Università degli Studi di Salerno

Dottorato di Ricerca in Informatica e Ingegneria dell'Informazione
Ciclo 30 – a.a. 2016/2017

TESI DI DOTTORATO / PH.D. THESIS

Fuzzy Models for Group Decision Making and Applications to e-Learning and Recommender Systems

NICOLA CAPUANO

SUPERVISOR: **PROF. VINCENZO LOIA**

PH.D. PROGRAM DIRECTOR: **PROF. PASQUALE CHIACCHIO**

Dipartimento di Ingegneria dell'Informazione ed Elettrica
e Matematica Applicata
Dipartimento di Informatica

As far as the laws of mathematics refer to
reality, they are not certain; and as far as they
are certain, they do not refer to reality.
(Albert Einstein)

The Guide is definitive.
Reality is frequently inaccurate.
(Douglas Adams)

Abstract

The work presented in this Ph.D. thesis deals with the definition of new fuzzy models for Group Decision Making (GDM) aimed at improving two phases of the decision process: preferences expression and aggregation. In particular a new preferences model named Fuzzy Ranking has been defined to help decision makers express fuzzy statements on available alternatives in a simple and meaningful form, focusing on two alternatives at a time but, at the same time, without losing the global picture. This allows to reduce inconsistencies with respect to other existing models.

Moreover a new preference aggregation model guided by social influence has been described. During a GDM process, in fact, decision makers interact and discuss each other exchanging opinions and information. Often, in these interactions, those with wider experience, knowledge and persuasive ability are capable of influencing the others fostering a change in their views. So, social influence plays a key role in the decision process but, differently from other aspects, very few attempts to formalize its contribution in preference aggregation and consensus reaching have been made till now.

In order to validate the defined models, they have been instantiated in two application contexts: e-Learning and Recommender Systems. In the first context, they have been applied to the peer assessment problem in massive online courses. In such courses, the huge number of participants prevents their thorough evaluation by the teachers. A feasible approach to tackle this issue is peer assessment, in which students also play the role of assessor for assignments submitted by others. But students are unreliable graders so peer assessment often provides inaccurate results. By leveraging on defined GDM models, a new peer assessment model aimed at improving the estimations of student grades has been proposed.

With respect to Recommender Systems, the group recommendation issue has been tackled. Instead of generating recommendations fitting individual users, Group Recommender Systems provide recommendations targeted to groups of users taking into account the preferences of any (or the majority of) group members together. The majority of existing approaches for group recommendations are based on the aggregation of either the preferences or the recommendations generated for individual group members. Customizing the defined GDM models, a new model for group recommendations has been proposed that also takes into account the personality of group members, their interpersonal trust and social influence.

The defined models have been experimented with synthetic data to show how they operate and demonstrate their properties. Once instantiated in the defined application contexts, they have been experimented with real data to measure their performance in comparison to other context-specific methods. The obtained results are encouraging and, in most cases, better than those achieved by competitor methods.

Table of Contents

PART 2: Applications

Introduction

Everyone's life is full of alternatives. In fact, from the early days of life to a venerable age, from morning awakening to nightly sleeping, a person needs to make decisions of some sort. So, decision making can be considered one of the most important and frequent human activities. It includes information gathering as well as data mining, modelling, and analysis. It requires formal calculus and subjective attitudes and may take different forms according to situations and circumstances.

One of the most complex decision making structures arises when several persons are involved in the decision process. This is known as Group Decision Making (GDM). GDM has been widely studied since it has applications in many fields. Several models and tools have been proposed for supporting this process in each of its steps, from the expression of the decision makers' opinions to their aggregation, from the selection of a feasible alternative to the achievement of the consensus on it. Among such approaches, those based on the *Fuzzy Sets Theory*, have shown to be the most effective to deal with the intrinsic uncertainty and imprecision of human judgments.

Nevertheless, fuzzy GDM models are not free of defects. In particular, the way decision makers express their preferences is often complex and requires to specify the degree to which each alternative is preferable to each other. This may result difficult and time-consuming and is likely to introduce errors and inconsistencies impacting the whole decision process. Moreover, even if several methods exist to integrate preferences expressed by decision makers, few attempts have been done till now to also consider social elements affecting the decision process like personality, interpersonal trust and influence.

To overcome these issues, in this Ph.D. thesis, new fuzzy models for GDM affecting both preferences expression and aggregation have been defined and experimented within two application contexts: e-Learning and Recommender Systems. The thesis is organized in two parts: in part 1 (chapters 1-3), the proposed models are defined and experimented in-silico to demonstrate their properties; in part 2 (chapters 4-6), the defined models are instantiated in the selected contexts and experimented to measure their performance, also in comparison with other context-specific methods.

In particular, *chapter 1* introduces the main concepts related to decision making, GDM and Fuzzy Sets, that are pre-requisite for the definition of the original models and methods described subsequently. The application of fuzzy sets to GDM is discussed and the most diffused fuzzy models and methods to represent and aggregate decision makers' preferences, rank the problem alternatives and identify the best solution are introduced. Fuzzy-based methods for the management of incomplete information are also described.

In *chapter 2*, the original *Fuzzy Ranking* model for preference elicitation is defined and compared with related work. After having deepened the classical ordinal ranking model, the proposed model is described as a fuzzy extension of the ordinal one. Conversion algorithms from fuzzy rankings to fuzzy preference relations and backward are then defined as well as similarity measures evaluating the concordance between experts' opinion.

In *chapter 3* a *Social Influence-Guided* GDM model, able to manage the effects of social influence in GDM, is defined. The model estimates the level of social influence basing on interpersonal trust with the assumption that, the more a decision maker trusts another, the more her opinion is influenced by him. After having introduced background concepts on social influence and related theories, the proposed model is outlined and described in each step. The advantages with respect to other existing models are then presented as well as the results of an in silico experiment.

In *chapter 4*, the application of the defined models to peer assessment in standard and massive *e-Learning* contexts is discussed. The peer assessment problem is described and formalized, existing approaches, aimed at improving peer assessment reliability are outlined and performance measures capable of establishing and comparing the goodness of different approaches are defined. Then, a new approach based on the instantiation on the defined GDM models is presented and compared with other existing methods.

In *chapter 5*, the application of the defined GDM models to the group recommendation problem (in the domain of *Recommender Systems*) is shown. After having introduced the most diffused approaches to individual and group recommendation, an original influence-based approach, based on the defined models, is defined and compared with related work. When generating group recommendations, the proposed model is able to take into account not only individual preferences (like most competitor methods) but also social elements like the personality of group members, their influence and mutual relationships.

In *chapter 6*, a set of experiments aimed at measuring the performance of the original peer assessment and group recommendation methods defined in chapters 4 and 5 are presented and compared with those obtained by other methods from the respective fields. Large-scale experiments with synthetic realistic data as well as small-scale experiments with real data have been performed. Results obtained, in both contexts, are encouraging and proposed methods, in most cases, outperform competitor methods.

Eventually, conclusions and on-going work are summarized.

Part 1

Fuzzy Models for Group Decision Making

Chapter 1

Background on Group Decision Making

This chapter presents the basic concepts of Decision Making (DM) and Group Decision Making (GDM), which are the basis for the definition of the original models and methods described later on. Then, essential notions on Fuzzy Sets are outlined and their application in a GDM process, as a way to deal with the inherent uncertainty and imprecision of human judgments, is discussed. To this end, the most diffused fuzzy models and methods able to represent and aggregate decision makers' preferences, to rank problem alternatives and to identify the best solution are described. Eventually, existing fuzzy-based methods that support incomplete and incoherent information processing in GDM are introduced.

1.1 Decision Making

Decision Making (DM) is a problem-solving activity aimed at the selection of a belief or a course of action among several alternatives. The typical DM process consists in the evaluation of the existing alternatives and the choice of the most satisfactory one, taking into account all the factors and contradictory requirements and according to the preferences of the decision-maker. It is therefore a process that can be more or less rational and based on explicit or implicit knowledge.

Any person makes decisions each and every day, often in an automatic and subconscious way. Some of these decisions are relatively small, such as deciding what to wear or what to have for lunch. Others are big and can have a major influence on the course of our life, such as deciding where to go to school or whether to have children. Some decisions take time while others must be made in a split-second.

If we look at organizations, we see that any of them has its goals and achieves them through the use of resources such as people, material, money, and the performance of managerial functions such as planning, organizing, directing, and controlling. To carry out these functions, managers, at various levels, are engaged in a continuous DM process related to problems that can concern aspects of logistics management, customer relationship, marketing, production planning, etc. [1].

Making a correct decision is not always easy. In many cases, in fact, the identified alternatives are related to complex situations that may have several factors of uncertainty like [2]:

- impossibility or inexpediency of obtaining sufficient amounts of reliable information;
- lack of reliable predictions of the characteristics and behavior of complex systems that reflect their response to external and internal actions;
- poorly defined goals and constraints in the project, planning, operation, and control tasks;
- impossibility of formalizing a number of factors and criteria.

In [1], the authors recognize that the DM process within organizations is even more complex today than in the past. This is explained by several factors like the availability of huge amount of information that fosters the generation of more and more alternatives; the amplified cost of making errors thanks to the complexity of operations, automation, and the chain reaction that an error can cause in many parts of the organization; the rapid changes in the environment that introduce uncertainty and require decisions to be made quickly. These reasons justify the requirement for increasing technical and methodological support to help DM.

A typical DM process can be split in four phases as shown in Figure 1 [3]. In the *intelligence phase* the reality is examined, the problem is identified, its limiting factors are analyzed and the problem statement is defined. In the *design phase*, a simplified model that represents the fragment of reality under examination is constructed and validated, and potential alternative solutions are identified. In the *choice phase* the identified alternatives are analyzed and a solution to the problem is proposed and tested to determine its viability. In the *implementation phase* the proposed solution is adopted. At every step it is possible to return to an earlier phase to refine the intermediate outcomes basing on their validation.

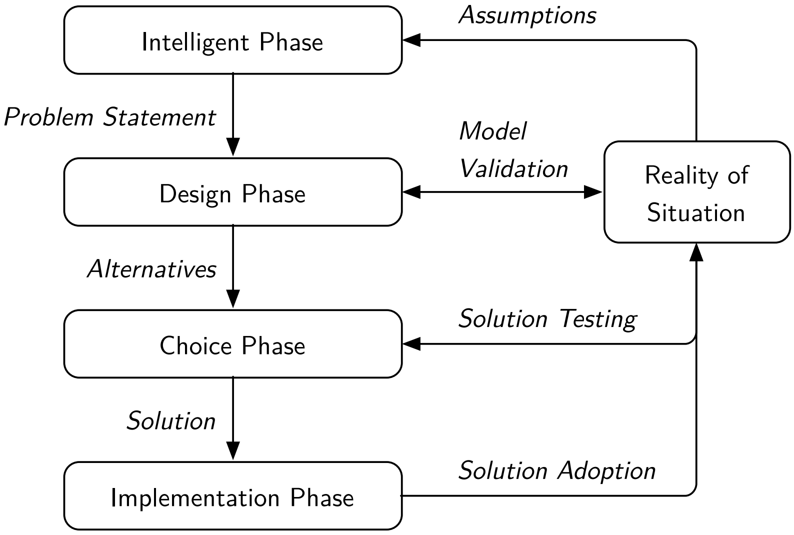

Figure 1. Steps of a DM Process

According to several authors [1, 2, 3] DM problems can be classified based on their structure. In *structured problems*, involved entities and relationships are convincingly established so that they can be numerically estimated. Such problems can be described and analyzed through standard mathematical models and methods coming from the fields of operational research, business analytics, simulation, statistics, forecasting, mathematical optimization, etc. Such problems are also referred to be “quantitatively formulated”.

Conversely, in *unstructured problems*, only the description of the most important entities is available while quantitative relationships between them are not known. These problems are also known as “qualitatively expressed” and cannot be described and analyzed through standard models and methods. Typical unstructured organizational problems include planning new services, hiring an executive, initiate a research and development project, etc..

According to [2, 4], a feasible (and sometimes the only possible) way to address this class of problems is to rely on the formulation of subjective estimates carried out by decision-makers (thus based on their own ideas on the efficiency of possible alternatives and importance of diverse criteria) and on the definition of the corresponding preferences. The heterogeneous and qualitative parameters of the problem can be so combined into a unique model, which permits alternatives to be evaluated.

The assumption is that experienced managers perceive, in a broad and well-informed manner, how many personal and subjective considerations they have to bring into the DM process. On the other hand, successes and failures of the majority of decisions can be judged by people on the basis of their subjective preferences.

In the middle between structured and unstructured problems, there are *semi-structured* problems having both quantitative and qualitative elements. Solving them involves a combination of traditional analytical models with models based on subjective preferences. Unstructured and semi-structured DM problems are also called *ill-structured.*

1.2 Group Decision Making

Decisions can be made by individuals or groups. While individual decisions are often made at lower managerial levels and in small organizations, group decisions are usually made at high managerial levels and large organizations. The DM process in which there is more than one individual involved is named Group Decision Making (GDM).

GDM is particularly useful when decisions require multiple perspectives and different areas of expertise. The main advantages of GDM, if compared to standard GM, can be summarized as follows [5]:

- more intellectual resources are gathered to support the decision including individual competencies, intuition, and knowledge;
- the work related to acquiring and processing the amount of available information can be distributed among group members;
- if the group exhibits divergent interests, the final decision tends to be more representative of the needs of the organization.

GDM can be cooperative or non-cooperative. In *cooperative* GDM all the members, each with their own knowledge, ideas, experience and motivation, are supposed to work together to achieve a common decision for which they will share the responsibility. Conversely, in *non-cooperative* GDM (otherwise known as *non-cooperative multi member DM*), the group members play the role of antagonists over some interest for which they must negotiate.

As in cooperative GDM the members share responsibility for the decision (and may also participate in its implementation), it is important to assure that each member is satisfied with it. For this reason, the ideal condition to terminate a GDM process is the achievement of a unanimous solution. In absence of unanimity, the most satisfactory alternative for the group should be selected. The most common approaches to find it are [1]:

- the group decision is made by the group leader after having discussed with the other group members (authority rule);
- the group decision is made by selecting the alternative that is preferred by the majority of members (majority rule);
- the group decision is made by repeatedly eliminating the most unpopular alternative until just one is left (negative minority rule);
- the group decision is constructed by combining ranking or scores provided by members, individually, for each alternative (ranking rule);
- the group decision is constructed to minimize the group discordance so that no member is extremely dissatisfied (soft consensus rule) [6].

As a matter of fact, the commitment to the implementation of a given solution strictly depends upon the level of consensus achieved by the group. According to this principle, a decision imposed by a dominant portion of the group has to be considered worse than a decision obtained achieving genuine consensus. The most common reasons for discordance among the group members can be summarized as follows [2]:

- although the group members are supposed to share the primary goal (i.e. to find the solution which most benefits the organization), they can have hidden or just partially shared secondary goals (e.g. to meet the priorities and needs of their respective departments);
- each member may have a distinct perception of the problem and intuition which may be hard to formalize and share to the group;
- each member may have access to different profiles of information, certain members may also have privileged access to restricted information.

These factors can be mitigated by promoting discussions and sharing all relevant information pertaining to the decision. However, even when this happens, there are other factors that can adversely affect the decision process like the need to obtain a solution rapidly or the pressure of concordant majorities on the other decision makers. Both factors are reflected in the group's tendency to prematurely converge on sub-optimal solutions [7, 8].

Some authors [1, 2, 9] stress the importance of including a *moderator* to support the GDM process. The moderator defines the process rules, assigns the tasks of each member, selects the appropriate technology, develops the schedule to be accomplished, identifies controversial opinions, promotes the discussion on them and verifies the reached level of concordance [10]. As shown in [11], the participation of a moderator, which may be human or automated, very often results in better outcomes.

Various types of *uncertain factors* are commonly met in GDM problems that may be related to the nature of the problem, the possible alternatives of the decision and the potential outcomes [12]. Uncertainty has often been associated to the gap between the information available and the information that decision makers would like to have [13] and may derive from incomplete or overwhelming information as well as from poor understanding. According to [2], such factors should be taken into account when defining mathematical models supporting GDM processes in order to increase the credibility and the factual efficiency of the decisions.

The first attempts made in this direction were based on *probability theory* [14, 15] but, in more recent works, some researchers criticized the validity of these approaches. In particular, in [16] it was pointed out that *similar to the solution of problems on the basis of deterministic methods, when we assume exact knowledge of the information, which usually does not correspond to reality, the application of probabilistic methods also supposes exact knowledge of the distribution laws and their parameters, which does not always correspond to the real possibilities of obtaining the entire spectrum of the probabilistic description.*

Alternative and more recent approaches able to deal with uncertainty in GMD rely, instead, on the *fuzzy set theory* established by Zadeh in 1965 [17]. According to [2], the application of such theory in GDM *opens an interesting avenue of giving up "excessive" precision, which is inherent in the traditional modeling approaches, while preserving reasonable rigor.* Following these considerations, the novel approaches defined in this Ph.D. thesis are precisely based on the fuzzy set theory. In order to provide a suitable background for appreciating them, an introduction to the basic concepts of such theory is given in the next sub-section.

1.3 Preliminaries on Fuzzy Sets

Fuzzy sets were introduced in [17] as an extension of classical sets. While in a classical (crisp) sets, each element can either belong to or not belong to a set, fuzzy sets allow various degrees of membership of an element to a set, ranging from 0 (no membership) to 1 (full membership). More formally, if X is a collection of objects, a fuzzy set A defined in X is a set of ordered pairs:

$$A = \{\big(x, \mu_A(x)\big) \mid x \in X\} \tag{1}$$

where $\mu_A(x)$, called *membership function*, maps X to the membership space [0,1]. According to this definition, a crisp set A of X can also be viewed as a fuzzy set in X with a membership function:

$$\mu_A(x) = \begin{cases} 1 & \text{if } x \in A \\ 0 & \text{if } x \notin A \end{cases} \tag{2}$$

The *support* of a fuzzy set A, denoted by $\mathrm{supp}(A)$, is defined as the crisp set $\mathrm{supp}(A) = \{x \in X \mid \mu_A(x) > 0\}$. The *height* of a fuzzy set A, denoted by $\mathrm{hgt}(A)$ is defined as:

$$\mathrm{hgt}(A) = \sup_{x \in X} \mu_A(x). \tag{3}$$

If $\mathrm{hgt}(A) = 1$ then A is said *normal.* A fuzzy set A is *empty*, denoted by $\emptyset$, if $\mu_A(x) = 0$ for any $x \in X$.

A fuzzy set A is called *subset* of a fuzzy set B, denoted by $A \subset B$, if $\mu_A(x) \leq \mu_B(x)$ for any $x \in X$. If $A \subset B$ and $B \subset A$ then A and B are called *equal*, denoted by $A = B$. The *union* of two fuzzy sets A and B, is the fuzzy set $A \cup U$, whose membership function is: $\mu_{A \cup B}(x) = \max\big(\mu_A(x), \mu_B(x)\big)$. The *intersection* of two fuzzy sets A and B, is the fuzzy set $A \cap B$, whose membership function is: $\mu_{A \cap B}(x) = \min\big(\mu_A(x), \mu_B(x)\big)$. The *complement* of a fuzzy set A, is the fuzzy set denoted by A^c, whose membership function is: $\mu_{A^c}(x) = 1 - \mu_A(x)$ [18]. Operations on fuzzy sets comply with reflexive, transitive, commutative, associative and distributive properties as well as with absorption, involution and De Morgan's laws. Instead, complementarity and mutual exclusivity laws are no longer valid for fuzzy sets.

Let A be a fuzzy set on a collection of objects X and $\alpha \in$ [0,1], the *α-cut* of A is the crisp set A_α given by:

$$A_\alpha = \{x \in X \mid \mu_A(x) \geq \alpha\}. \tag{4}$$

A fuzzy set A, which is defined on the set of real numbers $\mathbb{R}$, is called *convex* if all its α-cuts A_α are convex sets for any $\alpha \in [0,1]$ i.e. if it is verified that $\mu_A(\alpha x + (1-\alpha)y) \geq \min\big(\mu_A(x), \mu_A(y)\big)$ for any $\alpha \in [0,1]$ and $x, y \in \mathbb{R}$.

A *fuzzy relation* is a relation where various degrees of association strength between elements are allowed. Given two collections of objects X and Y, a fuzzy relation R from X to Y (or on $X \times Y$) is defined as:

$$R = \{\big((x,y), \mu_R(x,y)\big) \mid (x,y) \in X \times Y\}. \tag{5}$$

The relation R can be seen as a fuzzy subset of $X \times Y$. If $X = Y$ then R is called fuzzy relation on X.

Fuzzy relations in different spaces can be combined together. Let R be a fuzzy relation on the space $X \times Y$ and S a fuzzy relation on the space $Y \times Z$, the *max-min composition* of R and S, denoted by $R \circ S$, is defined as:

$$R \circ S = \left\{ \left((x,z), \max_y \Big(\min\big(\mu_R(x,y), \mu_S(y,z)\big)\Big) \right) \middle| \; x \in X, y \in Y, z \in Z \right\} \tag{6}$$

A fuzzy relation R on X is called *reflexive* if $\mu_R(x,x) = 1$ for any $x \in X$; it is called *symmetric* if $\mu_R(x,y) = \mu_R(y,x)$ for any $x, y \in X$; it is called *max-min transitive* if $R \circ R \subset R$. A fuzzy relation that is reflexive and symmetric is called *fuzzy proximity relation*; a fuzzy relation that is reflexive, symmetric, and max-min transitive is called *fuzzy similarity relation*.

A convex normal fuzzy set A on $\mathbb{R}$ is called a *fuzzy number* if there is exactly one $x \in \mathbb{R}$ so that $\mu_A(x) = 1$ (x is called *mean value* of A) and μ_A is piecewise continuous. A sample fuzzy number A with membership function $\mu_A\,(x) = \frac{1}{1+(x-5)^2}$ is shown in Figure 2.a. Following the *extension principle* defined in [19] it is possible to extend basic operations to fuzzy numbers. If A and B are fuzzy numbers and $*\colon \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ is a binary operation then the membership function of the fuzzy number $A * B$ is given by:

$$\mu_{A*B}\,(z) = \sup_{z=x*y} \min\big(\mu_A(x), \mu_B(y)\big) \;\text{ for any } x, y \in \mathbb{R} \tag{7}$$

A fuzzy number A is of *LR-type* if there exist functions L (left) and R (right), and scalars $\alpha > 0$ and $\beta > 0$ so that the membership function of A can be expressed as:

$$\mu_A(x) = \begin{cases} L\left(\dfrac{m-x}{\alpha}\right) & \text{for } x \leq m \\ R\left(\dfrac{x-m}{\beta}\right) & \text{for } x \geq m \end{cases} \tag{8}$$

where m is the mean value of A while α and β are called the left and right *spreads*, respectively. A LR fuzzy number A can be symbolically denoted as $(m, \alpha, \beta)_{LR}$. If the mean value is not a real number but an interval $[\underline{m}, \overline{m}]$ then A is called *LR fuzzy interval* and is denoted as$(\underline{m}, \overline{m}, \alpha, \beta)_{LR}$. Figure 2.b shows the membership function of the LR fuzzy number $(4,2,3)_{LR}$ with $L(x) = e^{-x^2}$ and $R(x) = e^{-2x}$.

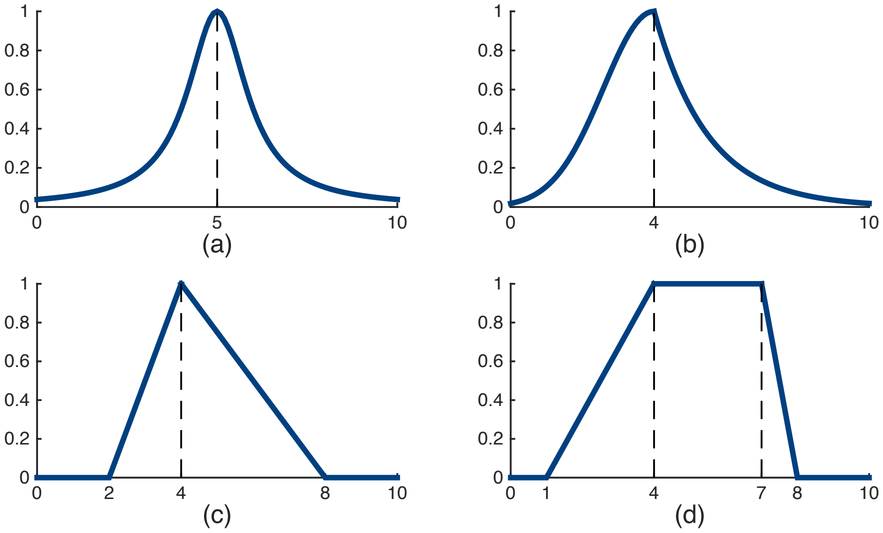

Figure 2. The membership function of sample fuzzy numbers

Operations with LR fuzzy numbers can be simplified with respect to the application of equation (7). Let $A = (m, \alpha, \beta)_{LR}$ and $B = (n, \gamma, \delta)_{LR}$ be LR fuzzy numbers then: $A + B = (m + n, \alpha + \gamma, \beta + \delta)_{LR}$; $-A = (-m, \beta, \alpha)_{LR}$;

$A - B = (m - n, \alpha + \delta, \beta + \gamma)_{LR}$. Approximate expressions for other kind of operations are also shown in [18].

A LR fuzzy number $(m, \alpha, \beta)_{LR}$ with $L(x) = R(x) = \max(0, 1 - x)$ is named *triangular fuzzy number* and can be alternatively denoted with the triplet $(m - \alpha, m, m + \beta)$. Figure 2.c shows the membership function of the sample triangular fuzzy number (2,4,8). A LR fuzzy interval $(\underline{m}, \overline{m}, \alpha, \beta)_{LR}$ with $L(x) = R(x) = \max(0, 1 - x)$ is named *trapezoidal fuzzy number* and can be denoted with $(\underline{m} - \alpha, \underline{m}, \overline{m}, \overline{m} + \beta)$. Figure 2.d shows the membership function of the sample trapezoidal fuzzy number (1,4,7,8).

Fuzzy sets are useful to describe and assess information when it is difficult or impossible to do that precisely in a quantitative manner. These situations often involve attempting to qualify an event or an object by our human perception, and therefore often they lead to use words in natural languages instead of numerical values. To deal with these situations, linguistic variables are often used. Such variables can assume values that are not numbers but words or sentences in a natural or artificial language and rules are provided to map such variables on fuzzy sets.

A *linguistic variable* L is characterized by a quintuple (x, T, U, G, M) where x is the name of the variable, T is the set of possible linguistic values of x, U is a collection of objects representing the universe of discourse, G is a syntactic rule for generating elements of T (usually a grammar) and M is a semantic rule for associating the meaning $M(t)$, which is a fuzzy subset of U, to each term $t \in T$.

1.4 Fuzzy Preferences Modeling in GDM

A GDM problem is characterized by a group of decision makers (also called *experts* hereinafter) $E = \{e_1, \dots, e_m\}$, each with her own knowledge, ideas, experience and motivation, that express their preferences on a finite set of alternatives $X = \{x_1, \dots, x_n\}$ to achieve a common solution. Several ways to express and model experts' preferences on available alternatives have been proposed so far by different researchers [2, 20]. We analyze below the main features of the most popular ones.

The ordering of alternatives from best to worst, also known as *ordinal ranking*, is one of the simplest preference expression models, useful when decision makers have difficulties in assessing quantitatively the strength of their preferences. In this case, according to [21], the possibilities of deriving recommendations based on incorrect information are reduced. The ordinal ranking provided by an expert $e_k \in E$ can be represented as an ordering array $O_k = \big(o_k(x_1), \dots, o_k(x_n)\big)$ being a permutation function which returns the position of any alternative $x_i \in X$ [22].

By using *utility values*, an expert $e_k \in E$ can expresses her preferences through the definition of an utility function $U_k\colon X \to [0,1]$ that associates a crisp value to each alternative [2]. Utility functions are supposed to preserve the preference ordering of the alternatives in a way that if $U_k(x_i) > U_k(x_j)$ then x_i is preferred to x_j while, if $U_k(x_i) = U_k(x_j)$, then x_i is indifferent to x_j for $x_i, x_j \in X$. Utility values allow experts to give precise estimates of their preferences but may introduce errors due to experts evaluating the same alternatives at different scales. To mitigate this issue, rating techniques have been defined based on anchors points and intervals [23].

With *fuzzy estimates*, each expert $e_k \in E$ associates a fuzzy number $l_k(x_i)$ to each alternative $x_i \in X$. Such fuzzy number can be specified or indirectly expressed by means of a linguistic term [24]. Figure 3 shows an example of linguistic terms that may be used in a GDM process and the membership function of the corresponding fuzzy numbers. The use of linguistic terms makes the preference elicitation process more intuitive but its effectiveness can be hindered by differences in the interpretation of the linguistic terms. Techniques for equalizing fuzzy sets have been defined for reducing this type of elicitation error [2].

By using *preference relations*, each expert is asked to express the relative preference of each alternative with respect to any other through the definition of a positive reciprocal $n \times n$ matrix M where each element m_{ij} is a preference intensity ratio and can be interpreted as "x_i is m_{ij} times as good as x_j" with $x_i, x_j \in X$. Under the condition of multiplicative reciprocity, once an expert provides a value for m_{ij}, m_{ji} is automatically obtained as $m_{ji} = 1/m_{ij}$ [25]. A consistent preference relation also satisfies the multiplicative transitivity property i.e. $m_{ik} = m_{ij} \cdot m_{jk}$ for each $i, j, k \in \{1 \dots n\}$. Unfortunately experts often provide preference relations that are only partially consistent. In these cases it is possible to apply specific algorithms to improve consistency [26].

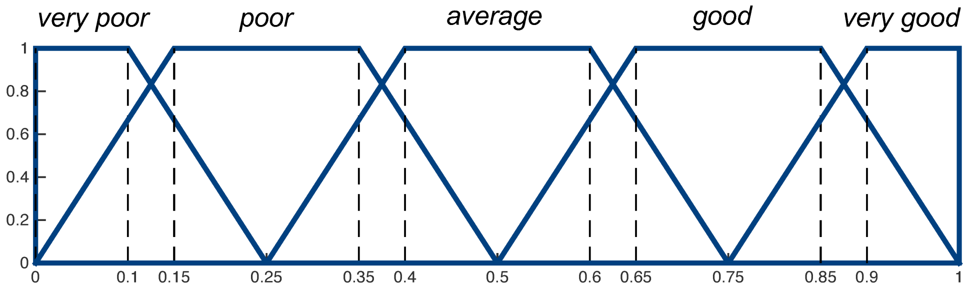

Figure 3. A sample set of linguistic terms for fuzzy estimates

With *fuzzy preference relations* (FPRs) each expert specifies the degree to which each alternative x_i is at least as good as any other alternative x_j with $x_i, x_j \in X$ by means of a fuzzy relation P. According to the definition of fuzzy relation given in section 1.3, a FPR P on X can be defined as a fuzzy set on $X \times X$ with a membership function $\mu_P\colon X \times X \rightarrow [0,1]$ such that [27]:

$$\mu_P(x_i, x_j) = \begin{cases} 1 & \text{if } x_i \text{ is definitely preferred to } x_j, \\ a \in (0.5, 1) & \text{if } x_i \text{ is slightly preferred to } x_j, \\ 0.5 & \text{if } x_i \text{ and } x_j \text{ are evenly preferred,} \\ b \in (0, 0.5) & \text{if } x_j \text{ is slightly preferred to } x_i, \\ 0 & \text{if } x_j \text{ is definitely preferred to } x_i. \end{cases} \tag{9}$$

A FPR P can be conveniently represented as a $n \times n$ matrix $P = (p_{ij})$ where $p_{ij} = \mu_P(x_i, x_j)$. A FPR satisfying the *additive reciprocity* property so that $p_{ij} + p_{ji} = 1 \; \forall \; i, j \in \{1, \dots, n\}$, is said to be *reciprocal.* This means that the preference relation is asymmetric, i.e. if x_i is preferred to x_j then x_j is not preferred to x_i and, as a consequence, $p_{ii} = 0.5\ \forall i \in \{1, \dots, n\}$ (i.e. any alternative is never preferred to itself). Moreover, a FPR that satisfies the *additive transitivity* property so that $p_{ij} + p_{jk} + p_{ki} = 1.5\ \forall i, j, k \in \{1, \dots, n\}$, is also said to be *additive consistent* [28].

Similarly to preference relations, in the elicitation process of FPRs it is necessary to collect $n(n-1)/2$ pairwise comparisons but it is also possible to collect just $n-1$ preferences and estimate the missing ones by enforcing additive transitivity with methods described in section 1.7. Conversely, if an expert provides all preferences but the FPR values do not satisfy additive transitivity, it is also possible to improve such values by modifying them to guarantee an acceptable level of consistency [29].

Among the existing preference models, FPRs are one of the most diffused. According to [30], the main advantage of FPRs is that they allow experts to focus on two alternatives at a time facilitating, in this way, the expression of more accurate preferences with respect to non-pairwise methods. They also ensure a high level of expressiveness and translation techniques are available to convert preference information from any other representation model to FPRs and backward [2]. For this reason, in the next sub-sections we assume that experts' preferences are available in form of FPR.

1.5 Fuzzy Preferences Aggregation in GDM

Once each expert $e_k \in E$ has expressed her preferences on each alternative $x_i \in X$, m individual FPRs $P_1, \dots, P_m$ are available where $P_k = (p_{ij}^k)$. A first step needed to reach a final decision is to aggregate available individual FPRs into a collective one by using some aggregation operator [1]. Several operators have been proposed for this purpose by different researchers, each based on a different mapping from $[0,1]^m$ to $[0,1]$. We describe the most diffused.

One of the simplest preferences aggregation operators is the *Weighted Arithmetic Mean* (WAM) [31]. Let $(p_{ij}^1, \dots, p_{ij}^m)$ with $i, j \in \{1, \dots, n\}$ be a list of preference values to aggregate, coming from $P_1, \dots, P_m$, the WAM operator on these values is defined as:

$$WAM(p_{ij}^1, \dots, p_{ij}^m) = \sum_{k=1}^{m} w_k p_{ij}^k \tag{10}$$

where $w_1, \dots, w_m \in [0,1]$ are weights such that $\sum_{k=1}^{m} w_k = 1$. Weights may represent the relative importance of each expert or can be selected with the aim of maximizing the consistency of the resulting FPR [27].

Another aggregation operator is the *Weighted Geometric Mean* (WGM) [31] that can be defined as follows:

$$WGM(p_{ij}^1, \dots, p_{ij}^m) = \prod_{k=1}^{m} (p_{ij}^k)^{w_k} \tag{11}$$

where each symbol has the same meaning of equation (10). Both WAM and WGM have a compensatory behavior (i.e. they allow a bad evaluation given by an expert to be compensated by a good one from another expert) while the compensatory character of WGM is weaker than that of WAM [2].

A non-compensatory aggregation operator is *min operator* [32] that can be trivially defined as follows:

$$min(p_{ij}^1, \dots, p_{ij}^m) = \min_{1 \leq k \leq m} p_{ij}^k \tag{12}$$

where each symbol has the same meaning of equations (10) and (11). The min operator is particularly helpful when the group agrees that the collective decision should be pessimistic, in the sense that an alternative which was badly evaluated by any expert should be badly evaluated by the group in a non-compensatory way [2].

The *Ordered Weighted Average* (OWA) [33] is among the most diffused aggregation operators. It can be defined as follows:

$$OWA(p_{ij}^1, \dots, p_{ij}^m) = \sum_{k=1}^{m} w_k p_{ij}^{\sigma(k)} \tag{13}$$

where each symbol has the same meaning of equations (10), (11) and (12) while $\sigma: \{1, \dots m\} \to \{1, \dots m\}$ is a permutation function aimed at reordering the values to aggregate such that $p_{ij}^{\sigma(k)} \geq p_{ij}^{\sigma(k+1)}$ for $k \in \{1, \dots, m-1\}$.

A basic aspect of this operator is the re-ordering step. In particular, *the degree of membership of an element in a fuzzy set is not associated with a particular weight. Rather a weight is associated with a particular ordered position of a degree of membership in the ordered set of relevant degrees of membership* [18].

