically optimal choice query sets. In *Advances in neural information processing systems*, pages 2352–2360, 2010.
- [144] C. Villavicencio, S. Schiaffino, J. A. Diaz-Pace, and A. Monteserin. Group recommender systems: A multi-agent solution. *Knowledge-Based Systems*, 164:436–458, 2019.
- [145] N. Vulkan and N. R. Jennings. Efficient mechanisms for the supply of services in multi-agent environments. In *Proceedings of the first international conference on Information and computation economies*, pages 1–10, 1998.
- [146] C. C. White, A. P. Sage, and S. Dozono. A model of multiattribute decisionmaking and trade-off weight determination under uncertainty. *IEEE Transactions on Systems, Man, and Cybernetics*, (2):223–229, 1984.
- [147] M. Wooldridge. *An introduction to multiagent systems*. John Wiley & Sons, 2009.
- [148] W. Yu. *Aide multicritère à la décision dans le cadre de la problématique du tri: concepts, méthodes et applications*. PhD thesis, Paris 9, 1992.

-
- [149] F. Zeuthen. *Problems of monopoly and economic warfare*, volume 25. Routledge, 2018.
 - [150] X. Zheng, P. Martin, and K. Brohman. Cloud service negotiation: Concession vs. tradeoff approaches. In *2012 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (ccgrid 2012)*, pages 515–522. IEEE, 2012.
 - [151] C. Zopounidis and M. Doumpos. Business failure prediction using the utadis multicriteria analysis method. *Journal of the Operational research Society*, 50(11):1138–1148, 1999.
 - [152] C. Zopounidis and M. Doumpos. A multicriteria decision aid methodology for sorting decision problems: The case of financial distress. *Computational Economics*, 14(3):197–218, 1999.

RÉSUMÉ

Dans cette thèse, nous développons un processus de classification ordinale pour l'aide à la prise de décision de groupe. Pour ce faire, nous avons développé un processus à deux phases. La première phase de pré-négociation consiste à assister un décideur unique dans son problème de classification ordinale multicritère en se basant sur un nouveau processus d'élicitation dite "incrémentale". Ce processus sélectionne les questions les plus informatives à poser au décideur dans le but d'apprendre ses préférences. La deuxième phase de négociation vise à faire converger plusieurs décideurs vers une classification de groupe basée sur leurs classifications individuelles (dérivées suite à la phase de pré-négociation). Le processus est fondé sur un système multi-agents afin de représenter les décideurs humains et de conduire le processus de manière distribuée et autonome. Ce processus vise à satisfaire équitablement tous les décideurs impliqués en tenant compte de l'évolution de leurs comportements et préférences et en optimisant la confidentialité de leurs classifications individuelles.

MOTS CLÉS

Aide à la décision multicritère, Décision de groupe, Elicitation des préférences, Classification ordinale, Elicitation incrémentale, Systèmes multi agents, Négotiation automatisée

ABSTRACT

In this thesis, we develop an ordinal classification process for group decision aiding. For this aim, we have developed a two-phase process. The first pre-negotiation phase consists in assisting a single decision maker in her multi-criteria ordinal classification problem based on a new "incremental" elicitation process. This process selects the most informative questions to ask the decision maker in order to learn her preferences. The second negotiation phase aims at converging several decision makers towards a group classification based on their individual classifications (derived from the pre-negotiation phase). The process is based on a multi agent system to represent the human decision makers and to conduct the process in a distributed and autonomous way. This process aims at satisfying fairly all the decision makers involved by taking into account the evolution of their behaviors and preferences and by optimizing the disclosure of their individual classifications.

KEYWORDS

Multicriteria decision aiding, Group decision aiding, Preference elicitation, Ordinal classification, Incremental elicitation, Multi agent systems, Automated negotiation