The behavior of OWA strictly depends on the used weight vector. In [22], the authors propose to initialize the weight vector starting from an *increasing proportional linguistic quantifier* to let OWA undertake the behavior of *soft majority*. While the majority is traditionally defined as a threshold number of individuals, soft majority is a fuzzy concept which is controlled through linguistically quantified propositions.

Quantifiers represent the amount of items satisfying a given statement. While classical logic is restricted to the use of two quantifiers (there exists, for all), human discourse is much richer and more diverse in its quantifiers (about 10, almost all, a few, many, most, as many as possible, nearly half, at least half, etc.). *Linguistic quantifiers* have been introduced in [34] to bridge the gap between formal systems and natural discourse. In particular, *absolute linguistic quantifiers* (about 2, more than 5, etc.) are represented as fuzzy subsets of $\mathbb{R}^+$, while *proportional linguistic quantifiers* (most, at least half, etc.) are represented as fuzzy subsets of the unit interval [0,1].

Given a *proportional linguistic quantifier* Q, the membership function $\mu_Q(r)$ represents the degree to which the proportion $r \in [0,1]$ is consistent with the meaning of Q. Functionally, linguistic quantifiers can be *increasing* (most, at least half, etc.), *decreasing* (a few, at most half, etc.) or *unimodal* (about half, about all, etc.). In particular, increasing quantifiers satisfy the property: $\mu_Q(r_1) \geq \mu_Q(r_2)$ for any $r_1 > r_2$. The membership function $\mu_Q(r)$ of an increasing proportional linguistic quantifier Q can be written as:

$$\mu_Q(r) = \begin{cases} 0 & \text{if } r < a, \\ \dfrac{r-a}{b-a} & \text{if } a \leq r \leq b, \\ 1 & \text{if } r > b. \end{cases} \tag{14}$$

with $a, b, r \in [0,1]$. Examples of increasing proportional linguistic quantifiers and the related membership functions are shown in Figure 4. The parameters (a, b) of such quantifiers are: (0,1); (0,0.5); (0.3,0.8); (0.5,1) respectively.

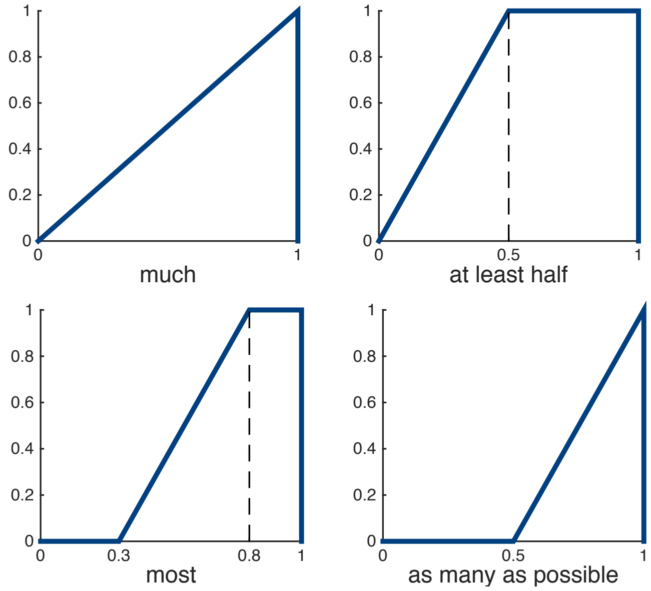

Figure 4. Example of increasing proportional linguistic quantifiers

The weights of an OWA operator of dimension m can be obtained from an increasing proportional linguistic quantifier as follows [33]:

$$w_k = \mu_Q\left(\frac{k}{m}\right) - \mu_Q\left(\frac{k-1}{m}\right); \; k \in \{1, \ldots, m\}. \tag{15}$$

where the quantifier Q must be selected to reflect the fusion strategy that the decision makers would apply (i.e. the ratio of experts that are expected to be satisfied with the aggregated preference value). In this way it is possible to obtain collective evaluations in which the opinions of most of the experts involved in the decision problem are considered.

In this way, every collective preference p_{ij} for $i, j \in \{1, \dots, n\}$ is obtained as: $p_{ij} = OWA_Q(p_{ij}^1, \dots, p_{ij}^m)$, where OWA_Q is the OWA operator initialized with the weights coming from the quantifier Q. By extending the notation to matrices, we can rewrite the equation as: $P = OWA_Q(P_1, \dots, P_m)$.

When the relative importance of each expert must be taken into account during the aggregation step (e.g. to reflect experts' different backgrounds and levels of knowledge about the problem) specific versions of the OWA operator can be used. For example, the *Induced OWA* operator (IOWA) induces the reordering of the set of values to aggregate on the reordering of a set of values associated with them [35].

Based on IOWA, the *Importance IOWA* operator (I-IOWA) has been defined in [36] to consider the importance of each expert in the aggregation step while being guided by a proportional quantifier as in equation (15). Let $(p_{ij}^1, \dots, p_{ij}^m)$ with $i, j \in \{1, \dots, n\}$ be a list of preference values to aggregate, coming from the FRPs $P_1, \dots, P_m$, let $u_i \in [0{,}1]$ be the importance degree of each $e_i \in E$ and Q a non-decreasing proportional fuzzy quantifier, then the I-IOWA operator is defined as follows:

$$\textit{I-IOWA}_Q\left((p_{ij}^1, u_1), \dots, (p_{ij}^m, u_m)\right) = \sum_{k=1}^{m} w_k p_{ij}^{\sigma(k)} \tag{16}$$

were σ: $\{1, \dots m\} \to \{1, \dots m\}$ is a permutation function so that $u_{\sigma(k)} \geq u_{\sigma(k+1)}$ for each $k \in \{1, \dots, m-1\}$ and the k-th weight w_k is obtained as follows:

$$w_k = \mu_Q\left(\frac{S(k)}{S(m)}\right) - \mu_Q\left(\frac{S(k-1)}{S(m)}\right);\ k \in \{1, \dots, m\}. \tag{17}$$

where $S(k) = \sum_{l=1}^{k} u_{\sigma(k)}$.

Extending the notation to matrices, given a set of individual FPRs $P_1, \dots, P_m$ and a vector of experts' importance degrees $U = (u_1, \dots u_m)$, the collective FPR P that takes into account the importance of each expert can be obtained as $P = \textit{I-OWA}_Q\big((P_1, u_1), \dots, (P_m, u_m)\big)$.

1.6 Fuzzy Alternatives Ranking in GDM

Once the individual FPRs have been aggregated in a collective one $P = (p_{ij})$ through one of the methods described in section 1.5, the available alternatives must be rated associating a *degree of preference* $\phi(x_i)$ to each $x_i \in X$. Then the best one (i.e. the one associated with the higher degree of preference) is selected. Several measures have been proposed so far to quantify the degree of preference of each alternative basing on the collective FPR. We describe below the most diffused ones.

In [27] the degree of preference of each alternative is calculated in terms of *Net Flow* (NF) as follows:

$$\phi_{NF}(x_i) = \sum_{j=1, j\neq i}^{n} p_{ij} - \sum_{j=1, j\neq i}^{n} p_{ji} \qquad (18)$$

where the first summation is the *leaving flow* i.e. the total degree of preference of x_i over all the other alternatives, while the last summation is the *entering flow* i.e. the total degree of preference of all the other alternatives over x_i.

A different measure has been proposed in [37, 38] where the score of an alternative is calculated in term of *Non-Dominance Degree* (NDD) i.e. the degree in which the alternative is not dominated by the others:

$$\phi_{NDD}(x_i) = 1 - \max_{1\leq j\leq n;\ j\neq i} (p_{ji} - p_{ij}, 0) \qquad (19)$$

In [22, 39] the *Quantifier Guided Dominance Degree* (QGDD) has been proposed to calculate the dominance that one alternative has over all the others in a soft majority sense:

$$\phi_{QGDD}(x_i) = OWA_Q(p_{ij};\ j = 1, \dots, n;\ j \neq i). \tag{20}$$

where OWA_Q specifies the OWA operator initialized with the weights coming from the increasing proportional linguistic quantifier Q as defined in 1.5.

In [22, 39] the authors propose a NDD version named *Quantifier Guided Non-Dominance Degree* (QGNDD) to calculate the degree in which a given alternative is not dominated by a soft majority of the remaining ones:

$$\phi_{QGNDD}(x_i) = OWA_Q\left(1 - \max_{1 \le j \le n;\ j \neq i}(p_{ji} - p_{ij}, 0)\right). \tag{21}$$

When the quantifier Q represents the statement "all", that has a membership function obtainable from equation (14) with $a = b = 1$, the definition of the QGNDD measure coincides with that of NDD.

GGDD and QGNDD can be also used in combination. In particular, two different selection policies can be applied according to [22]: a sequential or a conjunctive one. In the *sequential policy*, a measure is selected and applied obtaining a selection set of alternatives reaching the maximum score. If such set includes more than one alternative, then the other measure is applied to select the alternative of the above set with the best score. In the *conjunctive policy* both measures are applied obtaining two distinct selection sets that are intersected to obtain the final one. This policy is more restrictive than the former because it is possible to obtain an empty selection set. In [39] it is suggested to apply the conjunctive policy as the first step and then apply the sequential one just in case the first one returns an empty set.

After having rated the available alternatives with one (or a combination) of the described measures, the one with the highest degree of preference is the solution of the GDM problem.

1.7 Dealing with Incomplete Information

Sometimes, due to domain complexity, limited expertise or pressure to make a decision, it may be difficult or even impossible for an expert to express a preference on every pair of alternatives. This leads to incomplete FPRs where missing values have to be estimated in a non-contradictory way with respect to expressed preferences. Several methods have been proposed so far for this purpose as described below.

In [40] a method to estimate the missing values of an FPR P by applying reciprocity and additive transitivity properties on the existing values of the same FPR is proposed. In fact, the definition of additive transitivity provided in section 1.4, allows to obtain the following three estimates of the preference p_{ij}, of alternative x_i over alternative x_j, using an intermediate alternative x_k with $x_i, x_j, x_k \in X$:

$$\begin{aligned} \varepsilon_k^1(p_{ij}) &= p_{ik} + p_{kj} - 0.5; \\ \varepsilon_k^2(p_{ij}) &= p_{kj} - p_{ki} + 0.5; \\ \varepsilon_k^3(p_{ij}) &= p_{ik} - p_{jk} + 0.5. \end{aligned} \tag{22}$$

If P is additive consistent, then $\varepsilon_k^1(p_{ij}) = \varepsilon_k^2(p_{ij}) = \varepsilon_k^3(p_{ij})$ for all values $i, j, k \in \{1, \dots, n\}$. Unfortunately, user defined FPRs are not always additive consistent. In this case it is still possible to use equation (22) to identify missing values that are as consistent as possible with the existing ones by mediating the estimates over any defined intermediate alternative.

If $D = \big\{(i, j) \mid i, j \in \{1, \dots, n\};\ p_{ij} \text{ is defined in } P\big\}$ is a set including the positions of all the defined values of P and $U = \big\{(i, j) \mid i, j \in \{1, \dots, n\}\big\} \setminus D$ is the set including the positions of the undefined ones, the overall estimator of a missing preference p_{ij} with $(i, j) \in U$ can be defined as follows:

$$\varepsilon(p_{ij}) = \frac{\sum_{l=1}^{3} \sum_{k \in K_{ij}^l} \varepsilon_k^l(p_{ij})}{\sum_{l=1}^{3} |K_{ij}^l|} \tag{23}$$

where the sets $K_{ij}^1 = \{k \mid (i,k),(k,j) \in D\}$, $K_{ij}^2 = \{k \mid (k,j),(k,i) \in D\}$ and $K_{ij}^3 = \{k \mid (i,k),(j,k) \in D\}$ include the indexes of the defined intermediate alternatives, useful for each estimator of p_{ij}.

The generation of missing values through equations (22) and (23) is done in several iterations. In each iteration new values are computed based on those previously known and added to the FPR. In particular, being $P^{(0)}$ the initial FPR and $P^{(t)}$ the same FPR after t iterations, the missing FPR values that can be estimated at step $t+1$ are:

$$E^{(t+1)} = \left\{(i,j) \in U^{(t)} \mid K_{ij}^{1(t)} \cup K_{ij}^{2(t)} \cup K_{ij}^{3(t)} \neq \emptyset\right\} \tag{24}$$

where $U^{(t)}$ collects the position of undefined values of $P^{(t)}$ while each $K_{ij}^{l(t)}$ (with $1 \leq l \leq 3$) includes the indexes for the l-th estimator of p_{ij} in $P^{(t)}$. If at the t-th iteration the set $E^{(t+1)}$ is empty, then no more elements of P can be estimated and the process stops.

In [30], a different approach has been defined to estimate missing FPR values based on reciprocity and multiplicative transitivity. According to [41], a FPR P is multiplicative transitive if:

$$p_{ij} \cdot p_{jk} \cdot p_{ki} = p_{ik} \cdot p_{kj} \cdot p_{ji} \ \forall i,j,k \in \{1, \dots, n\}. \tag{25}$$

Applying equation (25), when an FPR P is multiplicative transitive, the preference p_{ij} of alternative x_i over alternative x_j can be estimated using an intermediate alternative x_k with $x_i, x_j, x_k \in X$ in this way:

$$\varepsilon_k(p_{ij}) = \frac{p_{ik} \cdot p_{kj} \cdot p_{ji}}{p_{jk} \cdot p_{ki}}. \tag{26}$$

By considering that, based on reciprocity, $p_{ij} = 1 - p_{ji} \forall\ i,j \in \{1, \dots, n\}$, we can modify equation (26) as follows:

$$\varepsilon_k(p_{ij}) = \frac{p_{ik} \cdot p_{kj}}{p_{ik} \cdot p_{kj} + (1 - p_{ik}) \cdot (1 - p_{kj})}. \tag{27}$$

When the FPR to be completed is not multiplicative transitive, it is still possible to use equation (27) to identify missing values that are as consistent as possible with the existing ones by mediating the estimates over any defined intermediate alternative. Let D and U be respectively the sets including the positions of defined and undefined elements of P (as previously defined) we can estimate a missing preference p_{ij} with $(i,j) \in U$ as follows

$$\varepsilon(p_{ij}) = \frac{\sum_{k \in K_{ij}} \varepsilon_k(p_{ij})}{|K_{ij}|}. \tag{28}$$

where the set $K_{ij} = \{k \mid (i,k),(k,j) \in D\}$ includes the indexes of any defined intermediate alternative between x_i and x_j, useful for the estimator. Also in this case the estimation of missing values proceeds in several iterations and the process stops when no additional elements can be estimated.

Both methods described in this sub-section use FPR values related to an alternative to infer missing FPR values connected to the same alternative. If no preferences at all are available for a given alternative, then it is impossible to estimate any of them. This happens when does exist an alternative $x_i \in X$ so that any p_{ij} and p_{ji} is undefined for any $j \in \{1, \dots, n\}$.

In [42], the authors refer to this case as an *ignorance* situation and suggest to initialize missing FPR values with some seed values that are subsequently refined through an iterative process based on equations (22)-(23) or (27)-(28) to make them as consistent as possible with the existing values. Four different ways to obtain *seed values* have been proposed:

- *indifference*: undefined preferences are initially set to 0.5;
- *alternative proximity*: seed values are obtained from the preferences given by the same expert to similar alternatives (this implies having additional information on problem alternatives allowing to define a distance measure between them);
- *collective seed value*: seed values are chosen from the collective FPR that is obtained by aggregating partial individual FPRs;

- *expert proximity*: seed values are chosen from the FPRs provided by the experts that are nearest to the expert whose FPR has to be completed (where distances between experts can be calculated by averaging the absolute differences between defined FPR values).

The first approach is useful when there are no additional external sources of information about the problem and when a high FPRs consistency level is required. The second approach is only feasible when some kind of metadata on alternatives is available. The third and fourth approaches, making the opinions of the experts closer, are useful when a fast consensus is needed. The fourth approach is also able to maintain high the FPRs consistency level. The first two approaches are also named *individual strategies* because they rely on information coming from the same expert to estimate missing values while the last two are named *social strategies* because they use information coming from other experts.

Once generated, seed values must be refined to make them more coherent with existing FPR values. If the set U includes the position of the undefined elements of P as defined before, in case of an ignorance situation, seed values are generated for any p_{ij} so that $(i,j) \in U$ and included in P. Then, $\varepsilon(p_{ij})$ is calculated through equations (23) or (28) for any p_{ij} so that $(i,j) \in U$ and obtained values are substituted to seed values in P.

Chapter 2

Modeling Expert Preferences with Fuzzy Rankings

Although FPRs are among the most commonly used preference models in GDM, they are not free from drawbacks. First of all, especially when dealing with many alternatives, the definition of FPRs becomes complex and time-consuming. Moreover they allow to focus on only two options at a time. This facilitates the expression of preferences but, on the other hand, let experts lose the global perception of the problem with the risk of introducing several inconsistencies that impact negatively on the whole decision process.

For these reasons, different preference models are often adopted in real GDM settings (as reported in section 1.4) and, if necessary, transformation functions are applied to obtain equivalent FPRs. In this chapter we propose *Fuzzy Rankings*, a new preferences model that offers and higher level of user-friendliness with respect to FPRs while trying to maintain an adequate level of expressiveness. Fuzzy rankings allow experts to focus on two alternatives at a time without losing the global picture so reducing inconsistencies.

After having deepened the ordinal ranking model (already introduced in section 1.4), the proposed model for fuzzy rankings is described as a fuzzy extension of the ordinal one. Conversion algorithms from fuzzy rankings to FPRs and backward are then defined as well as similarity measures useful when evaluating the concordance between experts' opinion. Eventually a comparison of the proposed model with related works is reported.

2.1 Ordinal Rankings

As seen in section 1.4, the *Ordinal Ranking* is one of the simplest preference models for GDM. Let $X = \{x_1, \dots, x_n\}$ be a set of alternatives, an ordinal ranking on X specifies an ordering $x_{\sigma(1)} \succ \cdots \succ x_{\sigma(n)}$ between its elements where $\sigma\colon \{1, \dots n\} \to \{1, \dots n\}$ is a permutation function. An ordinal ranking can be conveniently represented through an ordering array $O = (o_1, \dots, o_n)$ where each element $o_i \in \{1, \dots, n\}$ states the position of i-th alternative of X within the ranking.

Example 1. *Let $X = \{x_1, x_2, x_3\}$ be a set of alternatives, the ordering array $O = (2, 3, 1)$ specifies that the alternatives x_1, x_2 and x_3 are ranked second, third and first respectively. Using the alternate notation, case we can describe the same ordinal ranking as: $x_3 \succ x_1 \succ x_2$.*

In GDM problems, each expert $e_i \in E$ defines an individual ranking by specifying the ordering array $O_i = (o_k^i)$ with $i \in \{1, \dots, m\}$ and $k \in \{1, \dots, n\}$ on the same set X. To assess the level of agreement between experts, several methods to evaluate ranking similarity have been defined so far by different researchers [43]. Let O_i and O_j be two ordinal rankings on the same set X, the *Kendall's rank correlation coefficient* [44, 45] is defined as:

$$\tau(O_i, O_j) = \frac{2(c_{ij} - d_{ij})}{n(n-1)} \tag{29}$$

where c_{ij} is the number of concordant pairs and d_{ij} the number of discordant pairs between O_i and O_j. A concordant pair is pair of alternatives of X which have the same order in the two rankings while a discordant pair is a pair of alternatives which have the opposite order in the two rankings.

The Kendall's rank correlation coefficient is normalized in [−1,1]. In the case of maximus similarity between O_i and O_j (i.e. if rankings are identical), then $\tau(O_i, O_j) = 1$. In the case of maximum dissimilarity (i.e. if one ranking is the reverse of the other), then $\tau(O_i, O_j) = -1$. A value of zero indicates the absence of any association between the two rankings.

Another measure of a correlation between rankings is the *Spearman‘s rank correlation coefficient* [46] that is defined as:

$$\rho(O_i, O_j) = 1 - \frac{6\sum_{k=1}^{n}\left(o_k^i - o_k^j\right)^2}{n(n^2-1)} \tag{30}$$

where o_k^i is the k-th element of O_i and o_k^j is the k-th element of O_j. Also the Spearman‘s rank correlation coefficient is normalized in the interval $[-1,1]$ and its interpretation is analogous to the previous ones.

Example 2. *Let $X = \{x_1, \dots, x_5\}$ be a set of alternatives, $O_1 = (5,3,4,1,2)$ and $O_2 = (4,2,5,3,1)$ two ordering arrays defined on X, where the first one represents the ranking $x_4 \succ x_5 \succ x_2 \succ x_3 \succ x_1$ while the second represents the ranking $x_5 \succ x_2 \succ x_4 \succ x_1 \succ x_3$. According to equations (29) and (30) we obtain that: $\tau(O_1, O_2) = 0.4$ and $\rho(O_1, O_2) = 0.6$. So both indexes show a positive correlation between the two rankings.*

In order to use ordinal rankings in conjunction with fuzzy models and methods for GDM, it can be convenient to convert them in FPRs. According to [2], it is possible to convert an ordering array O of size n into an $n \times n$ FPR $P = (p_{ij})$ through any function $H: \{1, \dots, n\}^2 \to [0,1]$ satisfying the following conditions:

- $H(o_i, o_j)$ is a non-increasing function of the first argument and a non-decreasing function of the second argument;
- $H(o_i, o_i) = 0.5$ $\forall i \in \{1, \dots, n\}$;
- $H(o_i, o_j) > 0.5$ if $o_i < o_j$ $\forall i, j \in \{1, \dots, n\}$;
- $H(o_i, o_j) + H(o_j, o_i) = 1$ $\forall i, j \in \{1, \dots, n\}$ (additive reciprocity).

In [22] the following transformation function respecting these conditions has been proposed:

$$p_{ij} = H(o_i, o_j) = \frac{1}{2}\Big(1 + \frac{o_j - o_i}{n-1}\Big) \tag{31}$$

Moreover, the FPRs generated with this function are additive consistent with respect to the definition given in section 1.4.

To transform a FPR back to an ordering array it is possible to associate a degree of preference $\phi(x_i)$ to any $x_i \in X$ according to one of the FPR-based measures defined in section 1.6 and then rank the alternatives with respect to their associated degrees of preference.

Example 3. *Let* $X = \{x_1, x_2, x_3, x_4\}$ *be the set of available alternatives and* $O = (2, 1, 4, 3)$ *be the ordering array provided by an expert. By applying the equation (31) it is possible to obtain the corresponding FPR:*

$$P = \begin{pmatrix} 0.5 & 0.33 & 0.83 & 0.67 \\ 0.67 & 0.5 & 1 & 0.83 \\ 0.17 & 0 & 0.5 & 0.33 \\ 0.33 & 0.17 & 0.67 & 0.5 \end{pmatrix}$$

To transform P back to an ordinal ranking, it is possible to apply equation (18) to calculate the score of each alternative in terms of Net Flow as follows: $\phi_{NF}(x_1) = 0.67$*;* $\phi_{NF}(x_2) = 2$*;* $\phi_{NF}(x_3) = -2$*;* $\phi_{NF}(x_4) = -0.67$*. According to these values, the ordering array that corresponds to P is:* $O = (2, 1, 4, 3)$ *that is exactly the initial one.*

2.2 Evolution to Fuzzy Rankings

Ordinal rankings can be considered too simplistic to model preferences in real GDM problems. Experts are sometimes unable to assign a precise position in a ranking to alternatives that are considered equivalent or, when a position can be assigned, experts may need to specify at what extent an alternative is better than the following one. To overcome these limitations, we introduce in this section the notion of *Fuzzy Ranking* that can be considered as a fair compromise between the expressive capability of FRPs and the user-friendliness of ordinal rankings.

A fuzzy ranking is a sequence $R = (x_{\sigma(1)}\ s_1\ x_{\sigma(2)}\ \dots\ x_{\sigma(k-1)}\ s_{k-1}\ x_{\sigma(k)})$ with $k \leq n$. Terms in odd positions in the sequence represent a subset of the alternatives, while $\sigma: \{1, \dots n\} \to \{1, \dots k\}$ is a k-permutation function. Terms in even positions (separators) belong to the set of symbols $S = \{\gg, >, \geq, \approx\}$ and define a degree of preference between subsequent terms (with $\gg$ meaning "is much better than", $>$ "is better than", $\geq$ "is a little better than" and $\approx$ "is similar to"). Each alternative appears at most once in the ranking so cycles are not allowed although partial rankings are admitted.

Example 4. *The fuzzy ranking $R = (x_4 \gg x_5 \approx x_2 \geq x_3 > x_1)$ defined on $X = \{x_1, \dots, x_5\}$ states that, according to expert's opinion, the fourth alternative is much better than the fifth one that, in turn, is similar to the second one, while both are a little better than the third one that, in turn, is better than the first one.*

If we look at Example 4, it becomes clear that, by relying on standard ordinal rankings, it would have been impossible for the same expert to specify her belief so thoroughly. In fact, the ordinal ranking $x_4 \succ x_5 \succ x_2 \succ x_3 \succ x_1$ that can be extracted from R and can be summarized with the ordering array $O = (5, 3, 4, 1, 2)$, has a deeply different semantics: ties are not allowed so the equivalent alternatives x_5 and x_2 are artificially ordered while the preference gaps between x_4 and x_5, x_2 and x_3, x_3 and x_1 seems comparable in O while they are very different in expert's belief, as expressed in R.

Figure 5 graphically illustrates the interpretation of the expert's belief captured by the fuzzy ranking R of Example 4 and by the extracted ordinal ranking O. As it can be seen, fuzzy rankings offer more tools to highlight the differences between alternatives with respect to ordinal rankings. Inspired by studies on the use of linguistic labels in GDM like [47], the cardinality of S (i.e. the number of available symbols) has been chosen small enough so as not to impose useless precision to the experts and rich enough to allow a discrimination of the relative performance of the alternatives. On the other hand, the possibility to compose fuzzy rankings by chaining alternatives and symbols, allows to indirectly express a wide variety of preference levels.

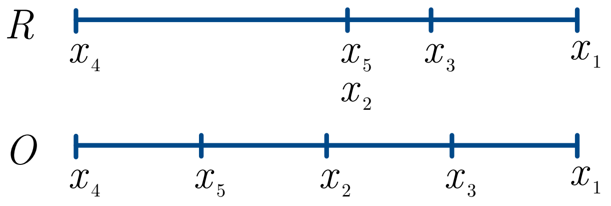

Figure 5. Interpretation of the fuzzy ranking R coming from Example 4 and of the extracted ordinal ranking O

As an option, experts may be allowed to provide multiple fuzzy rankings interesting disjoint subsets of X, rather than just one. In this way it is possible to deal with the case in which some options are considered as mutually incomparable. As for ordinal rankings, conversion algorithms to and from FPRs can be defined for fuzzy rankings, as well as similarity measures. Such methods are described in the next subsections.

2.3 From Fuzzy Rankings to FPRs

Starting from a fuzzy ranking $R = (x_{\sigma(1)}\ s_1\ x_{\sigma(2)}\ \dots\ x_{\sigma(k-1)}\ s_{k-1}\ x_{\sigma(k)})$ it is possible to generate the corresponding FPR $P = (p_{ij})$ in several ways. A *first approach* consists in associating a predefined preference degree $d(s)$ to each symbol $s \in S$ and obtain FPR elements from R in this way:

- $p_{\sigma(i)\sigma(i+1)} = d(s_i)\ \forall i \in \{1, \dots, k-1\}$;
- $p_{\sigma(i+1)\sigma(i)} = 1 - d(s_i)\ \forall i \in \{1, \dots, k-1\}$; (32)
- $p_{\sigma(i)\sigma(i)} = 0.5\ \forall i\{1, \dots, k\}$;

where the first statement transforms the degrees of preference embedded in R in values of P, while the second and third statements are aimed at ensuring the reciprocity of P according to the definition given in section 1.4. A feasible set of values for the function $d(s)$ is shown in Table 1 (second column).

Symbol	Preference degree $d(s)$	Relative strength $\|s\|$
$\gg$	0.85	2
$>$	0.65	1
$\geq$	0.58	0.5
$\approx$	0.50	0

Table 1. Feasible values for the preference degree and the relative strength associated to ranking string symbols

It should be noted that, by applying equations (32) on a fuzzy ranking R, only $3k-2$ elements of P can be defined. Even in the case that R involves all available alternatives, (i.e. when $k=n$), a number of n^2-3n+2 elements of P remain undefined and should be estimated through one of the methods proposed in section 1.7. Moreover, the generated FPR, even when completed in this way, is not guaranteed to be additive consistent.

Example 5. *If $R = (x_4 \gg x_5 \approx x_2 \geq x_3 > x_1)$ is a fuzzy ranking on the set $X = \{x_1, \dots, x_5\}$, the following FPR is generated according to equation (32) using preferences degree values coming from Table 1.*

$$P = \begin{pmatrix} 0.5 & - & 0.35 & - & - \\ - & 0.5 & 0.58 & - & 0.5 \\ 0.65 & 0.42 & 0.5 & - & - \\ - & - & - & 0.5 & 0.85 \\ - & 0.5 & - & 0.15 & 0.5 \end{pmatrix}$$

where the symbol – indicates an undefined cell. Applying equations (22)-(23) on P we can obtain the missing values as follows:

$$P = \begin{pmatrix} 0.5 & 0.27 & 0.35 & 0 & 0.27 \\ 0.73 & 0.5 & 0.58 & 0.15 & 0.5 \\ 0.65 & 0.42 & 0.5 & 0.07 & 0.42 \\ 1 & 0.85 & 0.93 & 0.5 & 0.85 \\ 0.73 & 0.5 & 0.58 & 0.15 & 0.5 \end{pmatrix}$$

A *second approach* for generating a FPR from a fuzzy ranking is through a transformation function similar to that described in section 2.1. A *relative strength* |s| is associated to each symbol $s \in S$ and, given a fuzzy ranking R, a *fractional rank* $r(x_i)$ is associated to each alternative so that:

- $r(x_{\sigma(1)}) = 1$;
- $r(x_{\sigma(i)}) = r(x_{\sigma(i-1)}) + \ |s_{i-1}| \ \forall \ i \in \{2, \dots, k\}$;
- $r(x_i)$ is undefined if $\sigma(i)$ is undefined i.e. if *the i*-th alternative does not appear in R.

(33)

A feasible set of relative strengths for proposed symbols is shown in Table 1 (third column). The relative strength of each symbol has been selected so that each symbol doubles the strength of the next one. By only using the symbol >, the fuzzy ranking becomes an ordering of alternatives and equation (33) generates an ordering array as defined in section 2.1. The use of the symbols $\gg$ or $\geq$ in place of >, respectively doubles or halves the distance of the preceding and subsequent terms in the ranking while the use of $\approx$ means that the preceding and subsequent terms have the same rank.

Then, for any pair of alternatives x_i and x_j appearing in R, basing on a modified version of equation (31), an element of P can be defined as follows:

$$p_{ij} = \frac{1}{2}\left(1 + \frac{r(x_j) - r(x_i)}{rmax - 1}\right) \tag{34}$$

where $rmax = r(x_{\sigma(k)})$ is the maximum rank. The special case $rmax = 1$, arising when an expert considers all alternatives as equivalent i.e. when she sets $R = (x_{\sigma(1)} \approx \cdots \approx x_{\sigma(k)})$, is handled by directly setting $p_{ij} = 0.5$ for any

x_i and x_j appearing in R. Differently from the first approach, by applying equation (34) it is possible to directly define k^2 elements of the corresponding FPR. When R involves all alternatives, (i.e. when $k = n$), the generated FPR presents no undefined elements.

Proposition. *If $P = (p_{ij})$ is a $n \times n$ FPR generated from a fuzzy ranking R according to equations (33)-(34), then the elements of P that exist verify the additive consistency property.*

Proof. *According to the definition given in 1.4, P is additive consistent if $p_{ij} + p_{jk} + p_{ki} = 1.5\ \forall i, j, k \in \{1, \dots, n\}$. Based on equation (34) we obtain:*

$$
\begin{aligned}
p_{ij} + p_{jk} + p_{ki} &= \frac{1}{2}\left(1 + \frac{r(x_j) - r(x_i)}{rmax - 1}\right) + \frac{1}{2}\left(1 + \frac{r(x_k) - r(x_j)}{rmax - 1}\right) \\
&+ \frac{1}{2}\left(1 + \frac{r(x_i) - r(x_k)}{rmax - 1}\right) \\
&= \frac{3}{2} + \frac{r(x_j) - r(x_i) + r(x_k) - r(x_j) + r(x_i) - r(x_k)}{2rmax - 2}.
\end{aligned}
$$

For $rmax \neq 1$ and because the fraction numerator is equal to 0, we have that $p_{ij} + p_{jk} + p_{ki} = 3/2 + 0 = 1.5$ proofing that P is additive consistent. The case $rmax = 1$, which leads to a $0/0$ indeterminate form, is treated separately by setting $p_{ij} = 0.5\ \forall i, j \in \{1, \dots, n\}$. In this case the proof that P is additive consistent is trivial given that:

$$p_{ij} + p_{jk} + p_{ki} = 0.5 + 0.5 + 0.5 = 1.5\ \forall i, j, k \in \{1, \dots, n\}.$$

Example 6. *Let $R = (x_4 \gg x_5 \approx x_2 \geq x_3 > x_1)$ be the same fuzzy ranking of the previous example. Using relative strength values coming from Table 1 in (33), we obtain the fractional rank of available alternative as: $r(x_1) = 4.5$, $r(x_2) = 3$, $r(x_3) = 3.5$, $r(x_4) = 1$, $r(x_5) = 3$. Then, according to equation (34), it is possible to generate the corresponding FPR P as follows:*

$$P = \begin{pmatrix} 0.5 & 0.29 & 0.36 & 0 & 0.29 \\ 0.71 & 0.5 & 0.57 & 0.21 & 0.5 \\ 0.64 & 0.42 & 0.5 & 0.14 & 0.43 \\ 1 & 0.79 & 0.86 & 0.5 & 0.79 \\ 0.71 & 0.5 & 0.57 & 0.21 & 0.5 \end{pmatrix}.$$

Differently from the previous example, there is no need to complete the FPR with techniques coming from section 1.7. Moreover, the resulting FPR can be shown to be additive consistent.

2.4 From FPRs to Fuzzy Rankings

In some cases it can be useful to translate the preferences expressed with a FPR back to a fuzzy ranking. This process can help making manifest and easy to understand experts' defined FPRs or obtaining a meaningful ranking of available alternatives from the collective FPR.

In both cases it is possible to calculate the degree of preference $\phi(x_i)$ of each alternative $x_i \in X$ starting from a (individual or collective) FPR P with one of the methods defined in section 1.6. Then, the corresponding fuzzy ranking $R = (x_{\sigma(1)}\ s_1\ x_{\sigma(2)}\ \dots\ x_{\sigma(n-1)}\ s_{n-1}\ x_{\sigma(n)})$ can be generated where σ is a permutation function such that $\phi(x_{\sigma(i)}) \geq \phi(x_{\sigma(i+1)})$ and $s_i \in S$ for any $i \in \{1, \dots, n-1\}$. Two approaches can be then adopted (reversing the two approaches proposed in section 2.3) to identify the symbols $s_1, \dots, s_{n-1}$.

Given two adjacent alternatives $x_{\sigma(i)}$ and $x_{\sigma(i+1)}$ in R, the *first approach* determines the intermediate symbol s_i from the preference value $p_{\sigma(i)\sigma(i+1)}$ of P as follows:

$$s_i = \begin{cases} \approx & \text{if } p_{\sigma(i)\sigma(i+1)} < 0.54 \\ \geq & \text{if } 0.54 \leq p_{\sigma(i)\sigma(i+1)} < 0.62 \\ > & \text{if } 0.62 \leq p_{\sigma(i)\sigma(i+1)} < 0.75 \\ \gg & \text{if } p_{\sigma(i)\sigma(i+1)} \geq 0.75 \end{cases} \tag{35}$$

for any $i \in \{1, \dots, n-1\}$, where the threshold values 0.54, 0.62 and 0.75 are obtained by averaging each pair of subsequent preference degree values from Table 1 (second column).

This approach is practicable when the starting FPR respects additive or multiplicative transitivity properties defined in sections 1.4 and 1.7 i.e. when every FPR value is consistent to the others. Otherwise, it is possible to select one or more non-coherent preference values and, consequently, to generate incongruent ranking symbols. Moreover by directly referring to FPR values, the possible transformations introduced in the calculation of the degree of preference of each alternative, according to the methods defined in section 1.6, are disregarded in the selection of the ranking symbols.

Example 7. *From the additive consistent FPR P resulting from Example 6 it is possible to generate the degree of preference of each alternative in terms of Net Flow according to equation (18):* $\phi_{NF}(x_1) = -2.14$*;* $\phi_{NF}(x_2) = 0$*;* $\phi_{NF}(x_3) = -0.71$*;* $\phi_{NF}(x_4) = 2.86$*;* $\phi_{NF}(x_5) = 0$*. According to these values, it is possible to define the alternative ranking:* $x_4 \succ x_2 \succ x_5 \succ x_3 \succ x_1$*, also representable with the ordering array* $O = (5, 2, 4, 1, 3)$*. The corresponding fuzzy ranking and the related separators can be obtained from equation (35) basing on the FPR values:* $p_{4,2} = 0.79$*;* $p_{2,5} = 0.5$*;* $p_{5,3} = 0.57$*;* $p_{3,1} = 0.64$ *as follows:* $R = (x_4 \gg x_2 \approx x_5 \geq x_3 > x_1)$*.*

Given two adjacent alternatives $x_{\sigma(i)}$ and $x_{\sigma(i+1)}$ in a fuzzy ranking R, the *second approach* determines the intermediate symbol s_i from the degrees of preference $\phi(x_{\sigma(i)})$ and $\phi(x_{\sigma(i+1)})$ that can be associated to the alternatives according to one of the methods defined in section 1.6:

$$s_i = \begin{cases} \approx & \text{if } \phi(x_{\sigma(i+1)}) - \phi(x_{\sigma(i)}) < 0.25 \cdot \delta \\ \geq & \text{if } 0.25 \cdot \delta \leq \phi(x_{\sigma(i+1)}) - \phi(x_{\sigma(i)}) < 0.75 \cdot \delta \\ > & \text{if } 0.75 \cdot \delta \leq \phi(x_{\sigma(i+1)}) - \phi(x_{\sigma(i)}) < 1.5 \cdot \delta \\ \gg & \text{if } (x_{\sigma(i+1)}) - \phi(x_{\sigma(i)}) \geq 1.5 \cdot \delta \end{cases} \tag{36}$$

where δ is the average difference between the degrees of preference of two subsequent alternatives in the ranking:

$$\delta = \frac{1}{n-1}\sum_{i=1}^{n-1}\Big(\phi(x_{\sigma(i+1)}) - \phi(x_{\sigma(i)})\Big) \tag{37}$$

and the threshold values 0.25, 0.75, 0.75 are obtained by averaging each pair of subsequent relative strength values from Table 1 (third column).

Being based only on preference degrees associated to each alternative, the second approach is insensible to the level of consistency of the original FPR. Moreover, any transformations introduced in the calculation of such degrees of preference (according to the methods defined in section 1.6), is considered in the selection of the ranking symbols too.

Example 8. *From the FPR resulting from Example 6, after having generated the degree of preference of each alternative in terms of Net Flow, as seen in Example 7, the ordering array of available alternatives is:* $O = (5, 2, 4, 1, 3)$. *By applying equation (37) on such degrees of preferences we obtain* $\delta = 1.25$. *Basing on equation (36) we can then obtain the fuzzy ranking of available alternatives as:* $R = (x_4 \gg x_2 \approx x_5 \geq x_3 > x_1)$.

2.5 Partial and Multiple Fuzzy Rankings

As specified in section 2.2, each available alternative appears at most once in a fuzzy ranking so *partial rankings* i.e. rankings involving only k alternatives with $k < n$ are admitted. The exclusion of one or more alternatives from a fuzzy ranking means that the expert who defined the ranking is unable to evaluate such alternatives or she considers them incomparable to the others.

In such cases, the transformation methods defined in section 2.3 produce incomplete FPRs. In particular, if R is a partial fuzzy ranking on the set X and $P = (p_{ij})$ is the corresponding FPR obtained with equations (32) or (33)-(34), for any $x_i \in X$ not included in R, the corresponding elements p_{ij} and

p_{ji} remain undefined for any $j \in \{1, \dots, n\}$. As explained in 1.7 this is an *ignorance* situation that can be solved with specific methods based on the injection of seed values and their subsequent refinement to make them as consistent as possible with other values.

Example 9. *Let $R = (x_4 \gg x_5 \approx x_2 > x_1)$ be a partial fuzzy ranking on the set $X = \{x_1, \dots, x_5\}$, using equation (33) with relative strength values coming from Table 1, we obtain the fractional rank of each alternative involved in R as: $r(x_1) = 4,\ r(x_2) = 3,\ r(x_4) = 1,\ r(x_5) = 3$. The fractional rank of x_3 is undefined given that it does not appear in R. According to equation (34), it is then possible to generate the corresponding FPR P as follows:*

$$P = \begin{pmatrix} 0.5 & 0.33 & - & 0 & 0.33 \\ 0.67 & 0.5 & - & 0.17 & 0.5 \\ - & - & - & - & - \\ 1 & 0.83 & - & 0.5 & 0.83 \\ 0.67 & 0.5 & - & 0.17 & 0.5 \end{pmatrix}.$$

The third row and the third column of P are completely undefined because no information has been provided on x_3. To complete P it is possible to inject seed values coming from other experts or similar alternatives according to section 1.7. The simpler (and rougher) method is to set undefined preferences to 0.5 assuming the indifference between x_3 and any other alternative and then iterate equations (22)-(23) until convergence obtaining the following updated version of P:

$$P = \begin{pmatrix} 0.5 & 0.33 & 0.33 & 0 & 0.33 \\ 0.67 & 0.5 & 0.47 & 0.17 & 0.5 \\ 0.67 & 0.53 & 0.5 & 0.27 & 0.53 \\ 1 & 0.83 & 0.73 & 0.5 & 0.83 \\ 0.67 & 0.5 & 0.47 & 0.17 & 0.5 \end{pmatrix}.$$

In order to make more evident the "artificial" evaluation made of alternative x_3 it is possible to convert P back to a fuzzy ranking by calculating the degree of preference of alternatives in terms of Net Flow according to equation (18) *as follows:* $\phi_{NF}(x_1) = -2$*;* $\phi_{NF}(x_2) = -0.4$*;* $\phi_{NF}(x_3) = 0$*;* $\phi_{NF}(x_4) = 2.8$*;* $\phi_{NF}(x_5) = -0.4$*. The ordering array of alternatives is then:* $O = (5, 3, 2, 1, 4)$*. By applying equation (37) we obtain* $\delta = 1.2$*. Basing on equation (36) we can then obtain the fuzzy ranking as follows:* $R = (x_4 \gg x_3 \geq x_2 \approx x_5 > x_1)$*.*

As anticipated in section 2.2, experts may be allowed to provide *multiple fuzzy rankings*: sets of partial fuzzy rankings $R^1, \ldots, R^l$ interesting disjoint subsets of X i.e. so that if an alternative $x_i \in X$ appears in a component fuzzy ranking R^j with $j \in \{1, .., l\}$, then x_i does not appear in any other component ranking R^k with $k \in \{1, .., l\} \setminus \{j\}$. The use of multiple fuzzy rankings allows experts to deal with subsets of alternatives they consider as mutually incomparable.

To simplify the notation we can represent a multiple fuzzy ranking within a single sequence $R = (x_{\sigma(1)} \; s_1 \; x_{\sigma(2)} \; \ldots \; x_{\sigma(k-1)} \; s_{k-1} \; x_{\sigma(k)})$ where terms in even positions belong to the upgraded set of symbols $S \cup \{\wedge\}$. The additional symbol $\wedge$ is used to interlock the component rankings $R^1, \ldots, R^l$ interesting disjoint subsets of X. Also in this case each alternative appears at most once in the ranking although partial rankings are admitted.

Example 10. *The fuzzy ranking* $R = (x_4 \gg x_1 \wedge x_2 \geq x_3 > x_5)$ *defined on* $X = \{x_1, \ldots, x_5\}$ *states that the fourth alternative is much better than the fifth one and that the second one is a little better than the third one that, in turn, is better than the first one. Moreover it manifests the expert's inability to compare alternatives coming from the subset* $\{x_1, x_4\}$ *with alternatives coming from* $\{x_2, x_3, x_5\}$*.*

To obtain a FPR P from a multiple fuzzy ranking R it is enough to iterate equations (32) or (33)-(34) on any component ranking $R^1, \ldots, R^l$ of R and merge the obtained FPRs $P^1, \ldots, P^l$. Being $R^1, \ldots, R^l$ partial fuzzy rankings interesting disjoint subsets of X, for any pair of alternatives $x_i, x_j \in X$ there exist at most one FPR $P^k = (p^k_{ij})$ with $k \in \{1, \ldots, l\}$ so that p^k_{ij} is defined.

For this reason, any element p_{ij} of the overall FPR P can be obtained from the elements of $P^1, \dots, P^l$ as follows:

$$p_{ij} = p_{ij}^k \colon k \in \{1, \dots, l\}, p_{ij}^k \text{ is defined.} \tag{38}$$

When for some $i, j \in \{1, \dots, n\}$, p_{ij}^k is undefined for any $k \in \{1, \dots, l\}$ then p_{ij} remains undefined too. This case happens when x_i and x_j only appear in different component rankings of R or when either x_i or x_j do not appear at all in any component ranking of R. In particular, the latter case happens when the multiple ranking only interests a subset of alternatives of X i.e. when it is also a partial ranking.

Example 11. *The multiple fuzzy ranking $R = (x_4 \gg x_1 \wedge x_2 \geq x_3 > x_5)$ coming from the previous example can be split in the two component rankings $R^1 = (x_4 \gg x_1)$ and $R^2 = (x_2 \geq x_3 > x_5)$. Applying equation (33) we obtain that $r(x_1) = 3$, $r(x_2) = 1$ from R^1 and $r(x_2) = 1$, $r(x_3) = 1.5$, $r(x_5) = 2.5$ from R^2. Applying equation (34) on such fractional ranks we then obtain the following FPRs:*

$$P^1 = \begin{pmatrix} 0.5 & - & - & 0 & - \\ - & - & - & - & - \\ - & - & - & - & - \\ 1 & - & - & 0.5 & - \\ - & - & - & - & - \end{pmatrix}; P^2 = \begin{pmatrix} - & - & - & - & - \\ - & 0.5 & 0.67 & - & 1 \\ - & 0.33 & 0.5 & - & 0.83 \\ - & - & - & - & - \\ - & 0 & 0.17 & - & 0.5 \end{pmatrix}.$$

Merging P^1 and P^2 through equation (38) the following FPR is obtained:

$$P = \begin{pmatrix} 0.5 & - & - & 0 & - \\ - & 0.5 & 0.67 & - & 1 \\ - & 0.33 & 0.5 & - & 0.83 \\ 1 & - & - & 0.5 & - \\ - & 0 & 0.17 & - & 0.5 \end{pmatrix}.$$

As it can be seen, preference values between alternatives from $\{x_1, x_4\}$ (that are referenced in R^1) and alternatives from $\{x_2, x_3, x_5\}$ (that are referenced *in R^2) remain undefined. As for Example 9, also in this case it is possible to estimate missing values with one of the methods proposed in section 1.7.*

2.6 Similarity Between Fuzzy Rankings

In order to assess the level of agreement between experts' opinions, as for ordinal rankings, it is useful to define similarity measures also between fuzzy rankings. A feasible approach for that is to extend to fuzzy rankings the two similarity measures defined in section 2.1.

Let R_i and R_j be two fuzzy rankings defined by the experts $e_i, e_j \in E$ on the same set X, the Kendall's rank correlation coefficient defined by equation (29) can be applied on R_i and R_j by computing the number c_{ij} of concordant pairs and the number d_{ij} of discordant pairs. Indeed, to take ties and partial rankings into account, it is needed to redefine c_{ij} and d_{ij} based on the notion of fractional rank defined in section 2.3.

If $r_i(x_k)$ denotes the fractional rank of an alternative $x_k \in X$ in a fuzzy ranking R_i and $\delta r^i_{kl} = r_i(x_k) - r_i(x_l)$ for $x_k, x_l \in X$, we can say that (x_k, x_l) is a concordant pair between R_i and R_j if both alternatives appear in both rankings and the condition $\delta r^i_{kl} \cdot \delta r^j_{kl} > 0$ or $\delta r^i_{kl} = \delta r^j_{kl} = 0$ is verified (i.e. δr^i_{kl} and δr^j_{kl} are both positive, both negative or both 0). Conversely, (x_k, x_l) is a discordant pair if both alternatives appear in R_i and R_j but the preceding condition is not met (i.e. δr^i_{kl} and δr^j_{kl} are one positive and the other negative or one equal to 0 and the other different from 0). Based on c_{ij} and d_{ij} we can define the *Kendall's correlation coefficient for fuzzy rankings* as follows:

$$\tau(R_i, R_j) = \frac{2(c_{ij} - d_{ij})}{kmax_{ij}(kmax_{ij} - 1)} \tag{39}$$

where $kmax_{ij} = \max(k_i, k_j)$ while k_i and k_j are the number of alternatives involved, respectively, in R_i and R_j (with $k_i, k_j \leq n$).

Example 12. *Let $R_1, \dots, R_5$ be fuzzy rankings defined on $X = \{x_1, \dots, x_5\}$ as reported in the first column of Table 2, using the relative strength values from Table 1 in equation (33), we obtain, for each ranking R_i, the fractional ranks $r_i(x_1), \dots, r_i(x_5)$ of any alternative of X as reported in columns 2-6 of Table 2 for $i \in \{1, \dots, 5\}$. Then, exploiting the definition of concordant and discordant pairs previously reported, we obtain: $c_{1,2} = 6$, $c_{1,3} = 10$, $c_{1,4} = 6$, $c_{1,5} = 1$, $d_{1,2} = 4$, $d_{1,3} = 0$, $d_{1,4} = 0$, $d_{1,5} = 9$. Basing on such values and considering that $kmax_{ij} = 5$ for $i, j \in \{1, \dots, 5\}$ we obtain from equation (39): $\tau(R_1, R_2) = 0.2$ (weak positive correlation), $\tau(R_1, R_3) = 1$ (equivalence), $\tau(R_1, R_4) = 0.6$ (moderate positive correlation), $\tau(R_1, R_5) = -0.8$ (strong negative correlation).*

Ranking	$r_i(x_1)$	$r_i(x_2)$	$r_i(x_3)$	$r_i(x_4)$	$r_i(x_5)$
$R_1 = (x_4 \gg x_5 \approx x_2 \geq x_3 > x_1)$	4.5	3	3.5	1	3
$R_2 = (x_5 > x_4 \geq x_3 \geq x_2 \approx x_1)$	3	3	2.5	2	1
$R_3 = (x_4 \geq x_5 \approx x_2 \gg x_3 \geq x_1)$	4	1.5	3.5	1	1.5
$R_4 = (x_4 \gg x_5 \approx x_2 > x_1)$	4	3	–	1	3
$R_5 = (x_3 \approx x_1 \geq x_5 \gg x_4 \geq x_5)$	1	1.5	1	3.5	4

Table 2. Five sample fuzzy rankings and the fractional rank of each involved alternative

A limit of the Kendall's correlation coefficient is that it considers only the position of alternatives in the ranking disregarding the preference gaps quantified by the separators. In Example 12, R_1 and R_3 are considered as equivalent even if, by looking at the separators used, we can see that the experts' beliefs captured by the two rankings are quite different. In fact the preference gap between x_4 and x_5 is wide in R_1 and thin in R_3 while the preference gap between x_2 and x_3 is thin in R_1 and wide in R_3.

To take separators into account when computing the similarity between fuzzy rankings, we introduce the *Spearman's correlation coefficient for fuzzy rankings* as follows:

$$\rho(R_i, R_j) = \frac{\sum_{k=1}^{n}(r_i(x_k) - \overline{r_\imath})(r_j(x_k) - \overline{r_\jmath})}{\sqrt{\sum_{k=1}^{n}(r_i(x_k) - \overline{r_\imath})^2}\sqrt{\sum_{k=1}^{n}(r_j(x_k) - \overline{r_\jmath})^2}} \tag{40}$$

where $\overline{r_\imath} = \frac{1}{n}\sum_{k=1}^{n} r_i(x_k)$ is the average fractional rank extracted from R_i and $\overline{r_\jmath}$ is the average fractional rank extracted from R^j in the same way.

Differently from the Kendall's correlation coefficient for fuzzy rankings, the Spearman's one cannot be directly obtained as an extension of equation (30) because, according to [46], such equation is inapplicable in case of ties. So equation (40) have been obtained from an alternative formulation of the Spearman's rank correlation coefficient defined in [48] as the covariance of two statistical variables divided by the product of their standard deviations where the values are converted in ranks before calculation.

Example 13. *Let $R_1, \dots, R_5$ be the fuzzy rankings defined in Example 12 and summarized in Table 2 with their fractional ranks. By applying equation (40) we obtain the following values for the Spearman's correlation coefficient: $\rho(R_1, R_2) = 0.41$ (moderate positive correlation), $\rho(R_1, R_3) = 0.83$ (strong positive correlation), $\rho(R_1, R_4) = 0.97$ (very strong positive correlation), $\rho(R_1, R_5) = -0.68$ (moderate negative correlation).*

By looking at the results of Example 13, it can be seen that R_1 and R_3 are only strongly correlated according to the Spearman's coefficient rather than equivalent as in the previous case. This happens because the Spearman's coefficient, being based on differences between fractional ranks, also takes into account the preference gaps quantified by the separators that are used within the fuzzy ranking.

2.7 Comparison with Related Works

To the best of our knowledge, the concept of fuzzy ranking is quite new. Nevertheless, an alternative formulation has been only recently proposed in [49] as a generalization of crisp rankings. While in a crisp ranking each object is assigned just one position, in the fuzzy ranking model defined in [49], the same object may be assigned to many positions with different degrees of membership. So, to characterize it, an $n \times n$ ordering matrix R is used whose generic element $r_{ij} \in [0,1]$ denotes the membership degree of the i-th object to the j-th position and $\sum_{i=1}^{n} r_{ij} = \sum_{j=1}^{n} r_{ij} = 1$ for all $i, j \in \{1, \ldots, n\}$.

The main difference with respect our model resides in the way the ranking concept is *fuzzyfied*. Instead of allowing the same object belong to multiple positions, in fact, our model allows to extend or contract the gap between subsequent positions to reinforce or weaken the ordering relation. As well as being more useful to support preferences expression in GDM, our approach also allows the use of a more compact and user-friendly notation for rankings definition. The definition of an ordering matrix, like that needed for the model described in [49] is in fact quite difficult and comparable to the direct definition of a FPR, nullifying in this way any advantage carried out by the adoption of an alternative model.

In [50, 51] *Linguistic Preference Relations* (LPRs) have been defined as an alternative preference model with respect to FPRs. In LPRs, the relative preference of each alternative with respect to each other is expressed with a linguistic term rather than with a membership degree in [0,1]. A LPR can be so represented with an $n \times n$ matrix $P = (p_{ij})$ where each element p_{ij} states the linguistically assessed preference degree of the alternative x_i over x_j.

Similarly to LPRs, fuzzy rankings allow to specify fuzzy statements about pairs of alternatives, differently from LPRs (where a linguistic term must be chosen for every pair of alternatives), in fuzzy rankings a fuzzy statement is specified only for a subset of all possible pairs i.e. only for alternatives that are adjacent in the ranking. On one hand, this allows fuzzy rankings to adopt a more compact and meaningful notation; on the other hand, it is possible to easily infer missing preferences by avoiding inconsistencies. Moreover, while fuzzy rankings can be transformed in FPRs and processed with standard GDM methods and tools (discussed in section 1), LPRs need specific fuzzy extensions of such methods and tools.

A topic quite related to fuzzy rankings is that of fuzzy numbers ranking. How to rank fuzzy numbers is an important problem in DM and GDM, and is particularly felt when experts use fuzzy estimates (maybe expressed in form of linguistic terms) to specify their preferences. According to [52], more than 30 fuzzy ranking indices have been proposed since 1976 for this purpose. By directly using fuzzy rankings instead of fuzzy estimates to specify preferences can be considered as a convenient and user-friendly method to overcome the fuzzy numbers ranking issue.

In [53, 54] the Fuzzy SQL (FSQL) language has been proposed to handle fuzzy information within databases. In order to perform queries involving fuzzy quantities, such language introduces several *fuzzy comparators* like: $F=$ (fuzzy equal than), $F<>$ (fuzzy different to), $F>$ (fuzzy greater than), $F>=$ (fuzzy greater or equal than), $F<$ (fuzzy less than), $F<=$ (fuzzy less or equal than), $F\gg$ (fuzzy much greater than), $F\ll$ (fuzzy much less than) where each comparator is associated to an algorithm able to compare fuzzy numbers as those used for ranking.

As it can be noted, there is a substantial similarity between symbols used by fuzzy comparators and those adopted by fuzzy rankings. Nevertheless, fuzzy rankings use such symbols to state fuzzy relations about crisp objects rather than to assess if a crisp relation exists between fuzzy quantities. For this reason, even if syntactically similar, the semantics under these symbols is very different.

Chapter 3

A Fuzzy GDM Model Guided by Social Influence

A promising research area in GDM is the study of interpersonal influence and its impact on the evolution of experts' opinions. As seen in section 1, in conventional GDM models, a group of experts express their preferences on a finite set of alternatives, preferences are aggregated and the best alternative, satisfying the majority of experts, is selected. Nevertheless, in real situations, experts form their opinions in a complex interpersonal environment where preferences are liable to change due to social influence.

In fact, experts are usually let free to interact and discuss each other exchanging opinions and information. During these interactions, experts with wider background, experience and knowledge are capable of influencing other experts. So, after a discussion, the preferences of such experts may undergo a modification due to social influence.

To manage the effects of social influence in GDM, we propose in this section a *Social Influence-Guided* GDM model based on interpersonal trust. The assumption is that, the more an expert trusts another expert, the more her opinion is influenced by him. Elaborating on the definitions given in [55], the concept of trust is interpreted as *the belief of an expert in the capability of another expert in finding the correct solution to a given problem.*

The proposed model adopts fuzzy rankings, defined in chapter 2, to collect both experts' preferences on available alternatives and trust statements on other experts. Starting from collected information, possibly incomplete, the configuration and the strengths of interpersonal influences are evaluated and represented through a *Social Influence Network* (SIN). The SIN, in its turn, is used to let the opinions expressed by each expert be completed (if partial) and evolved over time through the incorporation of elements captured from the opinion of trusted experts. The process then iterates until the convergence toward a shared solution to the GDM problem is reached.

After having introduced background concepts on social influence and related theories, the proposed model is outlined and described in each step. The advantages of the proposed model with respect to other existing models are then presented as well as the results of an in silico simulation that also illustrates the opinions evolution process and its convergence properties.

3.1 Theory of Social Influence and Opinion Change

Influence modelling and the appraisal of its effect on opinion change has been studied in [56, 57]. Influence is capable of playing a key role in GDM too but, despite that, the introduction of GDM models that takes into account social influence have just recently been proposed [58, 59]. According to [56, 57], the influence can be modelled through a so-called *Social Influence Network* (SIN): a directed graph between the set of experts E and where each arc (e_i, e_j) has a weight $w_{ij} \in [0,1]$ that represents the strength of the influence of the j-th expert on the i-th one. Figure 6 shows an example of SIN.

A SIN involving a set of experts $E = \{e_1, \dots, e_m\}$ can be summarized by an $m \times m$ fuzzy adjacency matrix $W = (w_{ij})$. In [56] it was suggested that the weights $w_{i1}, \dots, w_{im}$ are directly chosen by the expert $e_i \in E$ before she is informed of the preferences expressed by the others, on the basis of the relative importance she assigns to the opinion of the various experts, including himself. Selected weights must verify the normalization property so that:

$$\sum_{j=1}^{m} w_{ij} = 1 \ \ \forall\ i \in \{1, \dots, m\} \tag{41}$$

If $y^{(1)}$ is an $m \times 1$ vector representing the initial experts' opinions on a given alternative, it is supposed that, after having interacted, this opinion vector will change to $y^{(2)} = Wy^{(1)}$ due to interpersonal influence. If we suppose that each expert is informed that the others have changed their opinion, it is reasonable to expect that the expert will change again her opinion according to the same principle. By iterating the process, it is possible to obtain the experts' opinion after t interactions as:

$$y^{(t)} = Wy^{(t-1)}. \tag{42}$$

In [56] it was demonstrated that, if there exists a positive integer t so that every element in at least one column of W^t is positive, then the m opinions are expected to converge to the same value. In [57] it was suggested to also specify the susceptibility of each expert e_i to interpersonal influence as $a_{ii} \in [0,1]$. Then, being $y^{(1)}$ the initial experts' opinions, their opinions after t interactions are obtained iteratively as:

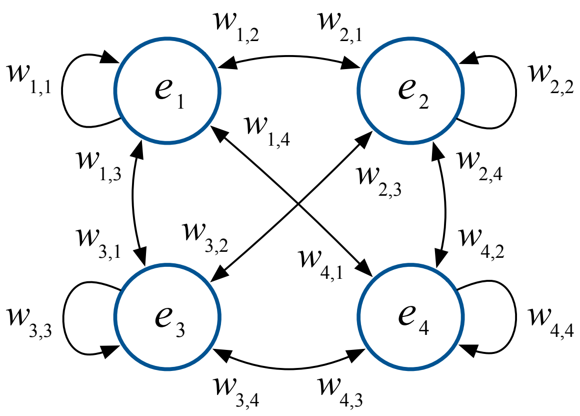

Figure 6. A sample SIN composed by 4 nodes

$$y^{(t)} = AWy^{(t-1)} + (I - A)y^{(1)} \tag{43}$$

where $\mathrm{A} = diag(a_{11}, \dots, a_{mm})$ and I is the $m \times m$ identity matrix. In other words, at each time, the current opinion of an expert is obtained as a linear combination of her initial opinion and the influenced opinion she had at the time immediately preceding. In [57] it was demonstrated that, if the matrix $I - AW$ is non-singular and $y^{(\infty)} = \lim_{t \to \infty} y^{(t)}$ exists (i.e. the process reaches an equilibrium), then:

$$y^{(\infty)} = (I - AW)^{-1}(I - A)y^{(1)}. \tag{44}$$

In [58], equations (43)-(44) have been applied for the first time in a GDM process where the experts provide opinions on a set $X = \{x_1, \dots, x_n\}$ of alternatives rather than on just one. For each expert e_i, the initial degree of preference $y_{ij}^{(1)}$ on each alternative x_j is calculated starting from expert's individual FPR via the application of the QGDD metric (as defined in section 1.6) to all preference values of the *j*-th row of the corresponding FPR.

Then, the influence model is applied on each column of the $m \times n$ matrix $Y^{(1)} = \left(y_{ij}^{(1)}\right)$ by extending equation (44) to matrices and obtaining that: $Y^{(\infty)} = (I - AW)^{-1}(I - A)Y^{(1)}$ where the *i*-th row of $Y^{(\infty)}$ represents the preferences of the expert e_i after having introjected the opinions of her peers. Eventually, influenced preferences are aggregated, available alternatives are ranked and the final solution obtained.

3.2 Outline of the Proposed Model

Based on the works described in the preceding section, the proposed model is aimed at taking into account social influence within a GDM process both in general and, especially, in presence of incomplete information. The research assumptions on which the model is built are two: experts influence each other and the more an expert trusts in the capability of another expert, the more her opinion is influenced by the trusted expert.

To make the model immediately applicable in practice, fuzzy rankings (defined in chapter 2) have been adopted for preference modeling since they are user friendly and less vulnerable to inconstancies than FPRs. The same model is used to collect opinions on alternatives as well as trust statements on experts. Given a set of experts $E = \{e_1, \dots, e_m\}$ and a set of alternatives $X = \{x_1, \dots, x_n\}$, the model works through the following steps:

1. *opinions collection*: each expert $e_i \in E$ specifies her preferences about alternatives in X in a (possibly partial or multiple) fuzzy ranking R_i;
2. *trust statements collection*: each expert $e_i \in E$ specifies the trust she has in all experts that belong to E (including himself) in a (possibly partial or multiple) fuzzy ranking R_i^e;
3. *fuzzy ranking conversion*: fuzzy rankings R_i and R_i^e are converted into the (possibly incomplete) individual FPRs P_i and P_i^e for $i \in \{1, \dots, m\}$;
4. *social influence network generation*: all FPRs P_i^e for $i \in \{1, \dots, m\}$, representing trust degrees between experts, are used to generate a SIN characterized by the $m \times m$ fuzzy adjacency matrix W;
5. *missing preferences estimation*: any individual FPR P_i for $i \in \{1, \dots, m\}$, in presence of missing information, is completed by injecting values from other FPRs according to influence information gathered by the SIN;
6. *influence-guided preferences evolution*: to simulate the effects of experts' interpersonal influence, any individual FPR P_i for $i \in \{1, \dots, m\}$, once completed, is updated according to the SIN until convergence;
7. *preferences aggregation*: the individual FPRs P_i for $i \in \{1, \dots, m\}$ once updated according to the previous step, are aggregated through OWA to obtain the collective FPR P.
8. *alternative selection*: the dominance degree $\phi(x_i)$ is estimated for each alternative $x_i \in X$ according to P, then the alternatives are ranked from the best to the worst and the first one is selected.

The information flow among the described steps is summarized in Figure 7 while the next sections provide details on each step. In particular section 3.3 deals with the collection of opinions and trust statements, their conversion into FPRs and the subsequent generation of the SIN (steps 1-4); section 3.4 explains how the generated SIN is applied to estimate missing preferences (step 5); section 3.5 deals with the application of the influence model on obtained FPRs, their aggregation and alternative selection (steps 6-8).

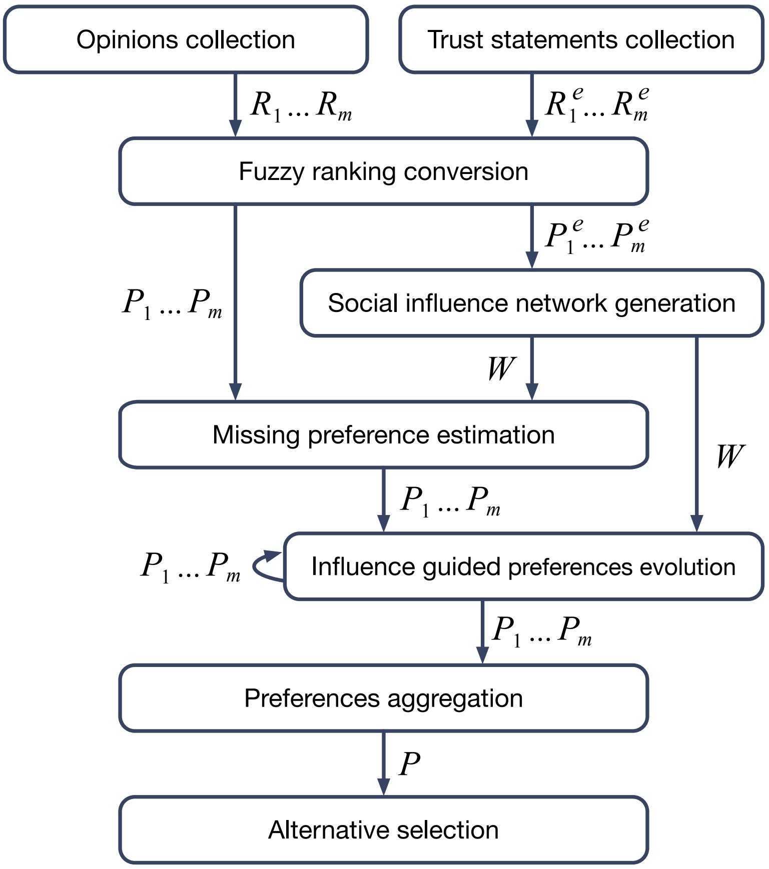

Figure 7. The information flow between model steps

3.3 FPRs and SIN Generation

Fuzzy rankings are used in our model to let experts express their opinion with respect to (a subset of) alternatives as well as their trust on (a subset of) experts. More formally, each expert $e_k \in E$ provides a fuzzy ranking R_k on the set of alternatives in X and a fuzzy ranking R_k^e on the set of experts E (including himself). Starting from R_k and R_k^e, by applying equation (34), the corresponding (incomplete) FPRs P_k and P_k^e are computed and taken forward to the next steps.

Example 14. *Let us suppose that we have a set $X = \{x_1, x_2, x_3, x_4, x_5\}$ of alternatives and a set $E = \{e_1, e_2, e_3\}$ of experts, that expert e_1 provides the fuzzy ranking of alternatives: $R_1 = x_4 \gg x_5 \approx x_2 \geq x_1$ and the fuzzy ranking of experts: $R_1^e = e_2 \gg e_1 \approx e_3$. In R_1 the expert states that the alternative x_4 is much better than x_5 and x_2 that, in their turn, are a little better than x_1. In R_1^e the expert states that she thinks that the expert e_2 is much more trustable than both e_1 (himself) and e_3. Starting from R_1 and R_1^e, through equations (33)-(34), the following corresponding FPRs are obtained:*

$$P_1 = \begin{pmatrix} 0.5 & 0.4 & - & 0 & 0.4 \\ 0.6 & 0.5 & - & 0.1 & 0.5 \\ - & - & - & - & - \\ 1 & 0.9 & - & 0.5 & 0.9 \\ 0.6 & 0.5 & - & 0.1 & 0.5 \end{pmatrix} ; \; P_1^e = \begin{pmatrix} 0.5 & 0 & 0.5 \\ 1 & 0.5 & 1 \\ 0.5 & 0 & 0.5 \end{pmatrix}.$$

The opinions on experts collected in $P_k^e = (p_{ij}^{ek})$ for $k \in \{1, \dots, m\}$ are used to generate a SIN. As explained in 3.1, a SIN is characterized by a fuzzy adjacency matrix $W = (w_{kl})$ where each element $w_{kl} \in [0,1]$ represents the strength of the influence of the l-th expert on the k-th one for $k, l \in \{1, \dots, m\}$. So, the elements of k-th row of W can be obtained from P_k^e through FPR measures defined in section 1.6 like QGDD. Moreover, to comply the SIN property so that $\sum_{i=1}^{m} w_{ki} = 1$, a normalization step is needed as follows:

$$w_{kl} = \frac{\phi_k(e_l)}{\sum_{i=1}^{m} \phi_k(e_i)} \tag{45}$$

where:

$$\phi_k(e_l) = OWA_Q(p_{lj}^{ek};\ j = 1, \dots, m\text{: } p_{lj}^{ek} \text{ is defined}). \tag{46}$$

Undefined elements of P_k^e are not considered in equation (46); when the l-th row of P_k^e is undefined (i.e. when e_k expresses no preferences on e_l) $\phi_k(e_l) = 0$; in the special case which the k-th expert only trusts himself, we obtain via equations (45)-(46): $w_{kl} = 0$ for $k \neq l$ and $w_{kk} = 1$ meaning that the expert is not influenced by any other.

Example 15. *Let X, E, R_1^e and P_1^e be as reported in Example 14, by applying equations (45)-(46) with values from P_1^e and using the fuzzy quantifier* (0,1) *corresponding to the linguistic label "much" (see Figure 3) to guide the OWA operator, the obtained SIN weights referring to the expert e_1 are: $w_{1,1} = 0.17$; $w_{1,2} = 0.67$; $w_{1,3} = 0.17$. If we suppose that the experts e_2 and e_3 define the following fuzzy ranking of experts: $R_2^e = e_1 \approx e_2 \gg e_3$ and $R_3^e = e_3 > e_2 > e_1$, then it is possible to obtain the SIN represented by the following matrix:*

$$W = \begin{pmatrix} 0.17 & 0.67 & 0.17 \\ 0.5 & 0.5 & 0 \\ 0.08 & 0.33 & 0.85 \end{pmatrix}.$$

Being m the number of experts and n the number of alternatives, the time complexity of the whole FPRs generation step is $\mathcal{O}(m \cdot n^2)$. Moreover, assuming that OWA uses state-of-the-art sorting algorithms, the overall time complexity of the SIN generation step is $\mathcal{O}(m \cdot n^2 \log n)$.

3.4 Using Social Influence to Estimate Missing Preferences

When some experts express their opinions only on a subset of alternatives, incomplete FPRs are generated through equations (33)-(34). In particular, if the i-th alternative does not appear in a given fuzzy ranking, then both the i-th row and the i-th column of the corresponding FPR remain undefined (e.g. alternative x_3 in Example 14). As seen in section 1.7, this is considered an *ignorance* situation that can be solved by selecting *seed values* to initialize the missing preferences and by iterating the equations (22)-(23) or (27)-(28) until the convergence is reached and the final estimates obtained.

Several methods have been proposed so far to obtain seed values. Here we propose to obtain seed values from preferences provided by the experts that are trusted by the one whose FPR has to be completed. This is to say that, when an expert is asked to evaluate an unknown alternative, she forms her judgment using the opinion of experts she trusts.

Based on the generated SIN, a missing preference p_{ij}^k of an FPR P_k coming from e_k is estimated through the I-IOWA operator (defined in 1.5) where the preferences to aggregate come from all the defined FPRs P_l with $l \in \{1, \dots, m\}$ while the importance degrees come from W and represent the trust degree of e_k on each expert of E. More formally, basing on equation (16), a missing preference p_{ij}^k is estimated as follows:

$$\varepsilon(p_{ij}^k) = \textit{I-IOWA}_Q\left((p_{ij}^l, w_{kl});\ l = 1, \dots, m\text{: } p_{ij}^l \text{ is defined}\right) \tag{47}$$

Undefined elements of P_l for $l \in \{1, \dots, m\}$ are not considered in equation (47). If seed values for some preferences are still missing (e.g. when the same preferences are missing in the FPRs of any trusted expert), then the estimation process based on equation (47) is repeated on FPRs injected with estimated values. The process is iterated until no additional seed values can be calculated. Then, the final estimates are computed through the iterative application of equations (22)-(23) or (27)-(28) until convergence is reached.

In some cases it is possible that some FPR value still remain undefined. Given an FPR P_k and an alternative $x_i \in X$, when none of the experts (directly or indirectly) trusted by e_k have an opinion on x_i i.e. when p_{ij}^l and p_{ji}^l are undefined for any $j \in \{1, \dots, n\}$ and any l so that a path (that excludes 0-weighted arcs) from e_l to e_k exists in the SIN, then both the i-th row and the i-th column of P_k remain undefined.

In case the SIN is a connected graph this means that all experts have no opinion on x_i. This suggests that the alternative is of no interest for the whole group so it can be removed from X. Conversely, in case the SIN is disconnected, it is possible that other (untrusted) experts have provided an opinion on x_i. In such cases x_i can't be removed and remaining undefined FPRs elements must be estimated through a different method among those discussed in 1.7 (e.g. through *indifference* by setting the seed value to 0.5).

Example 16. *Let X, E and P_1 be as reported in Example 14, let W be the SIN adjacency matrix calculated in Example 15 and suppose that the experts $e_2, e_3 \in E$ specify the following fuzzy rankings: $R_2 = x_4 \approx x_5 > x_3 > x_2$ and $R_3 = x_3 \approx x_5 \geq x_4 \gg x_1$. The FPRs corresponding to such fuzzy rankings, obtained through equations (33)-(34), are:*

$$P_2 = \begin{pmatrix} - & - & - & - & - \\ - & 0.50 & 0.25 & 0 & 0 \\ - & 0.75 & 0.50 & 0.25 & 0.25 \\ - & 1 & 0.75 & 0.50 & 0.50 \\ - & 1 & 0.75 & 0.50 & 0.50 \end{pmatrix}; P_3 = \begin{pmatrix} 0.50 & - & 0 & 0.10 & 0 \\ - & - & - & - & - \\ 1 & - & 0.50 & 0.60 & 0.50 \\ 0.90 & - & 0.40 & 0.50 & 0.40 \\ 1 & - & 0.50 & 0.60 & 0.50 \end{pmatrix}.$$

Seed values for missing preferences of P_1 are then generated from P_2 and P_3 basing on the first row of W: $w_{1,1} = 0.17$; $w_{1,2} = 0.67$; $w_{1,3} = 0.17$ through equation (47) and using the quantifier (0,1) to guide the I-IOWA operator. Estimated values are: $\varepsilon(p_{1,3}^1) = 0$, $\varepsilon(p_{2,3}^1) = 0.25$, $\varepsilon(p_{3,1}^1) = 1$, $\varepsilon(p_{3,2}^1) = 0.75$, $\varepsilon(p_{3,3}^1) = 0.5$, $\varepsilon(p_{3,4}^1) = 0.32$, $\varepsilon(p_{3,5}^1) = 0.3$, $\varepsilon(p_{4,3}^1) = 0.68$, $\varepsilon(p_{5,3}^1) = 0.7$. By iteratively applying equations (22)-(23) until convergence and injecting the *last estimates in P_1, the FPR coming from e_1 is completed as follows (where injected values are reported in bold):*

$$P_1 = \begin{pmatrix} 0.5 & 0.4 & \mathbf{0.19} & 0 & 0.4 \\ 0.6 & 0.5 & \mathbf{0.32} & 0.1 & 0.5 \\ \mathbf{0.81} & \mathbf{0.68} & \mathbf{0.5} & \mathbf{0.28} & \mathbf{0.59} \\ 1 & 0.9 & \mathbf{0.72} & 0.5 & 0.9 \\ 0.6 & 0.5 & \mathbf{0.41} & 0.1 & 0.5 \end{pmatrix}.$$

The time complexity of the preference estimation step is affected by the number of missing preferences, being m the number of experts and n the number of alternatives. Assuming that I-OWA uses state-of-the-art sorting algorithms, the overall time complexity of this step can be asymptotically limited by $\Omega(m \cdot n^2)$ and $\mathcal{O}(m \cdot n^3 \log n)$.

3.5 Preferences Evolution and Best Alternative Selection

To simulate the effects of social influence between experts, the individual FPRs obtained at the preceding steps are revised using the SIN generated with equations (45)-(46). The aim is to predict the final decision that will be adopted by the group of experts as a result of interaction, without the need to actually perform such interaction. To do that we apply an iterative process like that described in section 3.1 where at each step the individual FPR of each of the experts is slightly changed to take into account the influence coming from trusted experts. Differently from [58], in our model the influence model directly impacts individual FPRs rather than utility vectors obtained from them.

Being $P_k^{(1)} = \left(p_{ij}^{k(1)}\right)$ the FPR representing the initial opinion of the k-th expert with $k \in \{1, \dots, m\}$ and $i, j \in \{1, \dots, n\}$, it is possible to estimate the elements of the k-th expert's FPR after t interactions based on the SIN fuzzy adjacency matrix W as follows:

$$p_{ij}^{k(t)} = \textit{I-IOWA}_Q\left(\left(p_{ij}^{1(t-1)}, w_{k1}\right), \dots, \left(p_{ij}^{m(t-1)}, w_{km}\right)\right). \tag{48}$$

In other words, at each step, each preference is updated by composing the current preference with preferences coming from all the experts via the I-IOWA operator. The importance degree of each contribution matches the strength of the social influence coming from W. Extending the notation to matrices, we can rewrite equation (48) as follows:

$$P_k^{(t)} = \textit{I-IOWA}_Q\left(\left(P_1^{(t-1)}, w_{k1}\right), \dots, \left(P_m^{(t-1)}, w_{km}\right)\right). \tag{49}$$

Proposition. *When the fuzzy quantifier $Q = (0{,}1)$, corresponding to the label "much" (see Figure 3), is used to obtain the I-IOWA weights, it can be demonstrated that, if there exists a positive integer l so that every element in at least one column oW^lf is positive, then all the FPRs $P_k^{(t)}$ for $k \in \{1, \dots, m\}$ are expected to converge to the same FPR.*

Proof. *Combining equation (48) with the definition of the I-IOWA operator provided by equations (16)-(17), we obtain that, being $p_k^{(t)}$ a generic element belonging to the FPR $P_k^{(t)}$ for $k \in \{1, \dots, m\}$ and $t > 1$:*

$$\begin{aligned} p_k^{(t)} &= \textit{I-IOWA}_Q\left(\left(p_1^{(t-1)}, w_{k1}\right), \dots, \left(p_m^{(t-1)}, w_{km}\right)\right) \\ &= \sum_{i=1}^{m}\left(\mu_Q\left(\frac{S(i)}{S(m)}\right) - \mu_Q\left(\frac{S(i-1)}{S(m)}\right)\right) p_{\sigma(i)}^{(t-1)} \end{aligned}$$

where $S(i) = \sum_{j=1}^{i} w_{k\sigma(j)}$ and $\sigma\colon \{1, \dots m\} \to \{1, \dots m\}$ denotes a permutation function so that $w_{\sigma(i)} \geq w_{\sigma(i+1)}$ for each $i \in \{1, \dots, m\}$. Being $Q = (0{,}1)$, by substituting $a = 0$ and $b = 1$ in equation (14) we obtain: $\mu_Q(y) = \frac{y-0}{1-0} = y$ for $0 \leq y \leq 1$. Given that $S(i)$ and $S(m)$ are positive number and $S(m) \geq S(i)$ for $1 \leq i \leq m$, then we can say that $0 \leq \frac{S(i)}{S(m)} \leq 1$ so $\mu_Q\left(\frac{S(i)}{S(m)}\right) = \frac{S(i)}{S(m)}$. By substituting this in the preceding equation we obtain:

$$p_k^{(t)} = \sum_{i=1}^{m} \left(\frac{S(i)}{S(m)} - \frac{S(i-1)}{S(m)} \right) p_{\sigma(i)}^{(t-1)} = \sum_{i=1}^{m} \frac{\sum_{j=1}^{i} w_{k\sigma(j)} - \sum_{j=1}^{i-1} w_{k\sigma(j)}}{\sum_{j=1}^{m} w_{k\sigma(j)}} p_{\sigma(i)}^{(t-1)}$$
$$= \sum_{i=1}^{m} \frac{w_{k\sigma(i)}}{\sum_{j=1}^{m} w_{k\sigma(j)}} p_{\sigma(i)}^{(t-1)}.$$

Given that W is the fuzzy adjacency matrix of a SIN, thanks to equation (45) we have that: $\sum_{j=1}^{m} w_{kj} = 1$ for any $k \in \{1, \ldots, m\}$. Being σ a permutation function, $\sum_{j=1}^{m} w_{k\sigma(j)}$ simply sum the same elements in a different order so we can say that $\sum_{j=1}^{m} w_{k\sigma(j)} = 1$ too. By substituting this in the preceding equation we obtain:

$$p_k^{(t)} = \sum_{i=1}^{m} w_{k\sigma(i)} p_{\sigma(i)}^{(t-1)} = \sum_{i=1}^{m} w_{ki} p_i^{(t-1)}.$$

If we build the vector $p^{(t)} = \left(p_1^{(t)}, \ldots, p_m^{(t)} \right)^T$ including the same preference as expressed by all the m experts we can generalize the preceding equation using matrix notation as $p^{(t)} = Wp^{(t-1)} = W^{t-1}p^{(1)}$. As explained in [56]*, W can be so regarded as the one-step transition probability matrix of a Markov chain with m states and stationary transition probabilities.*
If there exists a positive integer l so that every element in at least one column of W^l is positive then the Markov chain is said regular and, thanks to the limit theorem for regular finite Markov chains [60]*, it exists a value p so that $\lim_{t \to \infty} p_k^{(t)} = p \, \forall k \in \{1, \ldots, m\}$ i.e. the preferences expressed by the m experts converge to the same value p. By extending this result (that regards a generic FPR preference) to the whole FPR, we can say that, if conditions are met, all the FPRs $P_k^{(t)}$ for $k \in \{1, \ldots, m\}$ converge to the same FPR.*

In practical applications the preferences evolution may be stopped after a fixed number of iterations or when the average absolute difference between FPRs values in two subsequent steps is under a given threshold θ i.e. when:

$$\frac{1}{m \cdot n^2} \sum_{1 \le i,j \le n;\ 1 \le k \le m} \left| p_{ij}^{k(t)} - p_{ij}^{k(t-1)} \right| \le \theta \tag{50}$$

When the stopping conditions are met, in case of lack of convergence, the obtained FPRs are aggregated through the OWA_Q operator defined in section 1.5, whose weights are initialized according to equation (15). A score value $\phi(x_i)$ is then calculated for each $x_i \in X$ through the QGDD operator defined by equation (20) and the best alternative is chosen as the result of the GDM problem. To obtain a more exhaustive and easy to understand solution to the problem, it is possible to convert the obtained score values back to a collective fuzzy ranking of alternatives through equations (36)-(37).

Example 17. *Let X, E, P_1, P_2, P_3 and W be as reported in the previous examples, using W, it is possible to complete the individual FPRs P_2 and P_3 through equation (47) as follows (injected values are represented in bold):*

$$P_2 = \begin{pmatrix} \mathbf{0.5} & \mathbf{0.56} & \mathbf{0.34} & \mathbf{0.77} & \mathbf{0.16} \\ \mathbf{0.38} & 0.5 & 0.25 & 0 & 0 \\ \mathbf{0.52} & 0.75 & 0.5 & 0.25 & 0.25 \\ \mathbf{0.86} & 1 & 0.75 & 0.5 & 0.5 \\ \mathbf{0.78} & 1 & 0.75 & 0.5 & 0.5 \end{pmatrix};$$

$$P_3 = \begin{pmatrix} 0.5 & \mathbf{0.41} & 0 & 0.1 & 0 \\ \mathbf{0.59} & \mathbf{0.5} & \mathbf{0.12} & \mathbf{0.16} & \mathbf{0.09} \\ 1 & \mathbf{0.88} & 0.5 & 0.6 & 0.5 \\ 0.9 & \mathbf{0.84} & 0.4 & 0.5 & 0.4 \\ 1 & \mathbf{0.91} & 0.5 & 0.6 & 0.5 \end{pmatrix}.$$

The completed FPRs are then updated according to equation (49) simulating the effect of social influence. The fuzzy quantifier $Q = (0{,}1)$, corresponding to the linguistic label "much", is used to guide the I-IOWA operator. The following matrices represent the evolution of P_1 after 2 and 6 iterations:

$$P_1^{(2)} = \begin{pmatrix} 0.5 & 0.51 & 0.26 & 0.07 & 0.17 \\ 0.45 & 0.5 & 0.24 & 0.04 & 0.1 \\ 0.65 & 0.76 & 0.5 & 0.31 & 0.35 \\ 0.89 & 0.56 & 0.69 & 0.5 & 0.55 \\ 0.78 & 0.9 & 0.65 & 0.45 & 0.5 \end{pmatrix};$$

$$P_1^{(6)} = \begin{pmatrix} 0.5 & 0.48 & 0.25 & 0.05 & 0.22 \\ 0.48 & 0.5 & 0.26 & 0.05 & 0.18 \\ 0.68 & 0.74 & 0.5 & 0.31 & 0.4 \\ 0.91 & 0.95 & 0.69 & 0.5 & 0.62 \\ 0.75 & 0.82 & 0.6 & 0.38 & 0.5 \end{pmatrix}.$$

After 6 iterations all individual FPRs converge to $P = P_1^{(6)} = P_2^{(6)} = P_3^{(6)}$ *that can be considered as the collective preference relation of consensus (so there is no need for aggregation). By applying equation (20), the preference degrees associated to available alternatives are:* $\phi(x_1) = 0.25$*;* $\phi(x_2) = 0.24$*;* $\phi(x_3) = 0.53$*;* $\phi(x_4) = 0.79$*;* $\phi(x_5) = 0.64$*. The best alternative is then* x_4 *which can be considered the solution of the GDM problem. Applying equations (36)-(37) it is also possible to obtain the following collective fuzzy ranking of problem alternatives:* $x_4 > x_5 > x_3 \gg x_1 \approx x_2$*.*

Being m the number of experts and n the number of alternatives, the time complexity of each iteration of preferences evolution is $\mathcal{O}(m \cdot n^3 \log n)$. Being the number of iterations limited by a constant, it can be considered as asymptotically negligible. The aggregation between FPRs (in case of lack of convergence) has a time complexity of $\mathcal{O}(m \cdot n^3 \log n)$, while the complexity of the alternative selection step is $\mathcal{O}(n^2 \log n)$.

3.6 Numerical Example

This section describes two in silico experiments of the proposed methodology aimed at illustrating its operational steps and convergence properties. Let $E = \{e_1, \dots, e_6\}$ be a set of experts that have to choose the best alternative among those available in the set $X = \{x_1, \dots, x_{10}\}$. According to the defined model, experts use fuzzy rankings to express their preferences on alternatives and their trust on other experts. Defined fuzzy rankings are reported in Table 3. As it can be seen, many experts provide incomplete information both with respect to alternatives and to other experts. For example e_1 just evaluates 7 alternatives over 10 and express her trust on 4 experts over 6.

Expert	Fuzzy rankings of alternatives	Fuzzy rankings of experts
e_1	$x_5 \gg x_7 \approx x_8 \geq x_1 \approx x_3 > x_4 \gg x_2$	$e_2 \gg e_1 > e_4 \geq e_5$
e_2	$x_{10} \approx x_6 > x_2 \geq x_1 \gg x_3 \geq x_9 \approx x_5$	$e_3 > e_2 \approx e_4 \geq e_5 > e_6$
e_3	$x_3 \approx x_5 > x_{10} \gg x_1 > x_2 > x_6 \approx x_7 \approx x_8$	$e_3 \gg e_6 \geq e_2 > e_5$
e_4	$x_6 > x_2 \geq x_1 > x_9 \approx x_5 > x_8$	$e_4 > e_3 > e_2 \approx e_1 > e_5 \approx e_6$
e_5	$x_3 > x_5 \gg x_8 > x_1 > x_{10} > x_6 > x_2$	$e_3 \geq e_5 \geq e_6 > e_1 \approx e_2$
e_6	$x_{10} \approx x_4 > x_5 \gg x_6 > x_2$	$e_6 \gg e_2 \geq e_5 > e_4$

Table 3. Collected fuzzy rankings of alternatives and experts (first case)

Applying equations (33)-(34), the fuzzy rankings on alternatives are converted into FPRs (see Table 4). As it can be seen, many elements remain undefined given the incompleteness of experts' opinion.

The same process is repeated with fuzzy rankings of experts and obtained FPRs (that are not reported for reasons of brevity) are, in turn, used to build a SIN via equations (45)-(46). It should be noted that, even if information on trust is incomplete, the SIN generation process is able to initialize any SIN weight. The obtained SIN, shown in Figure 8, can be summarized by the following fuzzy adjacency matrix:

$$W = \begin{pmatrix} 0.26 & 0.45 & 0 & 0.17 & 0.12 & 0 \\ 0 & 0.22 & 0.32 & 0.22 & 0.17 & 0.07 \\ 0 & 0.20 & 0.44 & 0 & 0.11 & 0.25 \\ 0.16 & 0.16 & 0.22 & 0.29 & 0.09 & 0.09 \\ 0.09 & 0.09 & 0.34 & 0 & 0.28 & 0.21 \\ 0 & 0.25 & 0 & 0.11 & 0.20 & 0.44 \end{pmatrix}.$$

P_1										P_2									
0.50	0.77	0.50	0.59	0.27	-	0.45	0.45	-	-	0.50	0.44	0.75	-	0.81	0.31	-	-	0.81	0.31
0.23	0.50	0.23	0.32	0.00	-	0.18	0.18	-	-	0.56	0.50	0.81	-	0.88	0.38	-	-	0.88	0.38
0.50	0.77	0.50	0.59	0.27	-	0.45	0.45	-	-	0.25	0.19	0.50	-	0.56	0.06	-	-	0.56	0.06
0.41	0.68	0.41	0.50	0.18	-	0.36	0.36	-	-	-	-	-	0.50	-	-	-	-	-	-
0.73	1.00	0.73	0.82	0.50	-	0.68	0.68	-	-	0.19	0.13	0.44	-	0.50	0.00	-	-	0.50	0.00
-	-	-	-	-	0.50	-	-	-	-	0.69	0.63	0.94	-	1.00	0.50	-	-	1.00	0.50
0.55	0.82	0.55	0.64	0.32	-	0.50	0.50	-	-	-	-	-	-	-	-	0.50	-	-	-
0.55	0.82	0.55	0.64	0.32	-	0.50	0.50	-	-	-	-	-	-	-	-	-	0.50	-	-
-	-	-	-	-	-	-	-	0.50	-	0.19	0.13	0.44	-	0.50	0.00	-	-	0.50	0.00
-	-	-	-	-	-	-	-	-	0.50	0.69	0.63	0.94	-	1.00	0.50	-	-	1.00	0.50
P_3										P_4									
0.50	0.60	0.20	-	0.20	0.70	0.70	0.70	-	0.30	0.50	0.43	-	-	0.64	0.29	-	0.79	0.64	-
0.40	0.50	0.10	-	0.10	0.60	0.60	0.60	-	0.20	0.57	0.50	-	-	0.71	0.36	-	0.86	0.71	-
0.80	0.90	0.50	-	0.50	1.00	1.00	1.00	-	0.60	-	-	0.50	-	-	-	-	-	-	-
-	-	-	0.50	-	-	-	-	-	-	-	-	-	0.50	-	-	-	-	-	-
0.80	0.90	0.50	-	0.50	1.00	1.00	1.00	-	0.60	0.36	0.29	-	-	0.50	0.14	-	0.64	0.50	-
0.30	0.40	0.00	-	0.00	0.50	0.50	0.50	-	0.10	0.71	0.64	-	-	0.86	0.50	-	1.00	0.86	-
0.30	0.40	0.00	-	0.00	0.50	0.50	0.50	-	0.10	-	-	-	-	-	-	0.50	-	-	-
0.30	0.40	0.00	-	0.00	0.50	0.50	0.50	-	0.10	0.21	0.14	-	-	0.36	0.00	-	0.50	0.36	-
-	-	-	-	-	-	-	-	0.50	-	0.36	0.29	-	-	0.50	0.14	-	0.64	0.50	-
0.70	0.80	0.40	-	0.40	0.90	0.90	0.90	-	0.50	-	-	-	-	-	-	-	-	-	0.50
P_5										P_6									
0.50	0.71	0.21	-	0.29	0.64	-	0.43	-	0.57	0.50	-	-	-	-	-	-	-	-	-
0.29	0.50	0.00	-	0.07	0.43	-	0.21	-	0.36	-	0.50	-	0.00	0.13	0.38	-	-	-	0.00
0.79	1.00	0.50	-	0.57	0.93	-	0.71	-	0.86	-	-	0.50	-	-	-	-	-	-	-
-	-	-	0.50	-	-	-	-	-	-	-	1.00	-	0.50	0.63	0.88	-	-	-	0.50
0.71	0.93	0.43	-	0.50	0.86	-	0.64	-	0.79	-	0.88	-	0.38	0.50	0.75	-	-	-	0.38
0.36	0.57	0.07	-	0.14	0.50	-	0.29	-	0.43	-	0.63	-	0.13	0.25	0.50	-	-	-	0.13
-	-	-	-	-	-	0.50	-	-	-	-	-	-	-	-	-	0.50	-	-	-
0.57	0.79	0.29	-	0.36	0.71	-	0.50	-	0.64	-	-	-	-	-	-	-	0.50	-	-
-	-	-	-	-	-	-	-	0.50	-	-	-	-	-	-	-	-	-	0.50	-
0.43	0.64	0.14	-	0.21	0.57	-	0.36	-	0.50	-	1.00	-	0.50	0.63	0.88	-	-	-	0.50

Table 4. Experts' initial opinions converted in FPRs (first case)

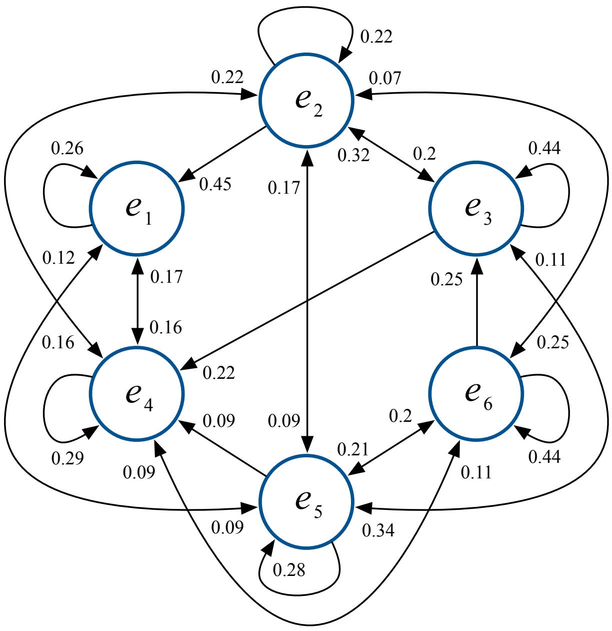

Figure 8. The generated SIN (first case)

Applying the process described in section 3.4 it is possible to estimate missing preferences injecting external seeds from trusted experts (according to the SIN) and to consolidate them through harmonization with existing preferences using the additive transitivity property. Completed FPRs are shown in Table 5. To make these results more readable, we apply equations (36)-(37) to obtain back the completed fuzzy rankings after the injection of external preferences. They are reported in Table 6.

The next step consists in executing the process described in section 3.5 to let experts' preferences evolve according to social influence. The process is expected to converge since all the elements of at least one column of W are positive. In fact, after 5 iterations, the experts' preferences converge to the same collective FPR P reported below:

P_1										P_2									
0.50	0.77	0.50	0.59	0.27	0.32	0.45	0.45	0.62	0.35	0.50	0.44	0.75	0.38	0.81	0.31	0.70	0.69	0.81	0.31
0.23	0.50	0.23	0.32	0.00	0.14	0.18	0.18	0.44	0.16	0.56	0.50	0.81	0.37	0.88	0.38	0.69	0.68	0.88	0.38
0.50	0.77	0.50	0.59	0.27	0.27	0.45	0.45	0.57	0.30	0.25	0.19	0.50	0.26	0.56	0.06	0.58	0.57	0.56	0.06
0.41	0.68	0.41	0.50	0.18	0.21	0.36	0.36	0.51	0.23	0.42	0.46	0.54	0.50	0.62	0.37	0.65	0.64	0.64	0.28
0.73	1.00	0.73	0.82	0.50	0.41	0.68	0.68	0.71	0.44	0.19	0.13	0.44	0.21	0.50	0.00	0.53	0.52	0.50	0.00
0.61	0.79	0.66	0.63	0.52	0.50	0.53	0.60	0.73	0.46	0.69	0.63	0.94	0.46	1.00	0.50	0.78	0.77	1.00	0.50
0.55	0.82	0.55	0.64	0.32	0.30	0.50	0.50	0.60	0.33	0.20	0.24	0.32	0.11	0.40	0.15	0.50	0.42	0.42	0.06
0.55	0.82	0.55	0.64	0.32	0.33	0.50	0.50	0.63	0.36	0.25	0.29	0.37	0.16	0.45	0.19	0.48	0.50	0.47	0.11
0.31	0.50	0.36	0.33	0.22	0.14	0.23	0.30	0.50	0.16	0.19	0.13	0.44	0.12	0.50	0.00	0.44	0.43	0.50	0.00
0.59	0.77	0.63	0.60	0.49	0.41	0.50	0.57	0.71	0.50	0.69	0.63	0.94	0.55	1.00	0.50	0.87	0.86	1.00	0.50

P_3										P_4									
0.50	0.60	0.20	0.32	0.20	0.70	0.70	0.70	0.63	0.30	0.50	0.43	0.42	0.50	0.64	0.29	0.60	0.79	0.64	0.34
0.40	0.50	0.10	0.21	0.10	0.60	0.60	0.60	0.52	0.20	0.57	0.50	0.38	0.47	0.71	0.36	0.57	0.86	0.71	0.30
0.80	0.90	0.50	0.54	0.50	1.00	1.00	1.00	0.84	0.60	0.58	0.62	0.50	0.59	0.62	0.53	0.69	0.79	0.69	0.42
0.48	0.62	0.26	0.50	0.30	0.67	0.65	0.65	0.64	0.32	0.46	0.50	0.38	0.50	0.50	0.41	0.57	0.67	0.57	0.30
0.80	0.90	0.50	0.53	0.50	1.00	1.00	1.00	0.84	0.60	0.36	0.29	0.38	0.47	0.50	0.14	0.57	0.64	0.50	0.30
0.30	0.40	0.00	0.16	0.00	0.50	0.50	0.50	0.47	0.10	0.71	0.64	0.47	0.56	0.86	0.50	0.66	1.00	0.86	0.39
0.30	0.40	0.00	0.11	0.00	0.50	0.50	0.50	0.42	0.10	0.37	0.40	0.28	0.37	0.40	0.31	0.50	0.57	0.47	0.20
0.30	0.40	0.00	0.11	0.00	0.50	0.50	0.50	0.42	0.10	0.21	0.14	0.21	0.30	0.36	0.00	0.40	0.50	0.36	0.13
0.24	0.38	0.02	0.09	0.06	0.43	0.41	0.41	0.50	0.08	0.36	0.29	0.24	0.33	0.50	0.14	0.43	0.64	0.50	0.16
0.70	0.80	0.40	0.52	0.40	0.90	0.90	0.90	0.83	0.50	0.66	0.70	0.58	0.67	0.70	0.61	0.77	0.87	0.77	0.50

P_5										P_6									
0.50	0.71	0.21	0.49	0.29	0.64	0.65	0.43	0.67	0.57	0.50	0.61	0.44	0.31	0.44	0.51	0.35	0.50	0.61	0.33
0.29	0.50	0.00	0.28	0.07	0.43	0.44	0.21	0.46	0.36	0.29	0.50	0.29	0.00	0.13	0.38	0.20	0.36	0.46	0.00
0.79	1.00	0.50	0.69	0.57	0.93	0.85	0.71	0.86	0.86	0.43	0.61	0.50	0.31	0.44	0.50	0.34	0.50	0.61	0.32
0.47	0.68	0.28	0.50	0.34	0.62	0.62	0.45	0.64	0.49	0.45	1.00	0.46	0.50	0.63	0.88	0.37	0.52	0.63	0.50
0.71	0.93	0.43	0.63	0.50	0.86	0.79	0.64	0.80	0.79	0.46	0.88	0.46	0.38	0.50	0.75	0.37	0.52	0.63	0.38
0.36	0.57	0.07	0.35	0.14	0.50	0.51	0.29	0.52	0.43	0.39	0.63	0.40	0.13	0.25	0.50	0.31	0.46	0.57	0.13
0.32	0.52	0.12	0.31	0.18	0.46	0.50	0.29	0.48	0.33	0.29	0.47	0.29	0.17	0.30	0.36	0.50	0.35	0.46	0.18
0.57	0.79	0.29	0.48	0.36	0.71	0.64	0.50	0.66	0.64	0.36	0.54	0.37	0.25	0.38	0.44	0.28	0.50	0.54	0.26
0.23	0.44	0.04	0.23	0.10	0.38	0.38	0.21	0.50	0.25	0.26	0.44	0.26	0.14	0.27	0.33	0.17	0.32	0.50	0.15
0.43	0.64	0.14	0.48	0.21	0.57	0.63	0.36	0.65	0.50	0.57	1.00	0.58	0.50	0.63	0.88	0.48	0.64	0.75	0.50

Table 5. Experts' opinions completed with preferences injected from trusted experts (first case)

$$P=\begin{pmatrix}0.5 & 0.58 & 0.4 & 0.39 & 0.44 & 0.51 & 0.59 & 0.61 & 0.67 & 0.36\\ 0.4 & 0.5 & 0.31 & 0.24 & 0.32 & 0.43 & 0.48 & 0.51 & 0.59 & 0.23\\ 0.57 & 0.67 & 0.5 & 0.46 & 0.51 & 0.61 & 0.69 & 0.71 & 0.71 & 0.44\\ 0.46 & 0.67 & 0.38 & 0.5 & 0.46 & 0.6 & 0.56 & 0.58 & 0.63 & 0.37\\ 0.54 & 0.68 & 0.46 & 0.45 & 0.5 & 0.6 & 0.67 & 0.69 & 0.68 & 0.42\\ 0.47 & 0.57 & 0.37 & 0.31 & 0.4 & 0.5 & 0.53 & 0.57 & 0.66 & 0.29\\ 0.3 & 0.42 & 0.2 & 0.21 & 0.23 & 0.36 & 0.5 & 0.42 & 0.45 & 0.17\\ 0.35 & 0.46 & 0.25 & 0.26 & 0.29 & 0.4 & 0.46 & 0.5 & 0.5 & 0.24\\ 0.25 & 0.35 & 0.2 & 0.17 & 0.25 & 0.27 & 0.35 & 0.38 & 0.5 & 0.12\\ 0.61 & 0.77 & 0.53 & 0.53 & 0.58 & 0.71 & 0.73 & 0.73 & 0.8 & 0.5\end{pmatrix}$$

From P, through equation (20), it is possible to calculate the degrees of preference associated to each alternative in terms of dominance degree as follows: $\phi(x_1)=0.51$, $\phi(x_2)=0.39$, $\phi(x_3)=0.6$, $\phi(x_4)=0.52$, $\phi(x_5)=0.58$, $\phi(x_6)=0.46$, $\phi(x_7)=0.31$, $\phi(x_8)=0.36$, $\phi(x_9)=0.26$, $\phi(x_1)=0.67$. So, the best alternative is x_{10}. In addition, the obtained dominance degrees can be used to generate the following collective fuzzy ranking of alternatives:

$$x_{10} \gg x_3 \geq x_5 > x_4 \geq x_1 > x_6 \gg x_2 \geq x_8 > x_7 > x_9.$$

Expert	Completed fuzzy rankings of alternatives
e_1	$x_5 > x_6 \geq x_{10} > x_7 \approx x_8 \geq x_1 \geq x_3 \gg x_4 \gg x_9 > x_2$
e_2	$x_{10} \geq x_6 \gg x_2 > x_1 > x_4 \gg x_3 \geq x_8 \geq x_5 \geq x_7 \approx x_9$
e_3	$x_3 \approx x_5 > x_{10} \gg x_4 \geq x_1 \gg x_2 \gg x_6 \approx x_7 \approx x_8 \geq x_9$
e_4	$x_{10} \geq x_6 > x_3 > x_2 \geq x_1 \geq x_4 \gg x_5 \geq x_7 \geq x_9 \gg x_8$
e_5	$x_3 > x_5 \gg x_8 > x_1 \approx x_4 > x_{10} \gg x_6 \geq x_7 > x_2 \geq x_9$
e_6	$x_{10} > x_4 > x_5 \gg x_1 \approx x_3 > x_8 \geq x_6 > x_7 > x_9 \geq x_2$

Table 6. Completed fuzzy rankings of alternatives (first case)

Figure 9 shows the evolution of the degree of preference associated to each alternative for the involved experts, which elucidates the convergence process versus the final preferences. The x-axis represents the number of performed iterations while the y-axis represents the dominance degree of each alternatives for each expert at a given iteration. Different colors correspond to different alternatives whose identifier is shown on the right. The first 5 alternatives are plotted on the left, the last 5 on the right. The figure allows to easily perceive the final ranking but also shows the process dynamics that led to the generation of the final decision. For example, it can be noticed that the most controversial alternatives are x_2 and x_6 since the convergence on them is reached later than for the other alternatives.

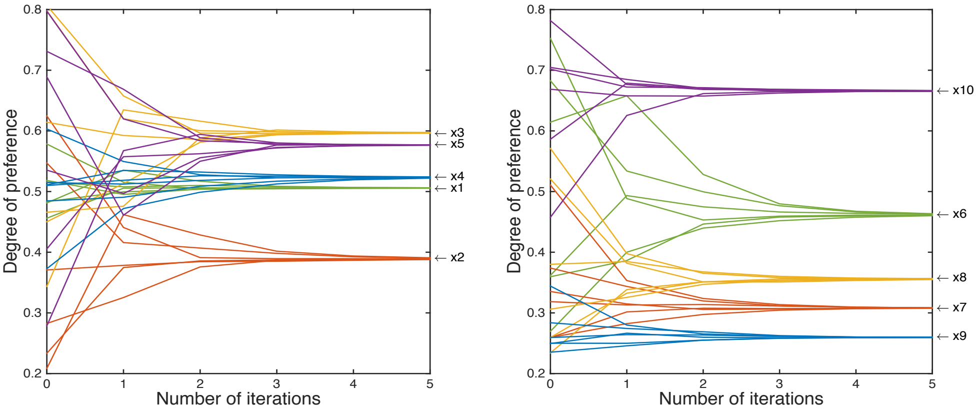

Figure 9. Evolution of preferences based on the influence model (first case)

A special case is when W does not respect the conditions for convergence. Let us suppose that the previous experts provide the same opinions about the alternatives but different fuzzy rankings about experts (as shown in Table 7). By applying equations (33)-(34), the fuzzy rankings are converted in FPRs and, then, used to build a SIN via equations (45)-(46). The SIN, shown in Figure 10, can be summarized by the following fuzzy adjacency matrix:

$$W = \begin{pmatrix} 0.61 & 0.28 & 0.11 & 0 & 0 & 0 \\ 0.17 & 0.67 & 0.17 & 0 & 0 & 0 \\ 0.33 & 0.33 & 0.33 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.67 & 0.17 & 0.17 \\ 0 & 0 & 0 & 0.17 & 0.67 & 0.17 \\ 0 & 0 & 0 & 0.5 & 0 & 0.5 \end{pmatrix}.$$

Expert	Fuzzy rankings of alternatives	Fuzzy rankings of experts
e_1	$x_5 \gg x_7 \approx x_8 \geq x_1 \approx x_3 > x_4 \gg x_2$	$e_1 > e_2 \geq e_3$
e_2	$x_{10} \approx x_6 > x_2 \geq x_1 \gg x_3 \geq x_9 \approx x_5$	$e_2 \gg e_1 \approx e_3$
e_3	$x_3 \approx x_5 > x_{10} \gg x_1 > x_2 > x_6 \approx x_7 \approx x_8$	$e_1 \approx e_2 \approx e_3$
e_4	$x_6 > x_2 \geq x_1 > x_9 \approx x_5 > x_8$	$e_4 > e_5 \approx e_6$
e_5	$x_3 > x_5 \gg x_8 > x_1 > x_{10} > x_6 > x_2$	$e_5 \geq e_4 \approx e_6$
e_6	$x_{10} \approx x_4 > x_5 \gg x_6 > x_2$	$e_4 \approx e_6 > e_5$

Table 7. Collected fuzzy rankings of alternatives and experts (second case)

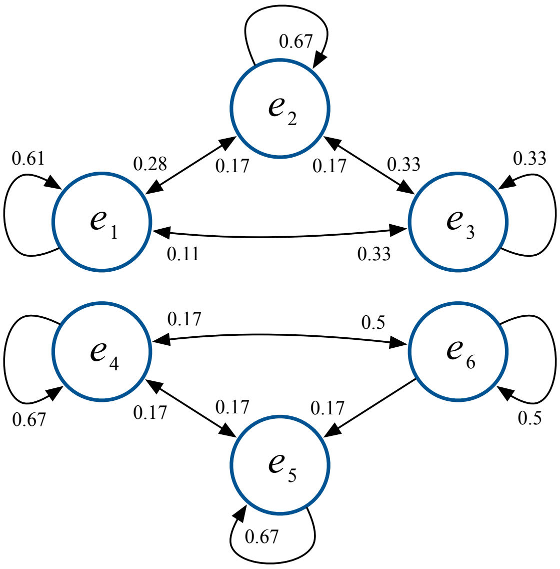

Figure 10. The generated SIN (second case)

Like in the previous case, the experts are initially in disagreement but, unlike the previous case, they grant their trust only to a small subset of colleagues so as to create two unconnected subgroups. As it can be seen from Figure 10 (but also from W), experts e_1, e_2 and e_3 do not provide trust information related experts e_4, e_5 and e_6 and vice versa, so their preferences are not mutually influenced by the model. It is easy to demonstrate that W does not meet the conditions for convergence since it is impossible to find a positive integer l so that every element in at least one column of W^l is positive. So it is expected that the influence process does not converge.

Since the fuzzy rankings on alternatives are the same as in the previous example, after conversion, the obtained FPRs are the same already shown in Table 4. Obtained FPRs are then completed according to the new SIN and used as input for the influence model. The completed FPRs converted back into fuzzy rankings are reported in Table 8. After 8 interactions, each of the two subgroups of experts reaches internal consensus on a single FPR but the FPRs obtained by the two subgroups of experts are different (the two FPRs are reported in Table 9).

Expert	Completed fuzzy rankings of alternatives
e_1	$x_{10} \geq x_5 \gg x_6 > x_7 \approx x_8 \geq x_1 \approx x_3 \gg x_4 \gg x_9 > x_2$
e_2	$x_{10} \geq x_6 \gg x_2 \geq x_1 \gg x_7 \approx x_8 \approx x_4 \geq x_3 > x_5 > x_9$
e_3	$x_3 \approx x_5 \gg x_{10} \gg x_1 \gg x_2 \geq x_4 > x_6 \approx x_7 \approx x_8 \geq x_9$
e_4	$x_6 \geq x_3 \gg x_4 \approx x_2 \geq x_1 > x_{10} > x_5 \geq x_7 \approx x_9 \gg x_8$
e_5	$x_3 > x_5 \gg x_8 \geq x_4 > x_1 > x_{10} \geq x_9 \approx x_7 \geq x_6 > x_2$
e_6	$x_4 \approx x_{10} \gg x_5 \gg x_1 \geq x_6 \gg x_3 \geq x_7 > x_9 > x_2 > x_8$

Table 8. Completed fuzzy rankings of alternatives (first case)

The evolution of the dominance degree of the first two alternatives is shown in Figure 11 where the x-axis represents the number of iterations and the y-axis represents the dominance degree of the plotted alternative for each expert at a given iteration. Different colors correspond to different experts, the identifiers for experts and alternatives are shown on the right. Equations (13)-(15) are used to aggregate the FPRs coming from the two subgroups of experts and the resulting dominance degrees, associated to each alternative, are: $\phi(x_1) = 0.48$, $\phi(x_2) = 0.39$, $\phi(x_3) = 0.51$, $\phi(x_4) = 0.43$, $\phi(x_5) = 0.49$,

$\phi(x_6) = 0.54,\ \phi(x_7) = 0.36,\ \phi(x_8) = 0.34,\ \phi(x_9) = 0.28,\ \phi(x_1) = 0.6$. Again, the final group solution is x_{10}, although the new collective fuzzy ranking of alternatives is:

$$x_{10} \gg x_6 > x_3 \geq x_5 \approx x_1 > x_4 > x_2 > x_7 \geq x_8 \gg x_9.$$

P'										P''									
0,50	0,58	0,56	0,58	0,51	0,41	0,58	0,58	0,72	0,29	0,50	0,52	0,25	0,29	0,49	0,41	0,32	0,63	0,55	0,44
0,42	0,50	0,48	0,45	0,43	0,36	0,46	0,46	0,67	0,24	0,43	0,50	0,15	0,15	0,41	0,38	0,23	0,60	0,52	0,29
0,44	0,52	0,50	0,55	0,45	0,33	0,57	0,57	0,63	0,21	0,56	0,69	0,50	0,38	0,56	0,59	0,41	0,66	0,55	0,56
0,36	0,48	0,39	0,50	0,31	0,28	0,44	0,44	0,52	0,13	0,46	0,70	0,30	0,50	0,53	0,60	0,37	0,58	0,51	0,51
0,49	0,57	0,55	0,63	0,50	0,35	0,65	0,65	0,65	0,23	0,46	0,59	0,29	0,32	0,50	0,47	0,33	0,63	0,52	0,49
0,58	0,63	0,66	0,59	0,63	0,50	0,63	0,63	0,80	0,37	0,54	0,62	0,25	0,25	0,53	0,50	0,33	0,71	0,64	0,40
0,40	0,53	0,42	0,48	0,34	0,32	0,50	0,48	0,54	0,16	0,30	0,43	0,14	0,17	0,32	0,33	0,50	0,41	0,34	0,30
0,40	0,53	0,42	0,48	0,34	0,32	0,48	0,50	0,54	0,16	0,30	0,35	0,15	0,17	0,32	0,24	0,20	0,50	0,37	0,33
0,23	0,28	0,32	0,28	0,30	0,13	0,32	0,32	0,50	0,02	0,33	0,39	0,16	0,19	0,39	0,27	0,23	0,51	0,50	0,33
0,70	0,75	0,78	0,73	0,76	0,61	0,79	0,79	0,91	0,50	0,42	0,66	0,24	0,32	0,46	0,55	0,31	0,52	0,45	0,50

Table 9. The influenced FPRs P' and P'' obtained within the first and the second experts' subgroups (second case)

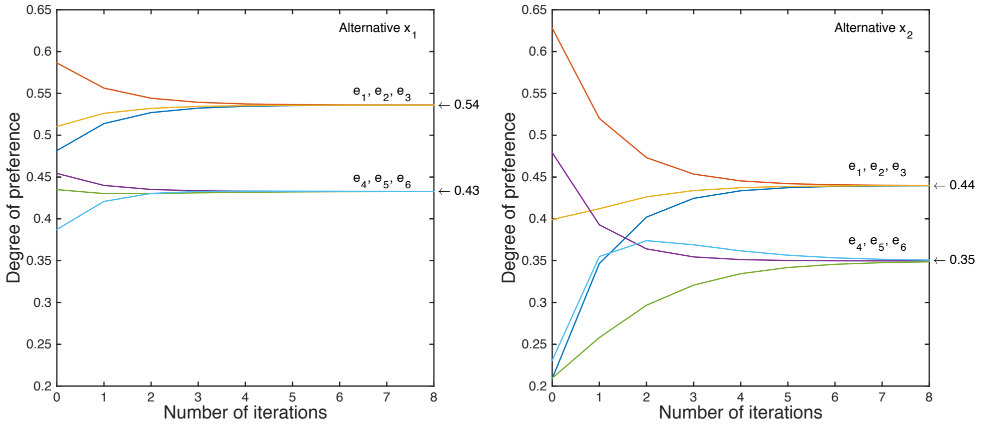

Figure 11. Evolution of experts' preferences for the alternatives x_1 and x_2 (second case)

3.7 Comparison with Related Works

As explained in section 3.1, the study of the effects of social influence in GDM has just begun and some early models dealing with influence are starting to be proposed. In [58], equations (43)-(44) have been applied for the first time in a GDM process to let the experts' individual opinions evolve, according to a predefined SIN, before being aggregated to form the collective FPR. Even though it is based on modified versions of the same equations, in our model the SIN is not predefined but generated from trust statements expressed by the experts in the same form of preferences about alternatives.

Interpersonal trust has been already used to improve the outcomes of a GDM process. In [61, 62], two models have been defined were each expert is explicitly asked to express their fuzzy trust statements on the other experts. Such statements are then aggregated and a global level of trust is calculated, associated to each expert and used to weight their opinions in the aggregation step. Instead, we propose to use trust statements to let the opinions of each expert evolve by incorporating elements captured from the opinion expressed by other experts she trusts.

In [59], the social influence among experts is calculated by combining the number of common connections with the number of direct interactions over a social network. The obtained value is then used to infer missing FPR values by selecting values from the opinions of influencing experts. Despite this method automates the influence estimation process, it does not guarantee that the tie strength over a social network is a good approximation of how an opinion can be influenced with respect to a DM problem. Moreover it requires that all experts are active members of the same social network.

It should be noted that, the use of data coming from social networks to support the DM process is not new. In [63] *Social Network Analysis* (SNA) is used to measure inter-organizational relationships to enhance a DM process for project selection while in [64] a consensus model based on SNA has been defined to reconcile conflicts in the collaborative annotation of media content.

According to [59], also in our model incomplete opinions are completed with data injected from trusted experts. In addition with respect to the same work, such opinions are further modified by simulating their evolution due to social influence. Moreover, unlike in [59, 61, 62], in our model the influence in not used to estimate a global importance level for each expert but to let the preference of each expert gradually evolve simulating interaction.

Our model uses for the first time fuzzy rankings to represent experts' opinions regarding both their preferences on the set of alternatives and their trust on other experts. Such preference model offers an higher degree of user friendliness and is less vulnerable to inconstancy than commonly used FPRs. Moreover, by asking experts to place themselves in the defined rankings, we avoid the complication of requiring the definition of a numerical value that represents the susceptibility level of each expert to influence (like in [58]) or a the interpersonal trust level as in [61, 62].

Simulating the natural evolution of opinions thanks to discussion, our model also tries to obtain the convergence between the experts' opinions. This is a distinctive feature with respect to existing models because social influence also impacts the preferences aggregation phase. In such sense, our model can be also used to support automated consensus processes. Table 10 summarizes the differences and the advantages of the proposed model with respect to other existing ones.

We believe that the defined model leads to a more accurate representation of the GDM process by formalizing important aspects that are commonly disregarded by other models. On the other hand, we estimate the level of social influence only based on interpersonal trust, without considering other psychological traits like leadership, charisma, persuasive ability, etc. that could strengthen or weaken influence when real interactions between experts take place. Nevertheless, we believe that the exclusion of these additional traits is advantageous and enables to reach more objective decisions.

The time complexity of the whole process embedded in the defined model is polynomial and limited by $\mathcal{O}(m \cdot n^3\, log\, n)$ where m is the number of experts and n is the number of alternatives.

	Our model	Model defined in [58]	Model defined in [59]	Models in [61, 62]
Estimation of Social Influence	Fuzzy rankings of experts	Predefined SIN	SNA	Numerical trust statements
Representation of Social Influence	SIN	SIN	Normalized tie strength	Normalized trust level
Applications of Social Influence	Estimation of missing preferences Evolution of preferences Selection of the best option	Evolution of preferences	Estimation of experts' importance Estimation of missing preferences	Estimation of experts' importance

Table 10. Comparison with other models

Part 2

Applications

Chapter 4

Applications to e-Learning

This chapter, that opens the second part of this thesis work, is aimed at the application of the GDM models and techniques, defined in the first part, to support peer assessment both in standard and massive educational contexts. *Massive Open Online Courses* (MOOCs) are becoming increasingly popular in education but, to reach their full extent, they require the resolution of new issues like assessing students at a massive scale. A feasible approach to tackle this issue is peer assessment, in which students also play the role of assessor for assignments submitted by others. Unfortunately, students are unreliable graders so peer assessment often does not deliver accurate results.

In this chapter, after having introduced the problem of student evaluation in massive contexts, peer assessment is described and formalized. Existing approaches, aimed at mitigating the problem of peer assessment reliability, are outlined and performance measures capable of establishing and comparing the goodness of different approaches are defined. Then, two novel approaches, aimed at improving peer assessment performance, are presented: the first one is based on graph mining techniques while the second one applies fuzzy GDM models and techniques defined in the first part.

4.1 Student Assessment in Massive Courses

The term MOOC was coined in 2008 to describe educational resources that show the following characteristics: *Massive* (there is no limit on attendance), *Open* (free of charge and accessible to anyone), *Online* (delivered via the Internet) and *Courses* (structured around a set of goals in a specific area of study) [65]. Since their introduction MOOCs have become a popular trend in online learning. According to [66], until the end of 2016, a total of 6.850 MOOCs have been launched from over 700 universities with the total number of students who signed up for at least one course estimated to be 58 million. Figure 12 shows the growth of MOOCs over years.

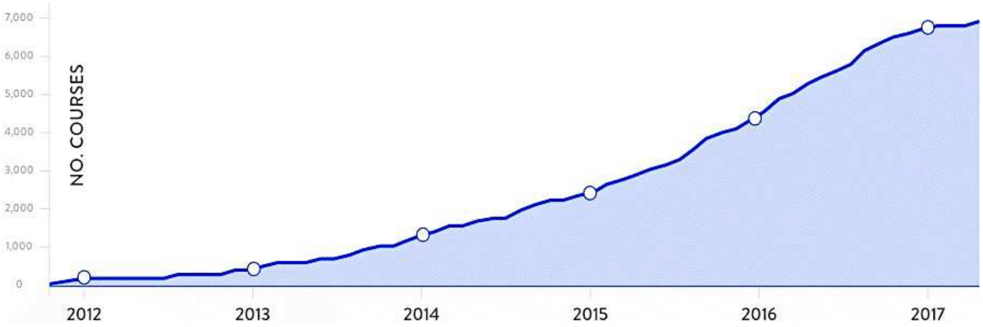

Figure 12. Growth of MOOC courses (source: Class Central)

According to [67], MOOCs are *a continuation of the trend in innovation, experimentation and use of technology initiated by distance and on-line learning, to provide learning opportunities for large numbers of learners.* Most of the discussions about MOOCs distinguish between two formats with two distinct pedagogical underpinnings [68]: *cMOOCs*, that are based on connectivism, emphasizes interaction with a distributed network of peers, learning artifacts, and learning technologies while *xMOOCs*, that are more structured and centralized, emphasize individual learning through video lectures and regular assessments.

Due to their scale, MOOCs introduce new technical and pedagogical challenges that require overcoming the traditional e-learning model based on tutor assistance to maintain a cheap and unrestricted access to high quality resources. Because of the high number of students enrolled and the relatively small number of tutors, in fact, tutor involvement during delivery stages has to be limited to the most critical tasks [69].

In [70], the key challenges that MOOCs designers and providers are facing are analysed. Massiveness and low teaching involvement have been identified as one of the biggest challenges. Moreover, since the heterogeneity of MOOC learners is high, and their level of maturity and experience is varied, courses have to be conceived taking into account different educational and cultural backgrounds. Another concern is a high students' dropout rate, with several sources indicating that only 10% of participants finish the courses on average. However, some authors suggest that such statistics might be interpreted in the light on the different personal goals that motivate students' attendance to a course besides finalizing it.

According to the same work, among the key challenges of MOOCs, the assessment of students' performance is one of the most prominent. In fact, given their discrepancy in number, it is not possible for the tutors to follow up with every student and review assignments individually. This also represents a major obstacle to the credential programs launched by MOOC players and targeted to people that want to achieve credits toward a degree or earn credentials to show to prospective employers.

A typical approach to overcome the assessment problem is to use close questions in exams and assignments so that grading can be automated [71]. Unfortunately, automated grading is limited, disappointing and insufficient, with no partial marks and, in some cases, with no detailed explanations of answers. It may result particularly unsatisfactory when applied to complex tasks like the evaluation of the students' ability of proving mathematical statements, expressing their critical thinking over an issue, demonstrating proficiency in skills like creative writing, etc. [72].

To overcome these limitations, an approach that is gaining a growing consensus is *Peer Assessment* that can support both the *formative assessment* task (aimed at monitoring student learning and providing ongoing feedback) and the *summative assessment* one (aimed at evaluating student learning at the end of the course). In peer assessment, students are required to grade a small number of their peers' assignments as part of their own assignment.

The final grade of each student is then obtained by combining information provided by peers.

The positive aspect of this approach is its capability of easily scale to any size: the number of assessors in fact naturally grows with the number of students. Conversely, its use may be seen as unprofessional and unreliable given that it is based on grades assigned by students lacking the needed expertise, both didactical and on the specific subject to be assessed. Some researches point out that students themselves seem to distrust the results of peer assessment [73]. To mitigate this issue, several corrected methods have been identified as described in the next section.

4.2 Peer Assessment Methods

Peer assessment has been used for many years as a tool to improve learning outcomes. In fact, the literature reports on many learning benefits for peer-assessors like the exposure to different approaches, the development of self-learning abilities, the enhancement of critical thinking, etc. [69]. Even if some studies suggest a good correlation between the results of peer assessment and instructor ratings in conventional classrooms and online courses (at least for specific, high structured domains), there is still a general concern on its use as a reliable strategy to approximate instructor marking [73].

Despite these concerns, given the growing diffusion of MOOCs and the related increasingly felt issue of students' assessment, the application of peer assessment as an evaluation tool is increasing. To improve its accuracy, several approaches, at various stages of development, have been proposed so far as summarized below.

The *Calibrated Peer Review* (CPR) proposes a calibration step to be performed by students before starting to assess other students' assignments [74]. During the calibration step, each student rates a set of assignments that have been already rated by the instructor. The discrepancy between students' and instructors' grades measures the accuracy of each student and is used to weight subsequent assessments provided by the same student. Obviously, the more accurate is an assessor, the more weight is given to her judgment on a peer assessment.

CPR has been experimented in several contexts demonstrating to be an effective instructional tool. Despite that, it requires additional work from those students who are asked to take part in the calibration step. Moreover, this method does not take into account the progresses that students make over time until a new calibration step is done. For this reason, additional approaches have been defined able to automatically tune peer grades based on different parameters.

In [75], three probabilistic models for tuning peer-provided grades are presented. Such models estimate the reliability of each assessor as well as her *bias* (i.e., a score reflecting the assessor's tendency to inflate or deflate her grade) based on the analysis of grading performance on special "ground truth" submissions that are evaluated either by the instructor or by a big number of peers (hypothesising that the mean of many grades should tend toward the correct grade). Reliability and bias of each student are then used to tune the provided grades to other submissions.

A similar approach has been applied in [76], where a *Bayesian* model has been used to calculate the bias of each peer assessor in general, on each item of an assessment rubric and as a function of the assessor grade assigned by the instructor. As in the previous case, obtained biases are used to tune the grades provided during peer assessment. Differently from the previous case, bias calculation is based on the results of a whole round of assessment rather than on just few "ground truth" submissions so, in the calibration step, the instructor should rate all the submissions. In [77] comparable results have been obtained with a hierarchical Bayesian model.

The *Vancouver* algorithm, defined in [78], measures the grading accuracy of a student by comparing the grades given by her to each assignment with the average grade for that assignment. Differently from the other approaches, the assessor accuracy is used as a modifier of the assessor's grade rather than of assessees' ones so that the student's grade can reflect not only the quality of her homework but also the quality of her work as a reviewer.

In [79] the ability of an assessor student to correctly rate peer students is assumed to be dependent on the grade obtained by the same student. In other words, final grades to be assigned to students are obtained by weighting the grades proposed by their assessors on the basis of the grades received by the assessors themselves. Given that students' grades recursively depend on other students' grades, an iterative algorithm, named *PeerRank* (inspired by *Google PageRank* [80]) is proposed for their calculation. The advantage of this approach, compared to the previous ones, is that it does not require any instructor's intervention given that there is no need of a ground truth of professionally graded assignments.

In [81] a different approach, aimed at making the assessment process as simple as possible, has been proposed. The authors have shown that *ordinal* feedback (e.g. "the report x is better than the report y") is easier to provide and more reliable than *cardinal* one (e.g. "the grade of report x is a B"). Basing on that assumption, the authors have defined a probabilistic model for obtaining student grades starting from partial rankings provided by the peers. An experiment with real data have demonstrated that the performance of such method is at least competitive with cardinal methods for grade estimation, even though it requires less information from the graders.

In [72], the authors have shown that *Ordinal Peer Assessment* is a highly effective and scalable solution for student evaluation. They have defined a model for distributing the assignments among peers so that the collected individual rankings can be merged into a global one that is as close as possible to the real ranking. They have demonstrated that, given k students, if each correctly ranks the received assignments, the defined aggregation method is able to recover a fraction $1 - \mathcal{O}(1/k)$ of the true ranking. They have also demonstrated that the same ordinal peer assessment method is quite robust even when students have imperfect capabilities as graders.

With respect to the application of *Fuzzy Set Theory* to peer assessment, some experiment has been already performed so far. In [82], the students of a class have been asked to express a grade, in terms of a fuzzy value in [0,1], for each assignment coming from the other students in the same class. The final grade of each assignment is then obtained by averaging the proposed grades, weighted with respect to expertise levels assigned by the teacher.

In [83], the authors have proposed a framework aimed at enhancing the effectiveness of peer assessment by letting students express peer grades as fuzzy membership functions with respect to a given set of assessment criteria. The proposed grades are then adapted basing on assessors' learning styles (through defined heuristics) and differences among grades are reconciled through agent negotiation based on fuzzy constraints.

In [84], the students of a class have been experimentally asked to evaluate the assignments coming from peers in terms of linguistic labels mapped to interval *Type-2* fuzzy sets. Then, the final grade of each assignment has been obtained by aggregating the grades proposed by peers and weighting them with respect to the expertise levels assigned by the teachers. Obtained results have been re-mapped on linguistic labels to obtain the final literal grades.

Basing on the reported literature, ordinal peer assessment methods have shown a more promising behavior with respect to cardinal ones. In particular, they overcome the problem that students may be grading on different scales in fact, by letting students propose ordinal statements rather than cardinal grades, there is no need to develop a scale from each student onto the peer assessment algorithm. On the other hand, the existing fuzzy-based methods seems to be mainly thought for small contexts and aimed at encouraging class students to participate in the evaluation of their learning, so enhancing their reflective and critical thinking, rather than at providing reliable grades for students in massive learning contexts.

4.3 Formalization of the Peer Assessment Problem

In a typical peer assessment scenario an *assignment* is given to n different students $S = \{s_1, \dots, s_n\}$. Each student elaborates her own solution (e.g. an essay, a set of answers to open-ended questions, etc.) generating a *submission*. Each student has then to grade m different submissions (with $m \leq n$) coming from other students (maybe based on an assessment rubric).

The assignment of submissions to assessor students is performed in accordance to an *assessment grid*: a Boolean $n \times n$ matrix $A = (a_{ij})$ where $a_{ij} = 1$ if the student s_j has to grade the submission of s_i while $a_{ij} = 0$ otherwise. The matrix A has the following properties:

- the sum of the elements in each row and column is equal to m (i.e. each student grades and is graded by m other students);
- the sum of the elements in the main diagonal is equal to 0 (i.e. nobody evaluates himself).

A feasible way to build an assessment grid is by filling it at random with an algorithm preserving the above properties. A possible (non optimized) algorithm starts with an $n \times n$ null matrix and initializes its elements basing on the following equation:

$$a_{\mathrm{mod}(i+j-1,n)+1,i} = 1 \ \forall i \in \{1, \dots, n\}, j \in \{1, \dots, m\} \tag{51}$$

where *mod* indicates the remainder after division of the first term by the second one. The obtained matrix is then shuffled in several iterations by randomly selecting a couple of rows (or columns) $i, j \in \{1, \dots, n\}$ such that $a_{ij} = a_{ji} = 0$ and swapping them.

Then, in a *Cardinal Peer Assessment* setting, each student $s_j \in S$ has to review and propose a grade for every peers' submissions according to the assessment grid i.e. to all students in $S_j = \{s_i \in S | \ a_{ij} = 1\}$. Proposed grades are collected in a $n \times n$ *grades matrix* $G = (g_{ij})$ whose generic element g_{ij}, so that $0 \leq g_{ij} \leq 10$, is the grade proposed by s_j for s_i. The *final grade* g_i for each student $s_i \in S$ can be then obtained starting from G according to the adopted method among those discussed in section 4.2.

In an ideal peer grading setting, every student performs the grading task so, the easiest way to estimate the *final grade* g_i of any student $s_i \in S$ is by averaging all the grades obtained by peers (a matrix row) as follows:

$$g_i = \frac{1}{m}\sum_{j=1}^{n} g_{ij} \ \ \forall\ 1 \leq i \leq n. \tag{52}$$

The same equation can be applied to non-ideal settings (when some students skip the grading task) by averaging on the total number $m_i' < m$ of grades proposed for *i.* Some authors propose to average all obtained grades apart the best and the worst, while other authors use the median in place of the average [78].

The assessment grid can be seen as the adjacency matrix of an m-regular directed graph where each node represents a student and each arc represents an assessment to be performed. In addition, the grades matrix can be seen as the weighted adjacency matrix of an m-regular directed graph where each node represents a student, each arc represents an assessment and the weight on arcs represent assigned grades. Figure 13 shows the graph interpretation of a grades matrix with 6 students and 2 submissions to be rated by each (i.e. so that $n = 6$ and $m = 2$).

Differently from the previous case, in a *Ordinal Peer Assessment* setting, each student s_j is asked to define an ordinal ranking $\succ_j$ (see section 2.1) over the subset of her assessee $S_j = \{s_1^j, \dots, s_m^j\}$ as follows:

$$s_{\sigma(1)}^j \succ_j s_{\sigma(2)}^j \succ_j \dots \succ_j s_{\sigma(m)}^j \tag{53}$$

where $\sigma\colon \{1, \dots, m\} \to \{1, \dots, m\}$ is a permutation function. Equation (53) means that, according to s_j, the submission of the student $s_{\sigma(1)}^j$ is better than that of $s_{\sigma(2)}^j$, etc. According to the notation introduced in section 2.1, the same ranking can be represented as an ordering array $O_j = \left(o_1^j, \dots, o_n^j\right)$ where $o_i^j \in \{1, \dots, m\}$ represents the position, within the ranking, of the submission coming from the student $s_i^j \in S_j$.

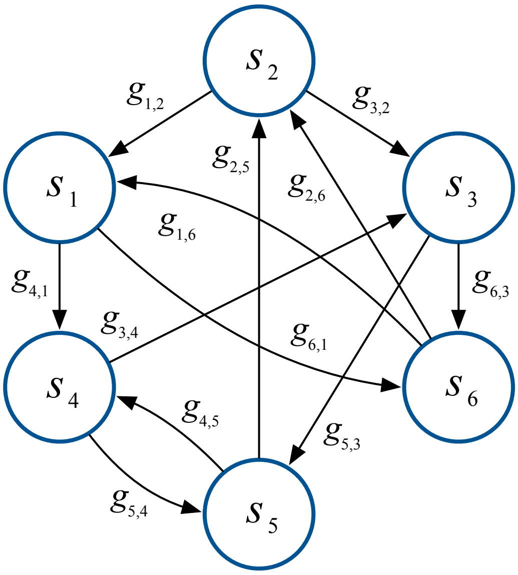

Figure 13. Graph interpretation of peer assessment

The ranking $\succ_j$ is undefined for elements not included in S_j so it is a *partial ranking* over S. The partial rankings defined by all students are so collected in a $n \times n$ *ranking matrix* $R = (r_{ij})$ whose generic element r_{ij} is the position of s_i in the ranking $\succ_j$ if $s_i \in S_j$ (i.e. the element o_i^j from the ordering array O_j), 0 otherwise. Starting from a ranking matrix, an *aggregation rule* is able to compute a complete ranking over the whole set of submissions.

Several aggregation rules have been defined so far, according to the methods discussed in section 4.2. A simple and effective aggregation rule is the classical *Borda count* [85] where the partial ranking provided by each assessor is interpreted as follows: m points are given to the submission ranked first, m—1 points to the one ranked second, etc. The Borda score of the submission coming from s_i is then calculated as follows:

$$Borda(s_i) = \sum_{j=1}^{n} a_{ij} \cdot (m - r_{ij} + 1). \tag{54}$$

The global ranking is then computed by ordering all the submissions in decreasing order of their Borda scores.

In [72], authors have demonstrated that Borda outperforms other, more complex aggregation rules like *Random Serial Dictatorship* [86] and *Markov chain* inspired methods [87] especially in case of imperfect grading (i.e. when partial rankings defined by students are not consistent to the ground truth). In [81] authors have defined other methods for ordinal peer assessment based on models that represent probabilistic distributions over rankings, obtained from the models of *Mallows* [88], *Bradley-Terry* [89] and *Plackett-Luce* [90]. Such methods have demonstrated better performance with respect to Borda also in case of imperfect grading and are also capable of detecting meaningful cardinal grades.

4.4 Measuring Peer Assessment Performance

The *Root Mean Square Error* (RMSE) is the most widely used performance indicator in peer assessment. Let g_i be the final grade estimated for a student s_i through peer assessment and $\overline{g_i}$ the ground truth i.e. a grade assigned to the same student by an experienced teacher for $i \in \{1, \dots, n\}$, the RMSE between estimated and real grades is calculated as follows:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} (g_i - \overline{g_i})^2}{n}}. \tag{55}$$

Statistically, the RMSE represents the sample standard deviation of the differences between predicted and observed values. The individual differences are called *residuals* when the calculation is performed over the data sample used for estimation, and *prediction errors* when it is performed out-of-sample. The RMSE allows to aggregate the magnitudes of the errors in predictions for various times into a single measure of predictive power. It is a measure of accuracy and is used to compare forecasting errors of different models for a particular task [91].

The effect of each error on RMSE is proportional to the size of the squared error thus larger errors have a disproportionately large effect on RMSE. As a consequence, RMSE is sensitive to outliers. For this reason some researcher recommends the use of alternative error measures like the *Mean Absolute Error* (MAE) where the influence of each error is proportional to the absolute value of the error [92]. The MAE is defined as follows:

$$MAE = \frac{\sum_{i=1}^{n}|g_i - \overline{g_\imath}|}{n} \tag{56}$$

where the symbols have the same meaning that in equation (55).

RMSE and MAE both summarize performance in ways that disregard the direction of over- or under- prediction. Both measures are scale-dependent, therefore they cannot be used to make comparisons between models that operate on different scales. MAE has advantages in terms of interpretability over RMSE but it is less widespread with respect to the evaluation of models for peer assessment.

If we refer to *ordinal peer assessment*, performance can be measured in terms of similarity between the ranking O estimated through peer assessment on the set S and the ground truth $\bar{O}$ i.e. a ranking defined by an experienced teacher on the same set. The *Kendall's* rank correlation coefficient $\tau\big(O,\bar{O}\big)$ or the *Spearman's* rank correlation coefficient $\rho\big(O,\bar{O}\big)$ defined in section 2.1 can be used for this purpose. A similar measure is the *Percentage of Correctly Recovered Pairwise Relations* (PCRPR) with respect to the ground truth [72] that can be calculated as follows:

$$PCRPR\big(O,\bar{O}\big) = \frac{2c}{n(n-1)} \tag{57}$$

where c is the number of concordant pairs between O and $\bar{O}$ i.e. the number of pair of elements of S which have the same order in the two rankings.

While τ and ρ are normalized in [−1,1] (where 1 means identity, 0 means lack of correlation and −1 means reverse correlation), PCRPR is normalized in [0,1] (where 1 means identity, 0 means absence of association). It should be noted that while RMSE and MAE are error measures (so smaller values correspond to better models); τ, ρ and PCRPR are similarity measures (so higher values correspond to better models).

4.5 Peer Assessment Methods based on Graph Mining

In section 4.3, a graph interpretation of peer assessment is proposed where the grades matrix G is seen as the weighted adjacency matrix of an m-regular directed graph with nodes representing involved students and weighted arcs representing performed assessments. Basing on such interpretation, *Graph Mining Peer Assessment* (GMPA) methods estimate the final grade of each student with techniques based on graph theory.

In [79] it has been proposed to weight the grade that each assessor student gives to another student by her own grade i.e. to use the grade of a student as a measure of her ability to grade correctly. Let $G = (g_{ij})$ be the grades matrix obtained by cardinal peer assessment on a set of submissions coming from students in $S = \{s_1, \dots, s_n\}$ according to an assessment grid $A = (a_{ij})$, the estimated grade g_i of a $s_i \in S$ can be so obtained as:

$$g_i = \frac{\sum_{j \to i} g_{ij} \cdot g_j}{\sum_{j \to i} g_j} \tag{58}$$

where both summations (at numerator and at denominator) are calculated over all students s_j that have evaluated s_i (indicated with $j \to i$) i.e. such that $a_{ij} = 1$.

Given that the grades of all assessor students are themselves weighted averages of grades obtained by their own assessors, an iterative process, named *PeerRank*, has been proposed to calculate the final grade of each student. Let be $g_i^{(t)}$ the grade of the student i at the t-th iteration, the grade of i at the iteration $t+1$ is defined as:

$$g_i^{(t+1)} = (1-\alpha)g_i^{(t)} + \alpha \frac{\sum_{j \to i} g_{ij} \cdot g_j^{(t)}}{\sum_{j \to i} g_j^{(t)}} \tag{59}$$

where $0 \leq \alpha \leq 10$ is a constant affecting the convergence speed and g_i^0 is initialised by simply averaging all the grades obtained by peers according to equation (52).

Equation (59) takes into account that each student only evaluates m peers according to the *assessment grid*. This is a more realistic setting with respect to the one described in [79] where each student is assumed to evaluate any other student. In the same paper, useful properties for the defined grade updating rule have been defined and it has been also demonstrated that, after a limited number of iterations, the rule converges to stable values.

It is interesting to note that equation (59) is a variation of the *Google PageRank* rule proposed in [80]. While, in PageRank, Web pages are ranked according to the ranks of the Web pages that link to them, in PeerRank, a grade assigned by a student is weighted on the grade assigned to her by other students. So, equation (59) can be seen as an indicator of the centrality [93] of each node of the graph obtained from the grades matrix G. According to this interpretation we classify PeerRank under the GMPA umbrella as well as the derived methods described below.

Equation (59) does not incentivize students to evaluate peers accurately. For this reason, in [79], the following update to the PeerRank rule has been proposed:

$$g_i^{(t+1)} = (1-\alpha-\beta)g_i^{(t)} + \alpha\frac{\sum_{j\to i} g_{ij}\cdot g_j^{(t)}}{\sum_{j\to i} g_j^t} + \beta\frac{\sum_{j\to i} 10 - |g_{ji} - g_j^t|}{m} \tag{60}$$

where $0 \leq \beta \leq 1$ is a constant, so that $\alpha + \beta \leq 1$, that weights the reward given to a student according to the inverse normalised absolute error in the grades provided by her.

If $\beta = 0$ then equation (60) degenerates to equation (59). For $\beta > 0$, if $g_{ji} = g_j^{(t)}$ for all $j \in \{1,\dots,n\}$ so that $a_{ji} = 1$, then all the grades assigned by s_i are accurate and the contribution of the third addendum is $10\cdot\beta$. At the opposite, if $\left|g_{ji} - g_j^{(t)}\right| = 10$ for all j so that $a_{ji} = 1$, then the grades assigned by s_i are wrong and the contribution of the third addendum is 0.

The updated PeerRank rule, described by equation (60), prescribes that the influence of the grade of an assessor student on any grade she proposes is linear. For sake of simplicity we can decompose equation (60) as the sum of tree different components as follows:

$$g_i^{(t+1)} = (1-\alpha-\beta)g_i^{(t)} + \alpha\gamma_i^{(t)} + \beta\delta_i^{(t)} \tag{61}$$

where the constants α and β have the same meaning as in equation (60), $\gamma_i^{(t)}$ is the contribution coming from peer graders while $\delta_i^{(t)}$ is the incentive for accurate grading.

In order to improve the quality of the final grades, we propose an updated rule named *F-PeerRank* that applies a super-linear modifier to the grades proposed by peer assessors by modifying the $\gamma_i^{(t)}$ component as follows:

$$\gamma_i^{(t)} = \frac{\sum_{j\to i} g_{ij}\cdot f\left(g_j^{(t)}\right)}{\sum_{j\to i} f\left(g_j^{(t)}\right)} \tag{62}$$

The function f, that affects the contribution given by the grades proposed by other peers, has the purpose of minimizing the contribution of low skilled student while maximising those of high skilled ones. Feasible functions are the power function $f(x) = x^n$ (for some $n > 1$) as well as the exponential one $f(x) = e^x$ (with e being the Euler's constant).

Bringing this reasoning to the extreme, we can imagine to assign the maximum influence only to the best grader for each student and no influence at all to any other proposed grade. This is the case of another approach we propose, named *BestPeer*. It calculates a transitory grade g_i' for any student s_i with one of the previous methods and then assigns to each student the final grade g_i according to the following rule:

$$g_i = g_{i,\underset{j \to i}{\operatorname{argmax}} g_j'} \tag{63}$$

where the function *argmax* (argument of the maximum) returns the value j so that g_j' is maximized for $j \in \{1, \dots, n\}$ and $a_{ij} = 1$.

This method is capable of performing particularly well when, for each student, at least one good grader is available. Unfortunately, this condition cannot be granted with the random assessor-assessee assignment proposed by equation (51) that can generate settings in which some student is assessed by only unreliable graders (i.e. students with a low grade). In this case, even weighting the grades, the overall peer-assessment performance may be poor.

Balancing reliable graders among students is a feasible approach to overcome this issue but, unfortunately, we have no information about the grades when the assessment grid is built. To overcome this issue it is possible to initialize the assessment grid $A = (a_{ij})$ based on grades coming from previous assessments. To do that, a feasible algorithm starts with a null matrix and initialises its elements according to the following equation:

$$a_{\text{mod}(m(i-1)+j-1),n)+1,rank(i)} = 1 \tag{64}$$

for each $1 \leq j \leq n$ and $1 \leq i \leq m$ and where $rank(i)$ denotes the position of the i-th student in the list of the students ordered decreasingly on the average grade obtained in previous assessments.

Equation (64) does not ensure the fulfilment of the second property of assessment grids. For this reason another check is needed and, if $a_{ii} = 1$ for some $i \in \{1, \dots, n\}$, then the closest column j of A so that $a_{ij} = 0$ and exists a $z \in \{1, \dots, n\}$ so that $a_{zi} = 0$ and $a_{ij} = 1$ is selected and the values of a_{ii} and a_{ij} are swapped as well as values of a_{zi} and a_{zj}. In other words, the student s_i does not assess himself anymore but the student s_z assigned to the closest performer s_j that, in turn, takes care of evaluating s_i.

A second option for optimizing the assessor-assessee assignment is to proceed incrementally (i.e., to perform the assessment session in m rounds). In the first round, just one student to grade is assigned to each other student. In each subsequent round, students are ranked in two lists: *list 1* orders students, decreasingly, on the average grade obtained in the preceding rounds (i.e. on their ability as graders); *list 2* orders students, increasingly, on the average grade obtained by their graders in the preceding rounds (i.e. on the quality of obtained grades).

Then, for the subsequent round, each student from list 1 has to grade the student from the list 2 with the same rank. This ensures that, in each step, the best graders are assigned to the students that, in the previous steps, have obtained grades from the worst ones. Some additional checks must be made to ensure that no student evaluates herself and that no student evaluates another student more than once.

This method has the advantage that it does not need any information about past assessments. Conversely, its incremental nature requires that every grade is assigned for a given round before starting the next one. This constraint can be very expensive, especially in massive contexts, when some student may be late in providing grades or may not provide grades at all.

4.6 Fuzzy Ordinal Peer Assessment

A peer assessment problem, as formalized in section 4.3, can be seen as a special case of GDM problem. In a typical GDM problem, a group of experts evaluate a set of alternatives, taking into account the involved factors and criteria, with the aim of selecting the best one to adopt. To this end, each expert expresses her preferences on alternatives, preferences are aggregated, a collective preference degree of each alternative is calculated and a ranking over alternatives is generated.

Similarly, in peer assessment, the involved students evaluate submissions made by other students (rather than alternatives) and their evaluations are aggregated to obtain the grade of each submission (rather than the degree of preference of each alternative). For these reasons, a peer assessment problem can be regarded as a GDM problem where:

- experts and alternatives belong to the same set (i.e. students evaluate the submissions made by other students);
- each expert only ranks a small subset of alternatives (i.e. few submissions are evaluated by each student);
- experts' opinion is not fully reliable (it should be taken into account that students are far to be perfect assessors).

These properties (in particular the last two) suggest to refer to GDM approaches able to deal with the uncertainty resulting from inaccuracy and lack of knowledge in experts' evaluations, like those based on the fuzzy set theory. Following these considerations, this section introduces a new peer assessment model, named *Fuzzy Ordinal Peer Assessment* (FOPA) based on the GDM models and techniques defined in the first part of this thesis.

As described in section 4.3, in ordinal peer assessment, each student of a set $S = \{s_1, \dots, s_n\}$ ranks the submissions coming from m other students according to an assessment grid $A = (a_{ij})$. By setting $E = X = S$ (where E and X are, respectively, the sets of experts and of alternatives of a standard GDM problem as seen in section 1) and assigning to each student $s_k \in S$ a subset $S_k = \{s_i \in S | \; a_{ik} = 1\}$ of submissions to be evaluated, we easily obtain the GDM problem corresponding to peer assessment.

In ordinal peer assessment each student $s_k \in S$ is asked to define a partial ranking on S_k. By leveraging on GDM, preferences between the elements of

S_k can be expressed in terms of a FPR P_k. Individual FPRs, coming from the assessor students, can then be aggregated and a global ranking of submissions can be calculated. Unfortunately, the definition of a FPR may result too complex and time-consuming for students with the risk of introducing errors and inconsistencies impacting assessment performances. To overcome this issue, FOPA adopts a simpler preference model based on fuzzy rankings (as defined in section 2).

In FOPA each student $s_k \in S$ proposes a fuzzy ranking R_k over S_k (that is a partial fuzzy ranking over S). Each R_k for $k \in \{1, \dots, n\}$ is then converted in a FPR P_k according to the methods introduced in section 2.3 and used for subsequent processing. The main advantage of this approach is that students not only order the submissions from the best to the worst but also express a degree of preference between them. As explained in the next section, this allows to obtain better performances when reconstructing the global ranking and, also, to obtain a reliable cardinal grade for each submission. Moreover, it mitigates the *bias* problem (seen in section 4.2) given that students provide relative evaluations that consider only a couple of submissions at a time.

Example 18. *Let $S = \{s_1, \dots, s_6\}$ be a set of students involved in a peer assessment session. Let us suppose that, according to a random assessment grid, the student s_1 has to evaluate the subset of students $S_1 = \{s_2, s_4, s_5, s_6\}$ and that she provides the following fuzzy ranking:*

$$R_1 = (s_4 \gg s_5 \approx s_2 > s_6).$$

The student states that, according to her opinion, the submission of s_4 is much better than that of s_5 and s_2 (considered at the same level) that, in turn, are better than that of s_6. Through equation (32) it is then possible to obtain the corresponding partial FPR as follows:

$$P_1 = \begin{pmatrix} - & - & - & - & - & - \\ - & 0.50 & - & - & 0.50 & 0.65 \\ - & - & - & - & - & - \\ - & - & - & 0.50 & 0.85 & - \\ - & 0.50 & - & 0.15 & 0.50 & - \\ - & 0.35 & - & - & - & 0.50 \end{pmatrix}$$

where the symbol – indicates an undefined cell. Applying equations (22)-(23) on P_1 we can obtain some of the missing values as follows:

$$P_1 = \begin{pmatrix} - & - & - & - & - & - \\ - & 0.50 & - & 0.15 & 0.50 & 0.65 \\ - & - & - & - & - & - \\ - & 0.85 & - & 0.50 & 0.85 & 1.00 \\ - & 0.50 & - & 0.15 & 0.50 & 0.65 \\ - & 0.35 & - & 0.00 & 0.35 & 0.50 \end{pmatrix}$$

Given n students and m assignments per student, for every defined fuzzy ranking R_k with $k \in \{1, \dots, n\}$, the conversion step produces an FPR P_k where only a fraction of m^2/n^2 elements are defined. In real contexts, hundreds of students (thousands in MOOCs) have to be evaluated in total (so n becomes very large) while each student can be requested to evaluate only a small number of other submissions (so m remains small). This means that every P_k becomes a sparse matrix with only few elements defined.

When all individual FPRs $P_k = (p_{ij}^k)$ with $k \in \{1, \dots, n\}$ are obtained, an *aggregation step* is needed to build the collective FPR $P = (p_{ij})$. To do that, FOPA adopts the OWA_Q operator defined in section 1.5 with the exception that individual FPRs are incomplete so undefined elements must be excluded. To this end, the following equation, that combines and adapts equations (13) and (15), is used to determine the collective FPR elements:

$$p_{ij} = \sum_{k \in K_{ij}} \left(\mu_Q \left(\frac{k}{\# K_{ij}} \right) - \mu_Q \left(\frac{k-1}{\# K_{ij}} \right) \right) p_{ij}^{\sigma_{ij}(k)} \qquad (65)$$

where $K_{ij} = \{k \mid p_{ij}^k \text{ is defined}\}$, $\sigma_{ij}\colon K_{ij} \to K_{ij}$ is a permutation function aimed at reordering the values of K_{ij} so that $p_{ij}^{\sigma_{ij}(k)} \geq p_{ij}^{\sigma_{ij}(k')}$ for any $k < k'$ with $k, k' \in K_{ij}$ and $\mu_Q\colon[0{,}1] \to [0{,}1]$ is the membership function of the selected linguistic quantifier.

After having aggregated individual preferences, it could happen that some values of the collective FPR P still remain undefined. In fact when none of the assessor students has expressed a preference between the i-th and j-th submissions for some $i, j \in \{1, \dots, n\}$, then the corresponding values p_{ij} and p_{ji} of the collective FPR can't be calculated. In most cases it does suffice to estimate missing values according to equations (22)-(23) or equation (28) as described in section 1.7.

For $n \gg m$ and when many students skip the assessment task for one or more submissions, some elements of P may still remain undefined. Such ignorance situation can be solved through seed-based approaches as described in section 1.7. For example it is possible to assume *indifference* for any undefined value by setting it to 0.5. Then, estimators defined by equations (22)-(23) or by equation (28) can be applied again to make seed values as consistent as possible to the other FPR values.

Example 19. *Let P_2 and P_3 be individual FPRs generated from the fuzzy ranking $R_2 = (s_1 \geq s_6 \approx s_5 \geq s_3)$ and $R_3 = (s_4 > s_1 \geq s_5 > s_6)$ as follows:*

$$P_2 = \begin{pmatrix} 0.50 & - & 0.65 & - & 0.58 & 0.58 \\ - & - & - & - & - & - \\ 0.35 & - & 0.50 & - & 0.43 & 0.43 \\ - & - & - & - & - & - \\ 0.43 & - & 0.58 & - & 0.50 & 0.50 \\ 0.43 & - & 0.58 & - & 0.50 & 0.50 \end{pmatrix},$$

$$P_3 = \begin{pmatrix} 0.50 & - & - & 0.35 & 0.58 & 0.73 \\ - & - & - & - & - & - \\ - & - & - & - & - & - \\ 0.65 & - & - & 0.50 & 0.73 & 0.88 \\ 0.43 & - & - & 0.28 & 0.50 & 0.65 \\ 0.28 & - & - & 0.13 & 0.35 & 0.50 \end{pmatrix};$$

the collective FPR obtained by aggregating them with P_1 (from Example 18) through equation (65) initialized with the increasing proportional linguistic quantifier most (see Figure 4), is shown below:

$$P = \begin{pmatrix} 0.50 & - & 0.65 & 0.35 & 0.58 & 0.64 \\ - & 0.50 & - & 0.15 & 0.50 & 0.65 \\ 0.35 & - & 0.50 & - & 0.43 & 0.43 \\ 0.65 & 0.85 & - & 0.50 & 0.78 & 0.93 \\ 0.43 & 0.50 & 0.58 & 0.20 & 0.50 & 0.61 \\ 0.34 & 0.35 & 0.58 & 0.05 & 0.36 & 0.50 \end{pmatrix}.$$

Then missing values are estimated on the collective FPR through equations (22)-(23) to complete it as follows:

$$P = \begin{pmatrix} 0.50 & 0.59 & 0.65 & 0.35 & 0.58 & 0.64 \\ 0.41 & 0.50 & 0.65 & 0.15 & 0.50 & 0.65 \\ 0.35 & 0.35 & 0.50 & 0.11 & 0.43 & 0.43 \\ 0.65 & 0.85 & 0.89 & 0.50 & 0.78 & 0.93 \\ 0.43 & 0.50 & 0.58 & 0.20 & 0.50 & 0.61 \\ 0.34 & 0.35 & 0.58 & 0.05 & 0.36 & 0.50 \end{pmatrix}$$

Once all values of the collective FPR have been defined, it is possible to calculate the degree of preference $\phi(s_i)$ for each $s_i \in S$ according to one of the measures defined in section 1.6 (i.e. NF, NDD, QGDD or QGNDD). The global ranking between the alternatives is then computed by ordering all the submission decreasingly on their preference degree. In alternative, one of the methods described in section 2.4 can be applied to directly obtain the global fuzzy ranking of all submissions from the collective FPR.

Starting from the preference degrees it is possible to calculate the *cardinal grade* of each submission, provided that a cardinal assessment is made by a reliable expert (e.g. the teacher) to the best and the worst submissions (i.e. the first and the last in the final ranking). Let g_{min} and g_{max} be the grades assigned to the best and the worst submissions, the estimated grade g_i for every $s_i \in S$ can be obtained via normalization as follows:

$$g_i = \frac{(\phi(s_i) - \phi_{min}) \cdot (g_{max} - g_{min})}{(\phi_{max} - \phi_{min})} + g_{min} \tag{66}$$

where ϕ_{min} and ϕ_{max} are the degrees of preference associated to the best and the worst submissions.

Example 20. *From the collective FPR P resulting from Example 20, it is possible to obtain the preference degree of each submission in terms of Net Flow through equation (18) as follows:* $\phi_{NF}(s_1) = 0.63$*;* $\phi_{NF}(s_2) = -0.28$*;* $\phi_{NF}(s_3) = -1.68$*;* $\phi_{NF}(s_4) = 3.23$*;* $\phi_{NF}(s_5) = -0.33$*;* $\phi_{NF}(s_6) = -1.58$. *The collective fuzzy ranking of submissions can be then obtained through equations (36)-(37) as follows:*

$$s_4 \gg s_1 > s_2 \approx s_5 > s_6 \approx s_3.$$

By applying equation (18) on obtained preference degrees with $g_{min} = 2$ *and* $g_{max} = 9$ *(supposed to be assigned by an expert assessor), the following grades can be estimated:* $g_1 = 5.3$*;* $g_2 = 4$*;* $g_3 = 2$*;* $g_4 = 9$*;* $g_5 = 3.9$*;* $g_6 = 2.2$.

Chapter 5

Applications to Recommender Systems

This chapter proposes the application of the GDM models and techniques defined in the first part of this thesis in the domain of *Recommender Systems* (RSs). In recent years RSs have become increasingly popular to handle the information overload problem. They are currently adopted in a variety of areas including movies, music, news, books, research articles, search queries, social tags, and products in general. Although the majority of RSs provides recommendations for individual users, there are several activities that can be performed by groups of people, like watching a movie, going to a restaurant or traveling with friends. In such cases, recommendations should by targeted to groups rather than individuals and the preferences of any (or the majority of) group members must be taken into account together.

Group Recommender Systems (GRSs) are RSs targeting groups of users. In addition to the previous cases, they can also play an important role in *Ambient Intelligence*, supporting applications that sense the environment and respond to the presence of people with personalized content. As most physical environments are used by many people at the same time, once their profiles are inferred or retrieved (e.g. via sensors, smart devices, RFID systems, etc.) GRSs can be used to select the most feasible content meeting all preferences. Example include the selection of the products to advertise on digital signage or the background music to be played in physical stores to maximize the well-being of present customers with a view to increasing sales.

The majority of existing GRS approaches are based on the aggregation of either the preferences or the recommendations generated for individual group members. Nevertheless, in many contexts, the personality of group members, their influence and mutual relationships play an important role in the final decision adopted by the group. So, such social elements, should be taken into account in the recommendation process to provide better results.

This chapter, after having summarized the main existing approaches to RS and GRS and the main metrics to measure their performances, introduces a novel influence-based approach to group recommendations based on the fuzzy GDM models and techniques defined in the first part.

5.1 Recommendation Algorithms

A formal definition of the recommendation problem can be expressed in these terms: let $U = \{u_1, \dots, u_m\}$ be a set of users, $X = \{x, \dots, x_n\}$ a set of items that can be recommended, R a totally ordered set whose values represent the utility of an item for a user (e.g. integers between 1 and 5 or real numbers between 0 and 1) and $f\colon U \times X \to R$ a utility function measuring how an item $x \in X$ is useful for an user $u \in U$; the purpose of a RS is to recommend, to each user u, the item x^* that maximizes the utility function so that [94]:

$$x^* = \operatorname*{argmax}_{x \in X} f(u, x) \tag{67}$$

The central problem of RSs is that f is not completely defined over the space $U \times X$ in fact, in typical applications, a user never expresses preferences on each available item. A RS shall then be able to estimate the values of the utility function also in the space of data where it is not defined, extrapolating from the points of $U \times X$ where it is known. In other words, the goal is to predict the rating that an user would give to an unknown item.

The techniques, by which it is possible to predict unknown ratings, are a fundamental aspect of RSs. In ***content-based approaches*** [95], the utility $f(u, x)$ of an unknown item x for the user u is predicted by considering defined values of f for items that are considered similar to x. For example, in an application for movies recommendation, the RS would try to understand the similarities between the movies that the user has positively rated in the past and those currently available (e.g. same genre, same director, common actors, etc.). After that, only items with high similarity would be proposed.

In such applications, each item $x \in X$ is associated with a profile, i.e. a set of attributes able to characterize the content, that is represented by a vector $p(x) = (w_{x,1}, \dots, w_{x,k})$ where $w_{x,i}$ with $i \in \{1, \dots, k\}$ is the weight of the i-th attribute or an indication of how the i-th attribute is able to characterize the item x. Weights can be either automatically generated (e.g. the frequency of keywords in text-based items) or manually provided (e.g. the presence or absence of a specific tag associated with the item).

Each user $u \in U$ is also associated with a profile $p(u) = (w_{u,1}, \dots, w_{u,k})$ where each weight $w_{u,i}$ with $i \in \{1, \dots, k\}$ denotes the importance of the i-th attribute for the user u. The user profile is based on the attributes of the items preferred by the user in the past. In the simplest formulation it can be obtained by averaging all profiles of the items for which u has expressed a rating and weighting them on the basis of the rating itself.

Once the profiles that characterize items and users have been defined, the utility of an unrated item x for an user u is calculated basing on the similarity between the two profiles. In other words $f(u,x) = sim\big(p(u), p(x)\big)$. Several similarity measures can be used for this purpose. One of the most common is the *cosine similarity* that calculates the cosine of the angle between the two vectors as follows:

$$sim\big(p(u), p(x)\big) = \frac{\sum_{i=1}^{k} w_{u,i} w_{x,i}}{\sqrt{\sum_{i=1}^{k} w_{u,i}^2} \cdot \sqrt{\sum_{i=1}^{k} w_{u,i}^2}} \tag{68}$$

The main advantage of this approach is that recommendations are only based on information related to domain items: first useful recommendations are so made immediately, with only one assessment available. On the other hand it tends to over-specialize predictions, therefore making them obvious and, consequently, uninteresting.

In ***collaborative approaches*** [96], unknown ratings are estimated from those made available by other users. The basic idea is that users who have evaluated the same items in a similar way, are likely to have the same tastes. *User-based* algorithms predict the utility $f(u, x)$ of an unrated item x for the user u by aggregating the utility expressed for x by similar users. One of the simplest aggregation functions is the average of ratings given by similar users, weighted on the degree of similarity as follows:

$$f(u,x) = \frac{\sum_{u' \in U_k} f(u',x) \cdot sim(u,u')}{\sum_{u' \in U_k} |sim(u,u')|} \tag{69}$$

where $U_k \subseteq U$ is the set of the k users considered most similar to u (with k chosen between 1 and the total number m of users).

The similarity among users is calculated on the vectors $\left(w_{u,x_1}, \cdots, w_{u,x_n}\right)$ that represent the ratings defined by an user $u \in U$ where $w_{u,x} = f(u,x)$, if defined, and $x \in X$. Several similarity measures exist to calculate such user similarity. Among them, one of the most commonly used is the *Pearson's correlation coefficient* defined as follows:

$$sim(u,u') = \frac{\sum_{x \in X} (w_{u,x} - \bar{u}) \left(w_{u',x} - \overline{u'}\right)}{\sqrt{\sum_{x \in X} (w_{u,x} - \bar{u})^2} \cdot \sqrt{\sum_{x \in X} \left(w_{u',x} - \overline{u'}\right)^2}} \tag{70}$$

where $\bar{u}$ and $\overline{u'}$ represent the mean rating assigned by users u and u'.

The advantage of computing recommendations basing on user similarity is to provide less obvious advice with respect to content-based approaches. On the other hand, when users provide few ratings, it is difficult to correlate them leading to inaccurate recommendations. *Item-based* algorithms [97] try to address this problem by estimating the utility of an unrated item x by aggregating the utility expressed by u to similar items. A simple aggregation function can be obtained by modifying equation (69) as follows:

$$f(u,x) = \frac{\sum_{x' \in X_k} f(u,x') \cdot sim(x,x')}{\sum_{x' \in X_k} |sim(x,x')|} \tag{71}$$

where $X_k \subseteq X$ is the set of the k items considered most similar to x (with k chosen between 1 and the total number n of items).

The similarity between two items is computed using the aforementioned similarity measures (68), (70) on the vectors $\left(w_{u_1,x}, \dots, w_{u_m,x}\right)$ that represent the ratings defined by the system users for the item $x \in X$ i.e. $w_{u,x} = f(u,x)$, if defined, and $u \in U$. This approach is capable of providing fairly accurate recommendations also to users who have rated only few items.

Collaborative approaches suffer of a *normalization issue* due to the fact that each user adopts its own personal scale to provide ratings. This lead to inaccurate results when ratings provided by different users are compared without normalization. The most popular normalization schemes are [98]:

- *mean-centering* – the mean rating provided by an user (or for an item in item-based approaches) is subtracted to each rating before calculating similarities between users (or items in item-based approaches);
- *z-score* – mean centered ratings are divided by the standard deviation of user ratings (or item ratings in item based approaches) before calculating similarities.

In ***model-based approaches*** [99] the history of the RS in not directly used to make predictions but to learn a model that is then used to generate recommendations. Popular implementations rely on *matrix factorization* [100], that map users and items to a latent factor space of dimensionality $d \ll m, n$. Each item $x \in X$ is then associated with a vector $q_x \in R^d$ and each user $u \in U$ with a vector $p_u \in R^d$. The elements of q_x measure the extent to which the item x possesses latent factors while the elements of p_u the extent of interest the user u has in items that are high on the corresponding factors.

The dot product between q_x and p_u captures the interaction between the user u and the item x representing the user's overall interest in the item's characteristics. The utility function can be so obtained as $f(u,x) = q_x^T p_u$.

The main challenge of this approach is to compute the mapping of each item and user to the latent factors. A common approach is to minimize the error on the set of known ratings as follows [101, 102]:

$$\min_{p,q} \sum_{(u,x)\in\kappa} (f(u,x) - q_x^T p_u)^2 + \lambda(\|q_x\|^2 + \|p_u\|^2) \tag{72}$$

where κ is the set of pairs (u, x) so that $f(u, x)$ is known (i.e. items that have been explicitly rated by users) and the constant λ controls the regularization extent and is usually determined by cross-validation.

Latent factor models combine good scalability with predictive accuracy. In addition, they are well suited to modeling temporal effects, which can significantly improve accuracy. In real applications, in fact, items perception and popularity constantly change as new selections emerge and, similarly, users' inclinations evolve, leading them to redefine their taste.

5.2 Measuring Recommendation Performances

RSs have several properties that may affect user experience and, connected to them, there are different metrics aimed at measuring RS performance with respect to each property. Among RS properties, the ***accuracy*** is one of the most discussed in RS literature. It may take different forms according to the way it is measured. The *rating prediction accuracy* measures the ability of the system to correctly predict unknown user ratings. In such cases RMSE and MAE metrics, already discussed in section 4.4 are commonly applied between predicted utilities and assigned ratings [103].

Nevertheless, in many applications, the final aim of a RS is the generation of useful recommendations rather than the ratings prediction. In these cases, the RS ends up with a list of recommended items for any user and measures for *usage prediction accuracy* can be applied to determine how correctly such lists predict how users will select available items in the future. In particular, each recommendation is capable of producing four different outcomes:

- *true positive* i.e. a recommended item is selected;
- *false positive* i.e. a recommended item is not selected;
- *true negative* i.e. a non-recommended item is not selected;
- *false negative* i.e. a non-recommended item is selected.

By counting the number of items that fall into each category it is possible to compute the following measures:

$$precision = \frac{\#tp}{\#tp + \#fp}; \quad recall = \frac{\#tp}{\#tp + \#fn} \tag{73}$$

where $\#tp$, $\#fp$ and $\#fn$ are, respectively, the number of true positives, false positives and false negatives. Longer recommendation lists typically improves recall reducing precision and vice-versa. For this reason, when possible, it is useful to compute curves comparing precision to recall with different lengths of the recommendation lists. A measure that summarizes precision and recall in a single value is the *F-measure* defined as follows [104]:

$$F = 2 \cdot \frac{precision \cdot recall}{precision + recall} \tag{74}$$

Sometimes an RS ends-up with a ranked list of recommendations for each user rather than with a flat list. In such cases *ranking prediction accuracy* measures can be used. When true rankings of all available items are known, rank correlation measures such as *Kendall's* τ or *Spearman's* ρ, introduced in section 2.1, are applicable. In the more realistic case that true rankings are available just for some items, alternative measures must be used.

The *normalized rank score* (NRS) metric, defined in [105], extends usage prediction accuracy metrics taking into account the position of recommended items. In particular, a decreasing utility is associated to the position in the item rank basing on the assumption that later positions have a higher chance of being overlooked. Given an user $u \in U$ the *rank score* can be defined as:

$$RS_u = \sum_{x \in tp} \frac{1}{2^{\frac{rank(x)-1}{\alpha}}} \tag{75}$$

where *tp* is the set of true positives (i.e. recommended items that are actually selected by the user), $rank(x)$ is the position of x in the recommendation list, α is parameter setting the half-life of utilities i.e. so that a successful hit at the first position has twice as much utility than one at the $\alpha + 1$ rank.

NRS normalizes RS by the maximum achievable score if all selected items are assigned to the lowest position in the recommendation list. Let *fn* be the set of false negatives, the NRS for an user $u \in U$ can be defined as:

$$NRS_u = \frac{RS_u}{RS_u^{max}} \quad \text{where} \quad RS_u^{max} = \sum_{i=1}^{|tp \cup fn|} \frac{1}{2^{\frac{i-1}{\alpha}}}. \tag{76}$$

The *normalized discounted cumulative gain* (NDCG) is a similar measure where positions are discounted logarithmically [106]. Let $X_u = (x_1^u, \dots, x_k^u)$ be a recommendation list generated for an user $u \in U$, the *discounted cumulative gain* of such list can be defined as:

$$DCG_u = \sum_{i=1}^{k} \frac{f(u, x_i^u)}{\log_2(i+1)} \tag{77}$$

where $f(u, x_i^u)$ is the real utility i.e. the true rating provided by the user u for the item x_i^u. NDCG normalizes DCG by the maximum achievable score i.e. considering $NDCG_u = DCG_u / DCG_u^{max}$ where DCG_u^{max} is the value that DCG can get by ordering recommended items according to the true ratings.

The calculation of NDCG relies on the assumption that true ratings are available for any recommended item. However in most cases users express a rating only for some items of the recommendation list. To overcome this issue, in [107] it was suggested to compute NDCG just on the subset of ranked items included in the recommendation list, sorted according to the ranking computed by the recommendation algorithm.

The prediction accuracy of a RS usually improves when the amount of available data increases. Nevertheless, in many cases, recommendations can be generated only on a portion of available data about items and users. The ***coverage*** of a RS measure the size of this portion. In particular, the *catalog coverage* is the percentage of available items which are recommended to some user i.e. the size of the union of all the recommendation lists divided by the number of available items. The more general *prediction coverage* represents instead the percentage of available items for which a recommendation can be generated. Similarly, the *user-space coverage* is the percentage of users for which a recommendation can be generated [108].

Often, the recommendation lists generated by a RS contains many similar items making them of limited value for users. In fact, in such cases, it may take longer to explore the full range of recommendations. To measure the ability of a RS to avoid this issue, some measures of ***diversity*** have been proposed [109]. Among them, one of the most used is the average distance between item pairs. Let $X_u = (x_1^u, \dots, x_k^u)$ be the recommendation list for an user $u \in U$, its diversity measure can be obtained as:

$$diversity_u = 1 - 2 \cdot \frac{\sum_{i=1}^{k-1} \sum_{j=i+1}^{k} sim\left(p(x_i^u), p(x_j^u)\right)}{n \cdot (n-1)} \tag{78}$$

where $sim\left(p(x_i^u), p(x_j^u)\right)$ denotes a similarity metric defined in [0,1] (like the cosine similarity) applied on the profiles characterizing the content of items x_i^u and x_j^u as described in section 5.1.

RSs might generate recommendations with high accuracy and reasonable diversity and coverage but that are useless for practical purposes [108]. For example, it happens when an RS makes obvious recommendations involving popular items or items that users would have chosen even without them being recommended. Conversely, the most valuable recommendations involve items users have never heard of, but would love. To measure the attitude of a RS to generate surprisingly successful recommendations, ***serendipity*** metrics have been introduced [110].

Several measures of serendipity exist. In [111] it was suggested to obtain serendipity as linear composition of *novelty* and *unpopularity.* The novelty of an item $x \in X$, recommended to an user $u \in U$, is calculated as the distance between the profile representing x from that representing u. Using a similarity metric defined in [0,1] it can be so obtained as $1 - sim\big(p(u), p(x)\big)$. Instead, the unpopularity of x can be obtained as $1 - m_x/m$ where m is the total number of users while m_x is the number of users who selected the item x in the past (or who gave it a positive rating).

5.3 Group Recommendation Strategies

Although the majority of RSs are designed to generate recommendations for individual users, in many circumstances the selected content is consumed in groups. Typical cases include the selection of movies or TV shows to be watched in a family context, the selection of restaurants, bars or cultural events for friends coming out together, the selection of holiday destinations for travel groups, etc. In such cases, provided recommendations should fit the preferences of any (or the majority of) group members [112].

Several group recommendation strategies have been proposed so far by different researchers. In [113], they have been classified in two broad classes depending on the stage in which information about individual group members is aggregated to obtain suggestions for the whole group. *Recommendations aggregation strategies* foresee the generation of individual recommendations for group members through a standard algorithm as those seen in section 5.1. Then, the individual lists of recommended items are merged into a single list addressing the group as a whole. Different algorithms have been proposed to perform the aggregation step and to decide whether to include or exclude individual suggestions in the group list.

Preferences aggregation strategies, instead, combine users' preferences (in terms of profiles or assigned ratings) in a single model that is used to obtain recommendations for the group through a standard algorithm. In this way the group is seen as a pseudo-user reflecting the interests of all members. Compared to recommendations aggregation, these strategies increase the chance of finding unexpected, surprising items. On the other hand, group recommendations cannot be directly linked to individual preferences, which may be disorienting and makes them difficult to explain [114].

In GRSs, a central role is played by the algorithm employed to aggregate recommendations or preferences (according to the selected strategy). In [112, 115] several aggregation methods have been proposed as summarized below.

The ***average*** method, in case of *recommendations aggregation*, merges the individual recommendation lists provided by a standard recommendation algorithm by calculating, for each item belonging to at least one of these lists, the average utility among all group members. Let $U_G \subseteq U$ be the set of users belonging to a group and $X_G \subseteq X$ the set including all items recommended to at least one member, the group utility of any $x \in X_G$ is estimated as:

$$f(U_G, x) = \frac{\sum_{u \in U_G} f(u, x)}{\#U_G} \tag{79}$$

The elements of X_G with highest utilities are then proposed to the group. In case of *preferences aggregation*, instead, the average method uses equation (79) to estimate the group utility of any $x \in X$. Then, the items of X with the highest utilities are proposed to the group. When some group members have more influence on the group decision, a weighted average may be used.

The ***average without misery*** method looks for the optimal decision for the group, without making some members really unhappy with such decision. In case of *recommendations aggregation*, any element $x \in X_G$ so that $f(u, x)$ is below a given threshold for at least one user $u \in U_G$, is removed from X_G or receives a penalty in the calculation of the group utility value. Similarly, in case of *preferences aggregation,* any item $x \in X$ whose utility is below a given threshold for at least one group member, receives a penalty.

Like for the average method, equation (79) is used to aggregate remaining elements of X_G or X. Penalties can be applied by multiplying the obtained group utility to a penalizing factor between 0 and 1 or by using the minimum individual utility in place of the group utility.

The ***least misery*** method aims at minimizing the overall misery of the group by considering the minimum individual utility as the group utility for each available item. Conversely, the specular ***most pleasure*** method aims at maximizing the overall pleasure of the group by considering the maximum individual utility as the group utility. Within both methods, in case of *recommendations aggregation*, the estimation is done only for items in X_G while, in case of *preferences aggregation,* any item $x \in X$ is considered.

The ***multiplicative*** method obtains the group utility of any item $x \in X$ by multiplying together the individual estimated utilities $f(u, x)$ of x for any group member $u \in U_G$:

$$f(U_G, x) = \prod_{u \in U_G} f(u, x) \tag{80}$$

In case of *recommendations aggregation*, the estimation is done only for items in X_G while, in case of *preferences aggregation,* any item $x \in X$ is considered.

The ***most respected person*** method adopts the recommendations made for the most influencing member as the group preferences. This method needs to know the influence level of group members.

In [112] an experiment was made to identify how users perceive group recommendations obtained with different aggregation strategies. Participants were given individual ratings for sample items and users as well as item sequences chosen by the aggregation strategies. They rated how satisfied they thought group members would be with generated sequences, and explained their ratings. According to participants, the multiplicative method performed best followed by average, average without misery and most pleasure.

When recommending items to groups of users, it is impossible to equally satisfy each member at all times. On the other hand, it is important that no one remains dissatisfied too many times. This leads to the need of balancing user satisfaction over time. In [116] several *satisfaction function* have been proposed to model members' satisfaction during multiple interactions with a group recommender system. Let $x^{(1)}, \dots, x^{(k)} \in X$ be the sequence of the last k proposed items for the group of users U_G, the current satisfaction $sat(u)$ of an user $u \in_G$ can be defined as follows:

- *addition* – summing the utilities of the last k selected items for the user u i.e. $sat(u) = \sum_{t=1}^{k} f(u, x^{(t)})$;
- *addition with normalization* – dividing the sum of the utilities of the last k selected items by the sum of the utilities of the top l preferred items for u in X;
- *addition with decay* – summing the utilities of the last k selected items weighted according to a decay function to give more importance to last selected items i.e. $sat(u) = \sum_{t=1}^{k} e^{1-t} \cdot f(u, x^{(t)})$

When the satisfaction of each group member has been estimated through one of the previous methods, the preference aggregation function may use these values to improve recommendations. For example the item which is most liked by the least satisfied user can be proposed to the group or the top k preferred items for the least l satisfied users can be pre-selected in X_G and a recommendations aggregation strategy can be applied on X_G to generate the final suggestions.

5.4 A GDM Model for Group Recommendation

The group recommendation problem, as formalized in section 5.3, can be seen as a special case of the GDM problem. In GDM, a group of experts evaluate a set of alternatives with the aim of selecting the best one to adopt. To this end, each expert expresses her preferences on alternatives, preferences are aggregated, a collective preference degree of each alternative is calculated and a ranking over alternatives is generated.

Similarly, the aim of GRSs, is to select from a given catalogue the item or the set of items that fit the preferences of all (or the majority of) members belonging to a group of users. Differently from GDM, users do not need to explicitly state their preferences on items but an utility function, estimating such preferences, is already available as output of a standard RS algorithm. As in GDM, user preferences must be aggregated, the collective preference degree of each item calculated and a ranking over items generated.

Let X_u be the recommendation list built for an user $u \in U$ by a standard RS algorithm as those described in section 5.1, $U_G \subseteq U$ a group of users and $X_G = \bigcup_{u \in G} X_u$ the set of items recommended to at least one group member. By considering U_G as a set of experts and X_G as a set of alternatives, we can translate the GRS problem in a GDM one. Then, the estimated utility $f(u,x)$, with $x \in X_G$ and $u \in U_G$, naturally represents experts' preferences.

In some implementations, the RS only provides, for each user, the list of suggested items without specifying their utilities. In other implementations only the utility of suggested items is known. Instead, to instantiate a GDM problem corresponding to a GRS one, the underling RS should be able to also estimate the utility of a non-recommended item. This is needed to estimate preferences users have for items recommended to other group members. In other words, values for $f(u,x)$ must be generated for any $x \in X_G$ and for any $u \in U_G$. Just in case of limited coverage i.e. when, due to limited data, it is impossible to estimate some utility values, the corresponding preferences of the GDM problem remain undefined.

It is important to consider that GDM problems, instantiated in this way, found the decision process on predicted utility values rather than on explicit preference statements collected among experts. This suggests to rely on GDM approaches that are intrinsically able to deal with the uncertainty resulting from prediction inaccuracy, like those based on the fuzzy set theory. As seen in section 1.4, such approaches foresee preference modeling in terms of FPRs whose elements must be defined starting from predicted utilities.

Given an utility function $f\colon U_G \times X_G \to [0,1]$, estimated by a standard RS, with $U_G = \{u_1^G, \dots, u_m^G\} \subseteq U$ and $X_G = \{x_1^G, \dots, x_n^G\} \subseteq X$, the corresponding FPR $P_k = (p_{ij}^k)$, representing preferences of each group member $u_k^G \in U_G$, must be defined. According to [2], utility values

$(r_1, \dots, r_n)$, normalized in [0,1], can be transformed into a FPR by any function H: $[0,1] \times [0,1] \to [0,1]$ so that the following conditions are satisfied:

- $H(r_i, r_j)$ is a non-decreasing function of the first argument and a non-increasing function of the second one $\forall\ i, j \in \{1, \dots, n\}$;
- $H(r_i, r_i) = 0.5\ \forall\ i \in \{1, \dots, n\}$;
- $H(r_i, 0) = 1\ \forall\ i \in \{1, \dots, n\}$ to reflect the fact that, if the utility of an alternative is zero, then any other alternative should be preferred to it with the maximum preference degree;
- $H(r_i, r_j) > 0.5$ iif $r_i > r_j\ \forall i, j \in \{1, \dots, n\}$;
- $H(r_i, r_j) + H(r_j, r_i) = 1\ \forall i, j \in \{1, \dots, n\}$ that is the additive reciprocity property described in section 1.4.

Several utility-to-FPR transformation functions exist. Among them, in [2], the following function that build FPRs satisfying the additive transitivity property too (as defined in section 1.4) has been proposed:

$$H(r_i, r_j) = \frac{1 + r_i - r_j}{2}. \tag{81}$$

Applying equation (81) on the utility function *f*, it is possible to obtain the elements of the FPRs P_k with $k \in \{1, \dots, m\}$, associated to any group member as follows:

$$p_{ij}^k = \frac{1 + f(u_k^G, x_i^G) - f(u_k^G, x_j^G)}{2}. \tag{82}$$

Example 21. *Let $U_G = \{u_{123}, u_{335}, u_{467}\}$ be the subset of RS users belonging to a given group G; $X_{123} = \{x_{12}, x_{25}, x_{39}, x_{77}\}$, $X_{335} = \{x_{12}, x_{46}, x_{67}, x_{77}\}$ $X_{467} = \{x_{39}, x_{46}, x_{77}, x_{89}\}$ the recommendation lists built for U_G members by a standard RS; $X_G = X_{123} \cup X_{335} \cup X_{467} = \{x_{12}, x_{25}, x_{39}, x_{46}, x_{67}, x_{77}, x_{89}\}$ the set of items recommended to at least one group member. By assuming that the individual utilities for items in X_G are those summarized in Table 11 (where – represents an undefined prediction), it is possible to build the FPR associated to u_{123} according to equation (82) as follows:*

$$P_{123} = \begin{pmatrix} 0.50 & 0.55 & 0.50 & 0.75 & 0.90 & 0.60 & - \\ 0.45 & 0.50 & 0.45 & 0.70 & 0.85 & 0.55 & - \\ 0.50 & 0.55 & 0.50 & 0.75 & 0.90 & 0.60 & - \\ 0.25 & 0.30 & 0.25 & 0.50 & 0.65 & 0.35 & - \\ 0.10 & 0.15 & 0.10 & 0.35 & 0.50 & 0.20 & - \\ 0.40 & 0.45 & 0.40 & 0.65 & 0.80 & 0.50 & - \\ - & - & - & - & - & - & - \end{pmatrix}.$$

Group members	Individual item utilities						
	x_{12}	x_{25}	x_{39}	x_{46}	x_{67}	x_{77}	x_{89}
u_{123}	1.00	0.90	1.00	0.50	0.20	0.80	–
u_{335}	0.90	0.30	0.60	1.00	0.90	1.00	0.10
u_{467}	0.10	0.30	0.90	0.80	–	0.80	1.00

Table 11. Individual item utilities used in Example 21

Given that, in the example, the RS is unable to predict the utility of x_{89} for the user u_{123}, the values corresponding to the last row and column of P_{123} remain undefined and should be estimated with one of the methods proposed in section 1.7. For example, assuming the indifference between x_{89} and any other item and iterating equations (22)-(23) until convergence, the following complete version of P_{123} is obtained:

$$P_{123} = \begin{pmatrix} 0.50 & 0.55 & 0.50 & 0.75 & 0.90 & 0.60 & 0.61 \\ 0.45 & 0.50 & 0.45 & 0.70 & 0.85 & 0.55 & 0.57 \\ 0.50 & 0.55 & 0.50 & 0.75 & 0.90 & 0.60 & 0.61 \\ 0.25 & 0.30 & 0.25 & 0.50 & 0.65 & 0.35 & 0.40 \\ 0.10 & 0.15 & 0.10 & 0.35 & 0.50 & 0.20 & 0.27 \\ 0.40 & 0.45 & 0.40 & 0.65 & 0.80 & 0.50 & 0.53 \\ 0.39 & 0.43 & 0.39 & 0.60 & 0.73 & 0.47 & 0.50 \end{pmatrix}.$$

When all individual FPRs P_k with $k \in \{1, \dots, m\}$ are obtained, they must be aggregated to obtain the collective FPR P through one of the functions defined in section 1.5, like OWA_Q. Then, it is possible to calculate the group preference $\phi(x_i^G)$ for each item $x_i^G \in X_G$ according to one of the measures defined in section 1.6 (i.e. NF, NDD, QGDD or QGNDD). The global ranking between the items is then computed by ordering them decreasingly on their group preference degree and the top-ranked elements can be recommended to the group. In addition, if needed, an estimation of the group utility of each item $x \in X_G$ can be obtained through normalization as follows:

$$f(U_G, x) = \frac{(\phi(x) - \phi_{min}) \cdot (f_{max} - f_{min})}{(\phi_{max} - \phi_{min})} + f_{min} \tag{83}$$

where ϕ_{min} and ϕ_{max} are, respectively, the minimum and maximum values of the group preference $\phi(x_i^G)$ for $i \in \{1, \dots, n\}$, while f_{min} and f_{max} are the minimum and maximum of the utility function $f(u_k^G, x_i^G)$ for $k \in \{1, \dots, m\}$ and $i \in \{1, \dots, n\}$.

Example 22. *Let U_G, X_G and P_{123} be as reported in the previous example and individual utilities of the X_G items as summarized in Table 11, by using equation (82) to obtain the FPRs P_{335} and P_{467}, equations (22)-(23) to complete P_{467} and equations (13)-(15) to aggregate FPRs with OWA_Q guided by the linguistic quantifier "much", the following collective FPR is obtained:*

$$P = \begin{pmatrix} 0.50 & 0.58 & 0.42 & 0.45 & 0.55 & 0.40 & 0.52 \\ 0.42 & 0.50 & 0.33 & 0.37 & 0.47 & 0.32 & 0.44 \\ 0.58 & 0.67 & 0.50 & 0.53 & 0.62 & 0.48 & 0.60 \\ 0.55 & 0.63 & 0.47 & 0.50 & 0.59 & 0.45 & 0.58 \\ 0.45 & 0.53 & 0.38 & 0.41 & 0.50 & 0.36 & 0.51 \\ 0.60 & 0.68 & 0.52 & 0.55 & 0.64 & 0.50 & 0.63 \\ 0.48 & 0.56 & 0.40 & 0.42 & 0.49 & 0.37 & 0.50 \end{pmatrix}.$$

Then, it is possible to obtain the group preference of each item in terms of QGDD through equation (20) and, in turn, their group utility with equation (83) as follows: $f(U_G, x_{12}) = 0.51$; $f(U_G, x_{25}) = 0.10$; $f(U_G, x_{39}) = 0.91$; $f(U_G, x_{46}) = 0.76$; $f(U_G, x_{67}) = 0.31$; $f(U_G, x_{77}) = 1$; $f(U_G, x_{89}) = 0.37$. The top-ranked items for group consumption so are: x_{77}, x_{39} and x_{46}.

An important aspect to take into account in the FPR aggregation process is the choice of the linguistic quantifier guiding the OWA_Q operator. While the selection of the quantifier "much" (like in Example 22) assigns the same importance to all individual utilities, by selecting a different quantifier it is possible to obtain a different behavior. For example, by using the "at least half" quantifier, it is expected that at least half of group users is satisfied with an item to recommend it. To this end, higher individual utilities are privileged with respect to lower ones. Conversely by using the "as many as possible" quantifier, it is expected that the majority of users is satisfied with an item so lower individual utilities are privileged over higher ones. By using the "most" quantifier, instead, lower-intermediate utilities are privileged over extreme (higher or lower) ones.

Example 23. *Let $U_G = \{u_1^G, u_2^G, u_3^G, u_4^G\}$ be the set of users belonging to a group G and $X_G = \{x_1^G, x_2^G, x_3^G, x_4^G\}$ the set of items recommended to at least one group member. Let assume that the estimated individual utility for each group member is that reported in Table 12, by using the defined GDM model to compute recommendations for the whole group we obtain different results according to the linguistic quantifier chosen for the OWA_Q operator during the FPR aggregation process. Table 13 summarizes the results obtained using different linguistic quantifiers and compare them with those obtained with standard group recommendations aggregators as seen in section 5.3.*
As it can be seen in Table 13, by averaging individual utilities (through the "average" aggregator) all items seem equally relevant to the group. The "average without misery" aggregator has a similar behavior but it excludes items x_1^G and x_4^G because their utility is too low for some users (u_1^G and u_2^G). The "least misery" and "most pleasure" aggregators use, in turn, the lower and the higher individual utility of each item while, the "multiplication" aggregator, privileges lower individual utilities over higher ones.

Group members	Individual item utilities			
	x_1^G	x_2^G	x_3^G	x_4^G
u_1^G	1.00	0.60	0.80	0.20
u_2^G	0.60	0.60	0.80	0.20
u_3^G	0.50	0.60	0.40	1.00
u_4^G	0.30	0.60	0.40	1.00

Table 12. Individual item utilities used in Example 23

Similarly, by adopting the defined GDM based model, different behaviors are obtained by choosing different quantifiers for the OWA_Q aggregator. Given that FPRs are built comparing, for each user, the individual utilities of different items, the GDM model operate on relative utilities rather than on absolute ones. To this end mean-centered individual utilities (where the mean is calculated with respect to each user) are reported in Table 14.

Aggregators	Ranked group utilities			
	1st	2nd	3rd	4th
average	x_1^G (0.60)	x_2^G (0.60)	x_3^G (0.60)	x_4^G (0.60)
average without misery	x_2^G (0.60)	x_3^G (0.60)	x_1^G (0.00)	x_4^G (0.00)
least misery	x_2^G (0.60)	x_3^G (0.40)	x_1^G (0.30)	x_4^G (0.20)
most pleasure	x_1^G (1.00)	x_4^G (1.00)	x_3^G (0.80)	x_2^G (0.60)
multiplication	x_2^G (0.13)	x_3^G (0.10)	x_1^G (0.09)	x_4^G (0.04)
at least half (GDM)	x_4^G (1.00)	x_1^G (0.35)	x_3^G (0.35)	x_2^G (0.20)
much (GDM)	x_1^G (0.50)	x_2^G (0.50)	x_3^G (0.50)	x_4^G (0.50)
most (GDM)	x_2^G (1.00)	x_3^G (0.65)	x_1^G (0.25)	x_4^G (0.20)
as many as possible (GDM)	x_2^G (1.00)	x_1^G (0.85)	x_3^G (0.85)	x_4^G (0.20)

Table 13. Ranked group utilities obtained using different aggregators

By considering again Table 13 in the light of the relative utilities shown in Table 14, it can be seen that the "much" quantifier in the GDM model behave like the "average" aggregator. The "at least half" quantifier, instead, privileges relative utilities associated to most enthusiastic users, so x_4^G wins thanks to its high estimated relative utility for users u_3^G and u_4^G. Conversely, the "as many as possible" quantifier privileges relative utilities associated to less enthusiastic users so x_2^G wins thanks to its high estimated relative utility for users u_1^G and u_3^G. Finally, the "most" quantifier privileges relative utilities associated to lower-intermediate users so, in this case, the winner is x_2^G.

Group members	Mean-centered individual utilities			
	x_1^G	x_2^G	x_3^G	x_4^G
u_1^G	0,35	-0,05	0,15	-0,45
u_2^G	0,05	0,05	0,25	-0,35
u_3^G	-0,12	-0,03	-0,22	0,38
u_4^G	-0,27	0,03	-0,17	0,43

Table 14. Mean-centered individual item utilities for group members

The previous example demonstrates the flexibility of the proposed model. In fact, it allows to design different aggregation strategies by simply selecting different linguistic quantifiers. Moreover, when needed, a new strategy can be introduced by simply defining a new quantifier.

5.5 Influence-Based Recommendations

When selecting an item for consumption within a group of users, often the final choice is deeply affected by the personality of group members. In fact, due to interpersonal influence, individual preferences may change during the selection process when information and opinions are exchanged in social interactions. In order to take social influence into account, we propose an improved GDM-based model for group recommendations that introduces social elements in accordance with the approach defined in chapter 3.

In particular, as proposed in section 3.1, the configuration and strength of social influence among group members is evaluated basing on interpersonal trust and represented within a SIN. The SIN, in its turn, is used to complete the generated FPRs in case of missing elements (e.g. due to limited coverage of the underlying RS) and to evolve them by incorporating elements captured from other FPRs simulating, in this way, the effects of social influence on opinion change. The process then iterates until convergence toward a shared FPR that is then used to build recommendations.

In analogy with the definition provided in section 3.1, a SIN represents a directed graph associating, to each pair of group members $(u_i^G, u_j^G) \in U_G^2$, a weight $w_{ij} \in [0{,}1]$ that measures the strength of the influence of the j-th member on the i-th one. SIN weights can be determined starting from explicit user-provided trust statements, like discussed in section 3.3, or inferred by analyzing past social interactions among group members e.g. by looking at implicit information contained in social networks like *Facebook* or *Twitter*. The main advantage of the second approach is that the process is completely transparent to users. On the other hand, it is required that all group members belong to the same social network but this issue is mitigated by the rising popularity of this kind of applications.

In [117] it was demonstrated that trust and tie strength are conceptually different but strongly correlated. In [118], 74 Facebook variables have been identified as potential predictors of tie strength. By relying on these results, in [119] interpersonal trust has been estimated as linear combination of 10 factors measured on Facebook profiles. Then, on the same paper, it has been demonstrated that a reliable estimation of trust strength can be obtained by just considering, for each $u_i^G, u_j^G \in U_G$ with $i \neq j$, the following 5 factors:

- $f_1(u_i^G, u_j^G)$ represents the amount of common friends between u_i^G and u_j^G, ranging from 0.1 (less than 5) to 1 (more than 25);

- $f_2(u_i^G, u_j^G)$ is the percentage of pictures where u_i^G and u_j^G appear together over the total number of pictures in the u_i^G profile;
- $f_3(u_i^G, u_j^G)$ is the duration of the relationship between u_i^G and u_j^G ranging from 0.1 (less than 1 year) to 1 (more than 10 years), obtained comparing information on age, schools, universities, work and family relations;
- $f_4(u_i^G, u_j^G)$ is the percentage of common interests described in the profiles of u_i^G and u_j^G (movies, books, joined groups, etc.) over the total number of interests declared in the u_i^G profile;
- $f_5(u_i^G, u_j^G)$ is the strength of the declared status between u_i^G and u_j^G ranging from 0.1 (barely know) to 1 (couple).

The trust level of any group member u_i^G in any other member u_j^G can be then obtained as weighted sum of such factors as follows:

$$trust(u_i^G, u_j^G) = \sum_{k=1}^{5} w_k f_k(u_i^G, u_j^G) \tag{84}$$

where $i \neq j$ and the weights w_k with $k \in \{1, \ldots, 5\}$ are chosen experimentally so that $\sum_{k=1}^{5} w_k = 1$. A feasible set of weights is: $w_1 = 0.4$; $w_2 = w_3 = 0.2$; $w_4 = 0.15$ and $w_5 = 0.05$.

Once the interpersonal trust among group members is estimated, to build a SIN, it is still needed to estimate users' *self-confidence*. Such value measures the attitude of an user to remain faithful to her initial preferences, mitigating the effects of social influence. In [119], a similar attribute is estimated based on the *Thomas-Kilmann Conflict Mode Instrument* (TKI), a test made of 30 questions with two possible answer each [120].

TKI define five personality modes of dealing with conflicts: competing, collaborating, avoiding, accommodating and compromising. Depending on the answers provided to test questions, a score is assigned to each personality mode. Then, the obtained results are summarized along two basic dimensions: *assertiveness* and *cooperativeness* through a weighted sum of the obtained scores. Given the assertiveness $a(u_i^G)$ and the cooperativeness $c(u_i^G)$ of an user$u_i^G \in U_G$ obtained in this way and assuming that both values are defined in [0,1], the self-confidence of u_i^G can be obtained as follows:

$$self(u_i^G) = \frac{1 + a(u_i^G) - c(u_i^G)}{2} \tag{85}$$

A problem of this approach is that it requires that group members fill a 30-questions test before start using the system. Nevertheless, several studies [121, 122] correlate conflict management styles with the so-called *five-factors personality traits* (extraversion, agreeableness, conscientiousness, neuroticism and openness). Basing on these studies, if the personality traits are known, it is possible to estimate the levels of assertiveness and cooperativeness of a given user and, through equation (85), the *self-confidence* too.

Several types of test exist to estimate such personality factors like the *Five-Factor Personality Inventory* or the *Revised NEO Personality Inventory* [123]. Unfortunately, such approaches suffer from the same limitations seen for the direct estimation of the conflict management styles i.e. users are needed to fill long questionnaires before system use. Nevertheless, some approaches exist to predict personality directly from the language used in social media. For example, in [124], an algorithm for the prediction of the five-factors traits from the textual analysis of users' Facebook status updates is defined. Moreover, from a similar work [125], the on-line tool *Apply Magic Sauce*[1] for personality prediction from Facebook has been implemented.

Once $self(u_i^G)$ and $trust(u_i^G, u_j^G)$ are estimated for $i, j \in \{1, \dots, m\}$, it is possible to obtain SIN weights as follows:

$$w_{ij} = \begin{cases} \left(1 - self(u_i^G)\right) \cdot \dfrac{trust(u_i^G, u_j^G)}{\sum_{k \in \{1,\dots,m\} \setminus i} trust(u_i^G, u_k^G)} & \text{if } i \neq j, \\ self(u_i^G) & \text{if } i = j. \end{cases} \tag{86}$$

[1] https://applymagicsauce.com/

The special case where $trust(u_i^G, u_j^G) = 0 \ \forall\ j \in \{1, \dots, m\} \setminus i$ (i.e. when an user does not trust any other user) is handled by setting $w_{ii} = 1$ and $w_{ij} = 0$ $\forall\ j \in \{1, \dots, m\} \setminus i$. It is trivial to demonstrate that the so obtained matrix W fulfills the normalization property defined by equation (41).

Example 24. *Let $U_G = \{u_{123}, u_{335}, u_{467}\}$ be the subset of RS users belonging to a given group G and that $trust(u_{123}, u_{335}) = 0.3$; $trust(u_{335}, u_{123}) = 0.8$; $trust(u_{123}, u_{467}) = 0.9$; $trust(u_{467}, u_{123}) = 0$; $trust(u_{335}, u_{467}) = 0.8$ and that $trust(u_{467}, u_{335}) = 0.8$. Let also assume $self(u_{123}) = 0.5$; $self(u_{335}) = 0.2$ and $self(u_{467}) = 0.8$, according to equation (86) it is possible to obtain the SIN represented by the following matrix:*

$$W = \begin{pmatrix} 0.5 & 0.12 & 0.38 \\ 0.4 & 0.2 & 0.4 \\ 0 & 0.2 & 0.8 \end{pmatrix}.$$

The obtained SIN is used to complete the generated FPRs in case of missing elements. According to section 3.4, seed values for missing elements are obtained from FPRs of group members that are trusted by the one whose FPR has to be completed through equation (47). Then, the final estimates are computed through the iterative application of equations (22)-(23) until convergence is reached.

To simulate the effects of social influence between group members, the individual FPRs obtained at the preceding steps are evolved using the SIN. According to section 3.5, an iterative process is applied where, at each step, the individual FPR of each group member is slightly changed to take into account the influence of trusted members through equation (49). When the stopping conditions defined by equation (50) are met, in case of lack of convergence, the obtained FPRs are aggregated through the OWA_Q operator as defined in section 1.5.

Finally, the group preference $\phi(x_i^G)$ for each item $x_i^G \in X_G$ is calculated as described in section 5.4 on the collective FPR, the global ranking between the items is computed and the top-ranked elements are recommended to the group. Optionally, an estimation of the group utility of each item $x \in X_G$ can be obtained through equation (83).

Example 25. *Let U_G and X_G be as defined in Example 21 and the individual utilities for items in X_G as reported in Table 11, through equation (82) it is possible to build the FPR associated to each group member. By relying on the SIN adjacency matrix W defined in Example 24, it is then possible to estimate missing FPR elements with equation (47) followed by the iterative application of equations (22)-(23). The obtained FPR for user $u_{123} \in U_G$ is reported below and should be compared to that obtained in Example 21 by applying the "indifference" estimation strategy.*

$$P_{123} = \begin{pmatrix} 0.50 & 0.55 & 0.50 & 0.75 & 0.90 & 0.60 & 0.58 \\ 0.45 & 0.50 & 0.45 & 0.70 & 0.85 & 0.55 & 0.54 \\ 0.50 & 0.55 & 0.50 & 0.75 & 0.90 & 0.60 & 0.62 \\ 0.25 & 0.30 & 0.25 & 0.50 & 0.65 & 0.35 & 0.41 \\ 0.10 & 0.15 & 0.10 & 0.35 & 0.50 & 0.20 & 0.33 \\ 0.40 & 0.45 & 0.40 & 0.65 & 0.80 & 0.50 & 0.54 \\ 0.42 & 0.46 & 0.38 & 0.59 & 0.67 & 0.46 & 0.50 \end{pmatrix}.$$

Such FPR, together with those obtained for $u_{335}, u_{467} \in U_G$ (not reported for shortness) are then evolved according to the process described in section 3.5 to simulate the effects of social interaction. After 6 iterations, all individual FPRs converge to the following FPR:

$$P = \begin{pmatrix} 0.50 & 0.50 & 0.26 & 0.30 & 0.34 & 0.28 & 0.29 \\ 0.50 & 0.50 & 0.27 & 0.30 & 0.30 & 0.28 & 0.29 \\ 0.74 & 0.73 & 0.50 & 0.53 & 0.52 & 0.51 & 0.53 \\ 0.70 & 0.70 & 0.47 & 0.50 & 0.51 & 0.48 & 0.50 \\ 0.66 & 0.70 & 0.48 & 0.49 & 0.50 & 0.47 & 0.56 \\ 0.72 & 0.72 & 0.49 & 0.52 & 0.53 & 0.50 & 0.52 \\ 0.71 & 0.71 & 0.47 & 0.50 & 0.44 & 0.48 & 0.50 \end{pmatrix}.$$

Then, it is possible to obtain the group preference of items through equation (20) and, with equation (83), their group utility as follows: $f(U_G, x_{12}) = 0.11$; $f(U_G, x_{25}) = 0.10$; $f(U_G, x_{39}) = 1.00$; $f(U_G, x_{46}) = 0.89$; $f(U_G, x_{67}) = 0.89$;

$f(U_G, x_{77}) = 0.97$*;* $f(U_G, x_{89}) = 0.85$*. The top-ranked items for the group so are:* x_{39} *and* x_{77} *followed by* x_{67} *and* x_{46}*.*
Compared with the results obtained in Example 22, we see that now x_{39} *is preferred over* x_{77} *even if it has an higher average of individual utilities. This is because* x_{39} *is preferred by users* u_{123} *and* u_{467} *that, according to the SIN, are more influencing than* u_{335} *(who prefers* x_{77} *instead). It should also be noted the good position reached of* x_{67} *due to the fact that the opinion of the influencing member* u_{467} *(initially unknown) is formed on that of* u_{335} *(that likes* x_{67}*) disregarding that of the untrusted member* u_{123}*(that dislikes* x_{67}*).*

5.6 Comparison with Related Works

Several GRSs have been proposed in the literature. Among the first systems there is *MusicFX* [126] that selects background music to be played in a fitness center to suit the group of people expected to exercise at a given time. User profiles are generated with an interview and the music selection is based on a variant of the least misery aggregation strategy (seen in section 5.3) that includes some randomness to avoid always choosing the same music. Another GRS for the selection of ambient music is *Flytrap* [127] that generates user profiles starting from the music people listen to on their computers and uses RFID badges to detected people present in the room.

Polylens [128] is a group extension of the popular *Movielens* system for movies recommendation[2]. It allows users to create groups and ask for group recommendations that are built by aggregating individual recommendations (generated basing on users' star ratings) through the least misery strategy and avoiding movies already seen by any group member. In the field of TV shows, *Yu's TV* [129] recommends television programs for families. It bases recommendations on the average strategy applied on individual preferences for program features (e.g. genre, directors, actors, etc.). *Family Interactive*

[2] https://movielens.org/

TV [130] also filters television programs according to the viewers' preferences and uses implicit relevance feedback assessed through the actual program the viewer has chosen for watching.

In the touristic field, the *Travel Decision Forum* [131] assists a group to agree on the desired attributes of a planned joint holiday. Users indicate their preferences on a set of features (room facilities, sightseeing attractions in the surrounding area, etc.), preferences are then aggregated and a mediator agent supports users to reach consensus. The *Collaborative Advisory Travel System* [132] is a similar system that induces group members' profiles by proposing holiday packages and collecting critiques on their features.

The *Pocket Restaurant Finder* [133] delivers restaurant recommendations for groups that are planning to go out eating together. The application bases recommendations on individual preferences related to cuisine type, restaurant amenities, price category, etc. also taking into account the physical location of users and restaurants. *Intrigue* [134] is a GRS for touristic places which build recommendations by relying on a single group profile obtained from the characteristic of the group (e.g. presence of children or disabled) as well as from the aggregation of individual preferences.

Beyond application-specific works, some studies evaluate the performance of different aggregation strategies for GRSs. A main issue of this task is that the majority of RS datasets just include single-user data. To overcome this limitation, in some works, like [107, 113], synthetic groups are generated on well-known dataset like *Movielens*. Unfortunately, being the true preferences of such groups unknown, the generated recommendations are compared with the individual ratings in the test set. Although calculating GRS performance in such way seems questionable, some useful result is obtained. In particular, the recommendations accuracy decreases as the group size increases and, the greater the similarity of group member profiles, the better the accuracy of recommendations.

Some small-scale experiments have been also performed with real users. In [112] participants rated how satisfied they thought group members would be with group recommendations generated according to different strategies. In such experiment, the multiplicative method performed best followed by average, average without misery and most pleasure. In [135], an experiment with real users was conducted to validate the results obtained with synthetic groups. One of the main conclusions of this study was that it is possible to realize trustworthy experiments with synthetic data, as the online user test confirmed the results of the offline experiment.

The preceding works have in common that the group recommendations just take users' individual preferences into account without considering either the user personality or the relationships among group members. Despite that in real contexts such aspects are crucial in the item selection process, systems dealing with them have been introduced only recently. For example, in [119] a 30-questions test is used to determine a value representing how selfish or cooperative an user is in conflict situations. Such value is used, in turn, to weight the preferences of group members during aggregation.

With respect to relationships among group members, in [136] it has been pointed out that people tend to rely more on recommendations coming from people they trust than on anonymous ratings coming from similar users. According to [119], this is even more important when users have to decide on items to be consumed within a group. *FilmTrust* [137] is an example of trust-aware RS, which builds a network of trust among users based on explicit feedback. Users are asked to provide a trust rating for each person they add as a friend. Then, unknown items for each user are rated according to the average rating of trusted friends weighted by the value of trust. Another example is *Epinions*[3], an e-commerce site which maintains a network of trust by asking users to indicate which members they trust or distrust. If no direct connections from an user to any rater exist for a given item, trust propagation and aggregation metrics are used to estimate indirect trust values.

[3] http://www.epinions.com/

A drawback of such personality and trust-based approaches is that they requires explicit feedback from users. To overcome this issue, a promising alternative is to build networks of trust from implicit information commonly shared on-line by users e.g. data contained in social networks. According to [138], the complete transparency of this process compensates the fact that the trust networks obtained in such way are less accurate than those obtained with explicit feedback. For example, in [119], interpersonal trust has been estimated as a combination of 10 factors measured on Facebook profiles and used within a GRS for movie recommendation named *Happy Movie*.

As an evolution of the latter approaches, the novel GRS models proposed in this chapter combine interpersonal trust and personality concepts in that of social influence. This is motivated by the fact that items selection in a group usually follows an argumentation process, where each member defends her preferences and rebuts other's opinions. In this process, interpersonal influence (that is dependent of both trust and personality) is a major factor affecting opinion change toward a common decision. Taking such factor into account allows to define a more accurate representation of the reality, leading to better recommendations.

In particular, to introduce social influence in GRS, we propose a GDM based approach. In fact, while items selection for individual consumption can be considered as an interaction-free process, so manageable with standard RS techniques, when interaction is needed to find an agreement among different hypotheses, a GDM problem can be outlined and specific techniques taking social influence into account can be applied. In addition, given that group recommendations are generated starting from individual predictions made for group members (rather than from explicit preference statements), fuzzy-based approaches, intrinsically able to deal with uncertainty and inaccuracy of such predictions, have been preferred.

The application of GDM techniques to support GRS is a young research area. To the best of our knowledge few works exist in this area and are mainly related to consensus-reaching among group members. For example, in [139] a collaborative approach is used to provide individual recommendations for a group of users and then, an automatic consensus model based on GDM is applied to update the preferences of the most discordant members making them as concordant as possible with average preferences. In [140] the same approach is applied to restaurant recommendations and takes into account geolocation too. With respect to such works, our approach applies a full GDM process that also deals with social aspects of influence, trust and personality.

Chapter 6

Experiments and Evaluation

This chapter presents a set of studies and experiments aimed at measuring the performance of the original peer assessment methods defined in chapter 4 in comparison with other existing methods. In particular, the results of two in-silico studies (made of several experiments with synthetic, realistic data), related to GMPA methods and FOPA, are reported and discussed.

The results of three experiments with real students are also reported, one related to GMPA methods, one to FOPA and one involving both at the same time. Two of these experiments have been made at the University of Salerno and one at the Open University of Catalonia. In two experiments cardinal peer grades have been collected and, when needed, converted in ordinal ones while, in the last experiment, students were asked to directly provide fuzzy rankings as the output of the assessment task. The results of each experiment are discussed in a specific subsection.

6.1 GMPA with Synthetic Data

To evaluate the performance of the GMPA methods defined in section 4.5 and compare them with existing methods, seven different experiments with synthetic data have been performed. In all experiments 100 students are supposed to have submitted a solution to an assignment composed of 10 questions. For each correct answer a student gains 1 point and for each wrong answer she gains 0 points. The real grade of each student is then an integer belonging to the set [0,10].

Each student has then to evaluate the submissions of m other peers. According to [79], we suppose that each student i with a real grade $\overline{g_\imath}$ has probability $\overline{g_\imath}/10$ of marking correctly each answer of a peer submission. So if the student i grades the submission of a student j (whose real grade is $\overline{g_\jmath}$), then the proposed grade g_{ji} is a random variable so that:

$$g_{ji} \sim \mathrm{B}\left(\overline{g_\jmath}, \frac{\overline{g_\imath}}{10}\right) + \mathrm{B}\left(1 - \overline{g_\jmath}, \frac{1 - \overline{g_\imath}}{10}\right) \tag{87}$$

where $\mathrm{B}(m, p)$ is a binomial distribution of m trials with probability p.

Each experiment is made of several iterations. For each iteration, real grades are randomly assigned (with different probability distributions). Then, the *assessment grid* is built (according to different methods) and the grades matrix is randomly filled according to the probability distribution given in equation (87). The *final grades* are then calculated (according to different methods) and compared to real grades by calculating the RMSE as defined in section 4.4. The details and the results of each experiment are discussed in the next sub-sections.

6.1.1 Binomial Distribution of Grades

In the first experiment, real grades are assigned according to a binomial distribution: each student, for each of the 10 questions of her assignment, has a probability p of answering correctly and a probability $1 - p$ of answering wrongly. The real grade of a student i is so assigned according to:

$$\overline{g_\imath} \sim \mathrm{B}(10, p). \tag{88}$$

In each step a probability p is chosen and 1000 iterations are performed. For each iteration, real grades are assigned according to equation (88) with probability p. Then, a 100×100 assessment grid is randomly generated with equation (51) so that each student evaluates 4 other peers ($m = 4$). A grades matrix, including all proposed grades, is then randomly generated from the distribution given in equation (87).

For each iteration, the final grade of each student is calculated in the following ways:

- as the *Average* of grades proposed by peers with equation (52);
- with the *PeerRank* rule described by equation (60);
- with the *F-PeerRank* the rule described by equations (61)-(62) selecting the function $f(x) = x^2$, named *PowPeerRank* hereinafter;
- with the *F-PeerRank* the rule described by equations (61)-(62) selecting the function $f(x) = e^x$, named *ExpPeerRank* hereinafter;
- with the *BestPeer* rule described by equation (63) using *ExpPeerRank* to obtain a first estimation of student grades.

For each iteration, the RMSE between final and real grades is calculated over all students and obtained values are mediated over all iterations. Figure 14 plots the performance obtained applying the five methods to the defined marking model in terms of mean RMSE against the probability p used to generate real grades. It results that *PeerRank* and *ExpPeerRank* outperform *Average* for $p > 0.6$. The performance of all methods is quite similar when $0.5 \leq p \leq 0.6$ while, for $p < 0.5$, the best method remains the *Average*.

Obtained results show that all GMPA methods need $p > 0.5$ to get useful signal out of the data. It is worth noting that $p = 0.5$ means that students are answering (or marking) questions just as well by tossing a coin. So, in real contexts, assuming that $p > 0.5$ is not a restrictive constraint. Moreover, as it can be seen, *PowPeerRank* performs a little better than *PeerRank* while *ExpPeerRank* outperforms both. Instead, *BestPeer* is better than other methods only for $p > 0.9$.

The best choice for this distribution of grades is so *ExpPeerRank* that ensures, in best cases, a decrease in RMSE of about 1 grade with respect to the baseline *Average* method. So, on average, each student will have a final grade closer to the real one of approximately 1 point over 10.

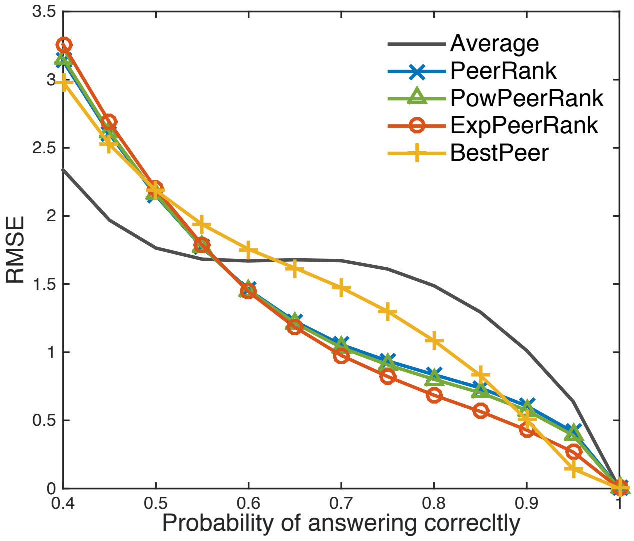

Figure 14. Performances or GMPA methods on a binomial distribution of grades with different values for p (probability of answering correctly).

6.1.2 Uniform Distribution of Grades

In the second experiment, the real grades are assigned according to a uniform distribution i.e. each student receives an integer random grade to the whole assignment from a minimum *min* to a maximum 10 where $0 \leq min \leq 10$. Hence the real grade of a student i is assigned according to:

$$\overline{g_\imath} \sim \mathrm{U}\ (\{min, \dots, 10\}) \tag{89}$$

where $\mathrm{U}(S)$ defines a discrete uniform distribution over the set S.

Figure 15 plots the performance, in terms of mean RMSE against the minimum grade *min*, obtained by applying the same methods of the first experiment to the defined marking model with $m = 4$. Also in this case *ExpPeerRank* outperform the other methods in almost all conditions while *PowPeerRank* is a little more performant than *PeerRank*. Only for $min = 0$ the performance of all methods is quite the same.

It is interesting to note that *BestPeer* behaves better than in the previous experiment, with a RMSE lower or equal to *PeerRank*. The best performance is obtained when $min \leq 5$ (high variance of real grades) and with $min \geq 8$ (high average real grade). This can be explained by the fact that, when there is a high variance in student levels, there is a high probability that a peer is evaluated also by unreliable graders and this affects the quality of the final grade in all methods (at different levels) apart from *BestPeer* where only the best grade is selected. This advantage disappears when *min* increases because in that case, proposed grades increase their average quality.

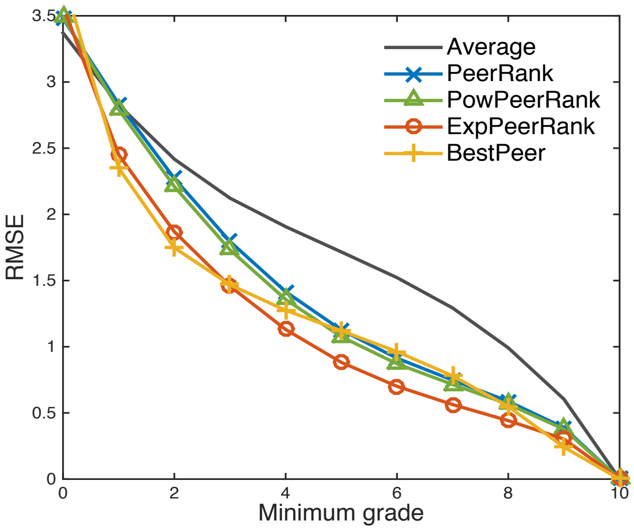

Figure 15. Performance of GMPA methods on a uniform distribution of grades with different values for min (minimum grade).

6.1.3 Binomial Distribution of Grades with Smart Assignment

This experiment replicates the one of section 6.1.1 with the difference that the assessment grid is generated according to equation (64) rather than to equation (51). In the model, we assume that the average grade obtained in previous assessments (needed to generate the student ranking) is equal to the assigned real grade. This is a simplification that supposes that students maintain a constant performance across several assignments. Given that, the results of this experiment can be considered as an upper bound of the results obtainable with smart assignment in real contexts.

Figure 16 plots the performance obtained applying the defined methods to the marking model with random (dashed lines) and smart (plain lines) assignment methods. Given that the performance of *PowPeerRank* is quite similar to that offered by the standard *PeerRank* method, we have removed this method from the figure to maintain an higher readability.

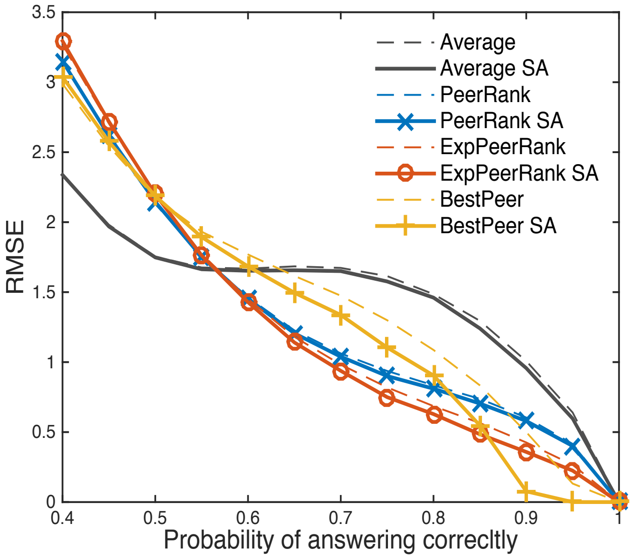

Figure 16. Performance of GMPA methods on a binomial distribution of grades with different assignment methods (SA = Smart Assignment).

As it can be seen, with a binomial distribution of real grades *Average*, *PeerRank* and *ExpPeerRank* are quite insensitive to the smart assignment. Instead, as it might be supposed, *BestPeer* has a substantial improvement because the smart assignment ensures that each student is assessed by at least one good grader whose proposed grade is selected as the final one.

6.1.4 Uniform Distribution of Grades with Smart Assignment

This experiment replicates the one of section 6.1.2 with the difference that the assessment grid is generated according to equation (64) rather than to equation (51), with the same assumptions made in section 6.1.3 with respect to the average grade obtained in previous assessments.

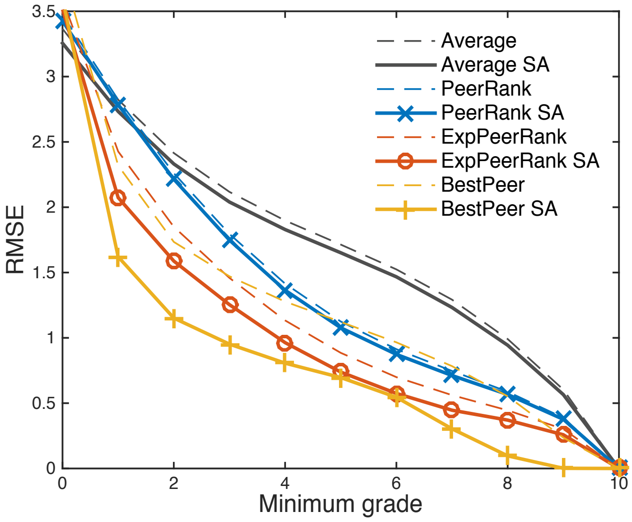

Figure 17. Performance of GMPA methods on a uniform distribution of grades with different assignment methods (SA = Smart Assignment).

Figure 17 plots the performance obtained by applying the four methods (also in this case we exclude *PowPeerRank* whose performance is similar to the standard *PeerRank*) to the defined marking model with random (dashed lines) and smart (plain lines) assignment methods in case of uniform distribution of real grades. In this case, while *Average* and *PeerRank* result again quite insensitive to smart assignment, *ExpPeerRank* and (to a greater extent) *BestPeer*, show a good improvement.

In particular, *BestPeer* outperforms all the other methods, especially in configurations with high grades variance ($min < 5$) and high average real grade ($min > 6$). Only for $min < 1$ its performance is comparable than that of other methods. Hence in this case, the best choice seems to be *BestPeer*, whose performance in contexts that present a high variance of student levels, is boosted by the smart assignment.

6.1.5 Binomial Distribution of Grades and Variable Number of Assessors per Student

The number m of submissions that each student has to evaluate is one of the main parameters that must be defined to setup a peer grading session. On one hand, such number must be kept as small as possible to avoid overloading the students, with the risk that they do not respond adequately to the exercise providing rough, partial or void estimations. On the other hand, this number corresponds to the number of assessors for each submission. Taking this into consideration, m should be kept as big as possible to have sufficient information to estimate the final grades.

To determine how the selection of m impacts on the performance of the defined GMPA methods, we have performed another experiment where the real grades are assigned according to a binomial distribution with probability $p = 0.7$ (a reasonable value in real contexts). In each step, the number m of assessors for each student is chosen from 1 to 12 and 1000 iterations are performed. For each iteration, real grades are assigned, then an assessment grid is generated with smart assignment according to equation (64). A grades matrix, including all proposed grades, is then randomly generated from the distribution given in equation (87).

Figure 18 plots the performance obtained by applying the five methods to the defined marking model in terms of mean RMSE against the number of assessors m. As expected, the error decreases when the number of assessor increases but the decrease is smoother as m increases.

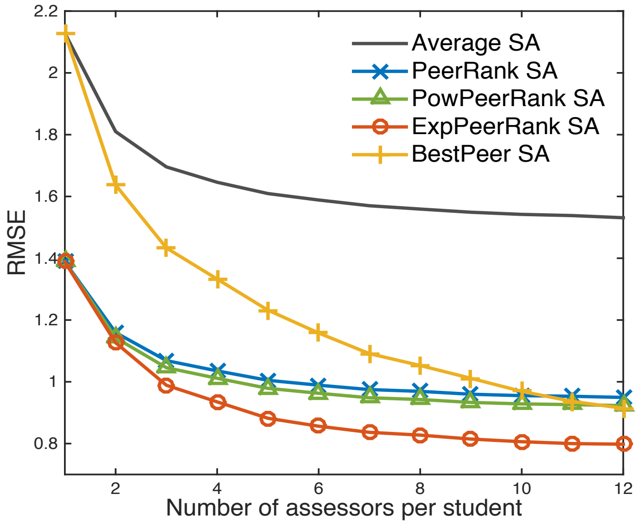

Figure 18. Performance of GMPA methods on a binomial distribution of grades with different number of assessors for student.

With *Average*, *PeerRank* and *PowPeerRank* algorithms, the increase in performance after the 4th assessor is negligible. *ExpPeerRank* offers good improvement until the 6th assessor while *BestPeer* has sensible improvements until the 10th assessor. Moreover, this latter becomes more performant of both *PeerRank* and *PowPeerRank* starting from the 12th assessor.

This fact can be explained by considering that the number of assignments evaluated by the best graders increase when more assessors are added. The impact on rules other than *BestPeer* is limited given that the resulting grades are obtained by considering also grades proposed by other assessors while the most positive impact is on *BestPeer* that only considers the grade assigned by the best grader.

6.1.6 Uniform Distribution of Grades and Variable Number of Assessors per Student

This experiment replicates the previous one but the real grades are assigned according to a uniform distribution and each student receives an integer random grade to the whole assignment from a minimum of 6 to a maximum of 10 i.e. $min = 6$ in equation (89). Figure 19 plots the performance obtained by applying the defined GMPA methods to the defined marking model (with smart assignment) in terms of mean RMSE against the number m of assessors per student.

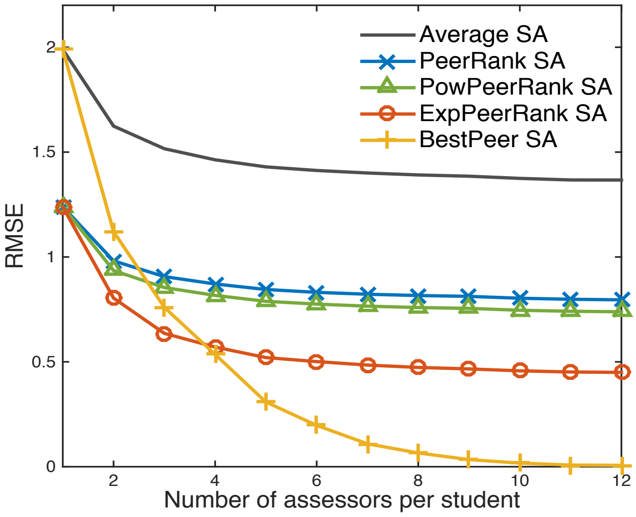

Figure 19. Performance of GMPA methods on a uniform distribution of grades with different number of assessors for student.

As in the previous case, the error decreases when the number of assessors increases and the decrease is smoother as m increases. It should be noted that *BestPeer* outperforms the other methods for $m \geq 4$. Moreover, for *BestPeer*, the RMSE asymptotically goes to 0 when the number of assessors increase. This is due to the same reasons already explained in the previous section and the effect is more evident with uniform distribution of real grades thanks to the high average level of the simulated class that results in a high number of reliable graders.

6.1.7 BestPeer and Support Methods

As described in section 4.5, the *BestPeer* method calculates the final grade for any student with one of the other methods, then assigns to each student the grade coming from the assessor with the best final grade. In the previous experiments *ExpPeerRank* has been used as support method for *BestPeer*. In this last experiment we wonder if *ExpPeerRank* is the best possible choice, at least in the configuration of the experiment made in section 6.1.2.

We have so repeated the same experiment only with *BestPeer*, adopting different support methods. Obtained results are shown in Figure 20 against the standard *Average* method. As it might be supposed, *ExpPeerRank* (that is the method with the best performance in the majority of configurations) represents the best choice.

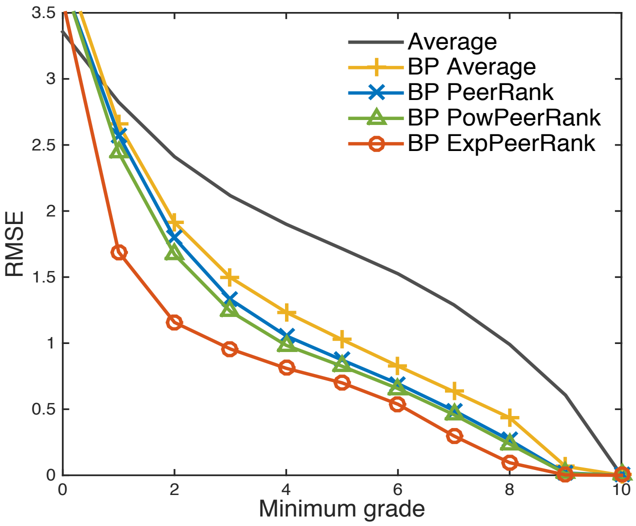

Figure 20. Performance of Best Peer with different support methods on a uniform distribution of grades with different values for m.

6.2 GMPA with Real Data

To evaluate the effectiveness of the GMPA methods defined in section 4.5 also with real users, we have applied them on peer grading data coming from an on-line course held in Spring 2014 at the Open University of Catalonia (UOC) [141]. The on-line course had 58 students enrolled and was divided in 7 subsequent modules. After having completed the study of a module, each student received an invitation to answer three open questions. When the answers were collected, each student had to access each classmates' answers and evaluate it according to a 5-point scale (A, B, C+, C-, D) before starting the subsequent module.

The peer grading core component was developed in Java and integrated in the UOC learning management system. It integrates two external Web applications: *Google Forms*[4] to collect the answers to module questions and *Lime Survey*[5] to let students evaluate peers' answers to module questions. To exchange data between the two tools a *Comma Separated Value* exchange model has been adopted and the *Super CSV*[6] package has been selected to deal with such format in *Java*.

Table 15 shows the statistics collected for each module. As it can be seen, the number of active students per module (students providing answers to module questions) has decreased about 70% over time: from 41 in module 1 to 12 in module 7 (on a total of 58 enrolled students). Despite it may seem discouraging, this result is in line with the problematic drop-out rate suffered by on-line courses (the mean drop-out ratio at UOC is about 50%).

Moreover, only a part of the active students have also executed the peer grading task. The second row of Table 15 reports on the number of students that, for each module, succeeded in evaluating (at least some of) their peers.

[4] https://docs.google.com/forms/

[5] https://www.limesurvey.org/

[6] http://super-csv.github.io/super-csv/

The remaining rows of the table report the mean grade obtained by students for each question of each module normalized between 0 and 10. If we consider that the three questions are graded separately, data for $3 \times 7 = 21$ separate assignments is available.

Modules	1	2	3	4	5	6	7
Active students	41	28	23	20	21	18	12
Peer Assessors	30	24	15	14	16	11	11
Mean grade (question 1)	7.3	8.0	7.5	7.3	7.8	7.5	7.5
Mean grade (question 2)	7.0	7.3	7.5	7.5	7.3	7.5	7.3
Mean grade (question 3)	7.5	7.8	7.3	7.8	7.3	7.8	7.5

Table 15. Main statistics of the performed experiment

In the experiment, students were asked to grade all peers. Conversely, in a MOOC peer grading setting, students would be asked to evaluate only a small subset of other students. In the absence of an assessment made by an expert tutor, this peculiarity allows us to calculate the *approximate real grade* $\overline{g_i}$ of a student i as the mean grade obtained by her over the whole population of assessors. According to [75], we have assumed that the mean of many student grades tend towards the correct real grade, especially for the first two modules where each submission were graded by 30 (for module 1) and 24 (for module 2) peer assessors.

Starting from this data we have then performed two different experiments as detailed in the next subsections. Once the assignment is selected among the 21 available, each experiment is made of several iterations. Given an assignment, for each iteration we have supposed that just m grades were proposed (randomly selected among those available) for each active student. This allow us to simulate the real conditions of a MOOC peer grading task.

So, for each iteration, the assessment grid is built by randomly selecting m assessors for each active student and the grades matrix is filled with grades proposed by that students. Final grades are then calculated (with different GMPA methods) and compared to the approximate real grade (obtained as previously described) by calculating the RMSE.

The purpose of the experiments is to determine which of the defined methods can estimate with better accuracy the approximate real grade (obtained by averaging all available evaluations) using only a small number m of randomly selected evaluations per submission. Considering that the approximate real grade is, in turn, an estimation of the real grade, we are indirectly finding the best estimator of the real grade.

6.2.1 Fixed number of peer assessors

This experiment is made of 7 steps (one for each module) and 21 sub-steps (corresponding to the three questions for each module). For each sub-step, 1000 iterations are performed. In each iteration, 4 assessors are randomly selected for each submission (i. e. $m = 4$) and both the *assessment grid* and the *grades matrix* are filled as previously explained. The dimension of these matrices is equal to the number of active students in the related module (from 41×41 in the first step to 12×12 in the seventh).

For each iteration, the final grade of each student is calculated as the *Average* of grades proposed by peers with equation (52); with the *PeerRank* rule described by equation (60); with the *F-PeerRank* the rule described by equations (61)-(62) selecting the functions $f(x) = x^2$ (*PowPeerRank*) and $f(x) = e^x$ (*ExpPeerRank*); with the *BestPeer* rule described by equation (63) using *ExpPeerRank* to obtain a first estimation of student grades. The RMSE between final and approximate real grades is calculated for each iteration over the active students.

Table 16 summarizes the performance obtained by the defined methods on the experimental data. The reported RMSE values are mediated over all iterations for each sub-step and over all stub-steps for each step. As it can be seen both *PeerRank* and *PowPeerRank* outperform the *Average* method in all conditions. They show a better accuracy in predicting the *approximate real grade* even with a small number of available evaluations for each student. Conversely, *ExpPeerRank* and *BestPeer* performances are worst.

Module	RMSE per method				
	Average	PeerRank	PowPeerRank	ExpPeerRank	BestPeer
1	1.00	0.96	0.94	1.40	2.13
2	0.87	0.82	0.81	1.16	1.87
3	0.88	0.83	0.82	1.13	1.82
4	0.82	0.77	0.77	1.01	1.80
5	0.81	0.76	0.75	1.02	1.74
6	0.80	0.76	0.75	1.07	1.87
7	0.65	0.61	0.61	0.77	1.49
Mean	0.83	0.79	0.78	1.08	1.81

Table 16. Performance obtained on experimental data

This result can be explained by the fact that, with both *ExpPeerRank* and *BestPeer*, the final grade of each student is extremely influenced by the grade proposed by one grader: the most reliable. This moves the final grade away from the *approximate real grade* obtained by mediating all available evaluations. In particular, *BestPeer* suffers from an approximation issue too. Indeed, by just considering the grade proposed by the best grader, the final grade results in an integer from 1 to 5 (a point from the 5-point scale) normalized in the interval [0,10].

It should be noted that, when the total number of active student decreases (as the progressive module number increases), the performance of all methods improves. This behaviour is explained by the fact that the number of evaluations used for prediction is fixed (i.e. $m = 4$) while the total number of evaluations (used to calculate the *approximate real grade*) decreases. Therefore, the ratio of available data over the whole set increases, resulting in better performance.

6.2.2 Variable number of peer assessors

In this experiment the attention is focused just on one assignment (i.e. the first question of the first module) but the number m of assessors for each submission is increased from a minimum of 2 to a maximum of 10. In each step, the number m of assessors for each student is chosen in this range and 1000 iterations are performed. For each iteration the *assessment grid* and the *grades matrix* have been generated as in the previous experiment and the final grades are calculated according to the defined methods.

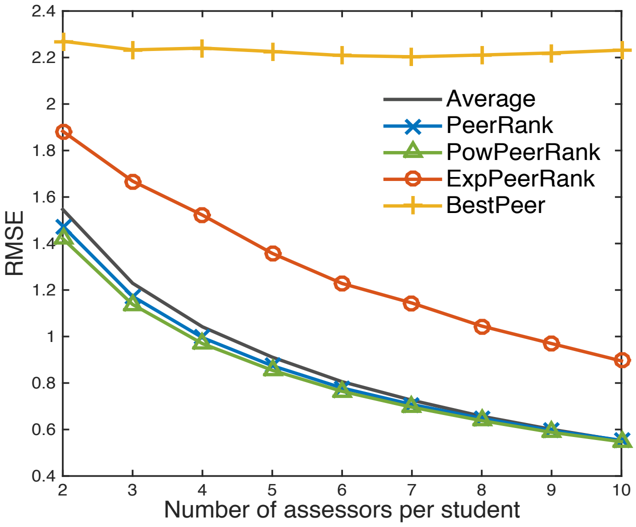

Figure 21. Performance of GMPA methods on experimental data with different number of assessors for student.

Figure 21 shows the performance obtained by the five methods in terms of mean RMSE against the number of assessors m. As in experiments reported in sections 6.1.5 and 0 (executed on synthetic data), the error decreases when the number of assessor increases and the decrease is smoother as m increases. An exception is *BestPeer* that has uniform performance regardless of the selected number of assessors. This can be explained through the same approximation issue pointed out in the preceding sub-section.

As it can be seen, both the *PeerRank* and *PowPeerRank* methods show better performance with respect to the average aggregation rule. Indeed, the performance gap between these methods decreases with the increase of the number of assessors i.e. when the quantity of information available becomes closer to information used to calculate the *approximate real grade*. It should be noted that the comparison against approximated real grades obtained by averaging a large number of peer grades (rather than against teachers' provided grades) obviously advantages the *Average* aggregation rule. Taking this into consideration, the performance achieved by GMPA methods in such experiment can be considered as a lower bound to the performance obtainable in contexts where also teachers' grades are available.

6.3 FOPA with Synthetic Data

To demonstrate the effectiveness of FOPA and to compare it with different approaches, we have performed several experiments with synthetic data. In all the experiments, 100 students are supposed to have submitted a solution to a given assignment. The submission of each student s_i has a real grade $\overline{g_i}$ belonging to [0, 10] assigned according to a normal distribution $\overline{g_i} \sim \mathcal{N}(6,2)$ centered in 6 with a standard deviation of 2.

Each student has then to evaluate the submissions of m peers (with m constant or variable according to the specific experiment) matching a random assessment grid $A = (a_{ij})$ defined as specified by equation (51). Students are imperfect graders so, according to [75], we have modelled such imperfection with two parameters:

- a *bias* term $b \geq 0$ that reflects a tendency of an assessor student to either inflate or deflate her assessment (i.e. high biases describe lenient assessors while low biases describe stringent ones);

- an *unreliability* term $u \geq 0$ that reflects how far, on average, a grader's assessment tends to land with respect to the corresponding true grade (i.e. a low unreliability describes a proper attitude to distinguish between good and bad submissions).

Basing on these two parameters, the perceived grade g_{ij} of a student s_i from the assessor student s_j, is defined according to the following probability distributions:

$$g_{ij} \sim \mathcal{N}(\overline{g_i} + b_j, u) \text{ so that } b_j \sim \mathcal{N}(0, b). \tag{90}$$

The fuzzy ranking R_j is then defined for each assessor student $s_j \in S$ through equations (36)-(37) by setting $\phi(s_i) = g_{ij}$ for each $s_i \in S_j$.

Starting from synthetic data generated in this way, the global ranking and the absolute grades have been estimated for each submission according to the model defined in section 4.6 and compared to real grades (and related rankings). This has allowed us to measure FOPA performances in revealing the ground truth also in presence of noisy data (taking into account different values for bias and reliability) and in comparisons to existing ordinal and cardinal peer assessment methods (described in sections 4). The details and the results of such experiments are discussed in the next sub-sections.

6.3.1 Optimal Parameters Setting

This experiment is aimed at discovering the best settings for the parameters used by FOPA. This is done by measuring the performance obtained in reconstructing the global ranking of submissions both in case of perfect grading (i.e. when students make no errors when assessing other students) that in the more realistic case of imperfect one. The results obtained with different settings are then compared to discover the most promising settings to be used in next experiments.

The first parameter to set is the ranking measure to adopt for quantifying the degree of preference of each submission among those defined in section

1.6 i.e. the one that offers the best performance for the specific problem. Moreover, according to section 1.5, the aggregation of preferences based on OWA can be done starting from several linguistic quantifiers like *much*, *at least half*, *most* and *as many as possible*. Another parameter to set is so the quantifier to apply.

To identify which setting offers the best performances, we have executed the experiment described so far with 100 students and 4 assignments to be evaluated by each (so $m = 4$). When generating perceived grades, we have set $b = 0$ and u ranging from 0 (perfect grading) to 3 (average difference of 3 between the real grades and the perceived ones). For each value assigned to u we have repeated the experiment 1000 times and mediated the obtained results in terms of PCRPR as defined by equation (57). Then, we have repeated the process by setting $u = 0$ and b ranging from 0 (no bias at all) to 3 (average bias of 3).

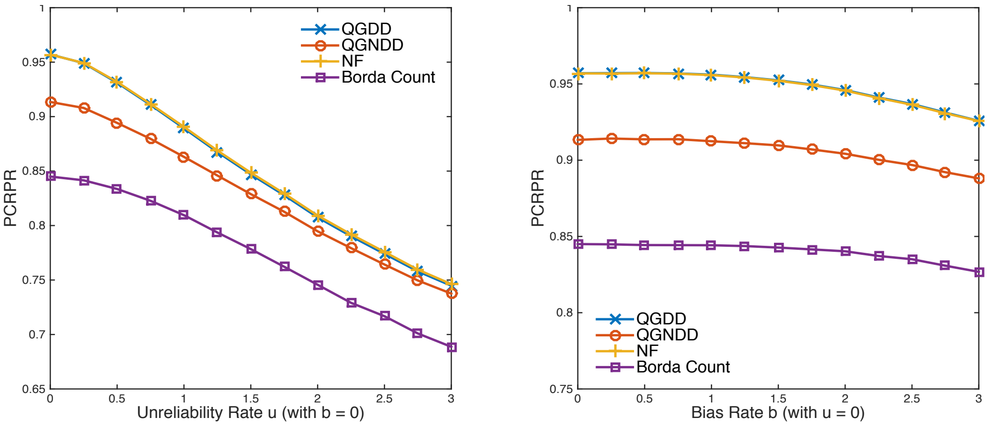

Figure 22. Performances of the QGDD, QGNDD, and NF ranking measures compared with the Borda count in terms of PCRPR.

Figure 22 shows the results in terms of PCRPR, obtained by FOPA, changing the applied ranking measure among *Quantifier Guided Dominance Degree* (QGDD), *Quantifier Guided Non-Dominance Degree* (QGNDD) and *Net Flow* (NF), against the unreliability rate u (on the left) and the bias rate (on the right). The figure shows that, among the available measures, two obtain the best performances with any value of u and b: QGDD and NF. In case of perfect grading (i.e. when $u = b = 0$), they show a PCRPR of 95.7%, that is far beyond the 84.5% obtained by the Borda count.

Both measures demonstrate a fair robustness to unreliability but, the improvement with respect to the Borda count, decreases when u increases. Moreover, it should be noted that all the methods are very robust with respect to the bias with average variations of less than 1% in terms of PCRPC for each increase of 1 grade in bias. Nevertheless, this is a common advantage of ordinal grading methods.

On the other hand, FOPA results to be insensitive with respect to the selection of the OWA quantifier for the aggregation step: the same results are in fact obtained regardless of the adopted one. The same level of insensitivity has been also detected by changing the fuzzy quantifier adopted within the QGDD and QGNDD measures. For this reason, the results obtained changing the quantifier are not shown in the figure.

6.3.2 Comparison with other Ordinal Peer Assessment Methods

This experiment is aimed at comparing the performance of FOPA with that of the other methods for ordinal peer assessment described in section 4.3 in case of perfect and imperfect grading. To do that, we have executed the same experiment described so far with 100 students and 4 assignments to be evaluated by each. When generating perceived grades, we have set $b = 0$ and u ranging from 0 to 3. For each value assigned to u we have repeated the experiment 1000 times and mediated the obtained results in terms of PCRPR, calculated according to equation (57).

Then, for each iteration and experimented method, the obtained scores have been transformed in grades through the equation (66), setting g_{min} and g_{max} equal, respectively, to the minimum and the maximum real grade. Then the RMSE between the grades estimated through each experimented method and the real grades have been calculated according to equation (55).

Figure 23 (on the left) shows the results in terms of PCRPR, obtained by FOPA (adopting the Net Flow aggregation measure) compared with the models of *Mallows* (MAL), *Bradley-Terry* (BT), *Plackett-Luce* (PL) and *Borda*. An additional model named *Score-Weighted Mallows* (MALS) defined in [81] as an improved version of the *Mallows* model has been also tested. The same figure (on the right) plots the results in terms of RMSE of the same models after having transformed the scores in grades as described so far. To experiment the methods described in [81], we have used a software tool named *PeerGrader*[7] made publicly available by the authors.

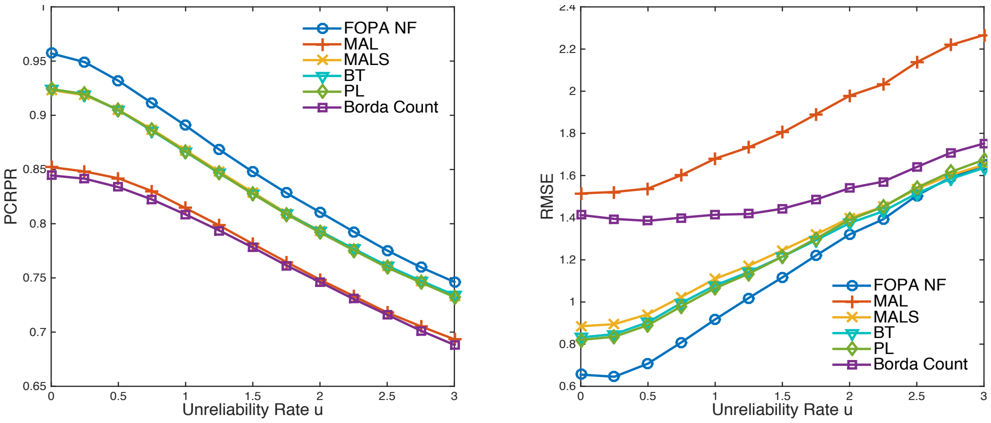

Figure 23. Performances of FOPA against MAL, MALS, BT and PL in terms of PCRPR and RMSE.

Among the introduced methods, MALS, BT and PL show similar PCRPR values while PL performs a little better than the other two in terms of RMSE, at least with $u < 1.5$. The performance of MAL are worst and comparable with those of Borda in terms of PCRPR while, with respect to RMSE, MAL

[7] www.peergrading.org

reaches a higher error rate even with small unreliability rates. Nevertheless, it should be noted that, as explained in [81], MAL (as Borda) is not conceived for obtaining cardinal grades and this is the reason why the authors have improved MAL defining MALS.
The plots show that FOPA outperforms the other methods both in terms of PCRPS that in terms of RMSE. When considering PCRPC, FOPA gains about 4% against MALS, BT and PL in case of perfect grading (from 92.4% to 95.7%) but the improvement decreases when u increases until about 2% for $u = 3$ (from 73.2% to 74.6%). When considering RMSE, FOPA is able to lower the mean error of about 0.2 grades in case of perfect grading (from 0.82 of PL to 0.65 of FOPA) while this difference tends to nullify when increasing the unreliability until $u = 3$.

6.3.3 Comparison with Cardinal Peer Assessment

This experiment is aimed at measuring the performances of FOPA (and some other ordinal approaches) in comparison to cardinal peer assessment where the grade g_{ij} proposed by an assessor student $s_j \in S$ for a student $s_i \in S_j$ is set equal to the perceived grade defined by equation (90) and the final grade of each student is obtained by averaging all the grades obtained by peers according to equation (52).

To compare FOPA and CPA we have executed the same experiment described so far with 100 students and 4 assignments to be evaluated by each. When generating perceived grades, we have considered both b and u ranging from 0 to 3. For each setting, we have repeated the experiment 1000 times and mediated the obtained results in terms of RMSE, calculated according to equation (55).

Figure 24 shows the results in terms of RMSE, obtained by FOPA (with *Net Flow*), by the *Plackett-Luce* method (PL), by *Borda* and by *Cardinal Peer Assessment* (CPA) while ranging the bias rate from 0 to 3. The plot on the left considers that assessor students are perfectly reliable ($u = 0$) while the plot on the right considers a moderate level of unreliability ($u = 1$). As it can be seen, CPA is very sensitive to the bias rate compared with ordinal approaches. In both cases CPA introduces a lower error with respect to FOPA until the bias rate reaches a given threshold, variable according to the unreliability rate (about 1.4 for $u = 0$, 1.7 for $u = 1$). After the threshold, the gap in term of RMSE between CPA and FOPA increases until a difference of about 0.60 for $u = 0$ and $b = 3$ and about 0.43 for $u = 1$ and $b = 3$. It is worth noting that, in all cases, FOPA outperforms the other ordinal methods.

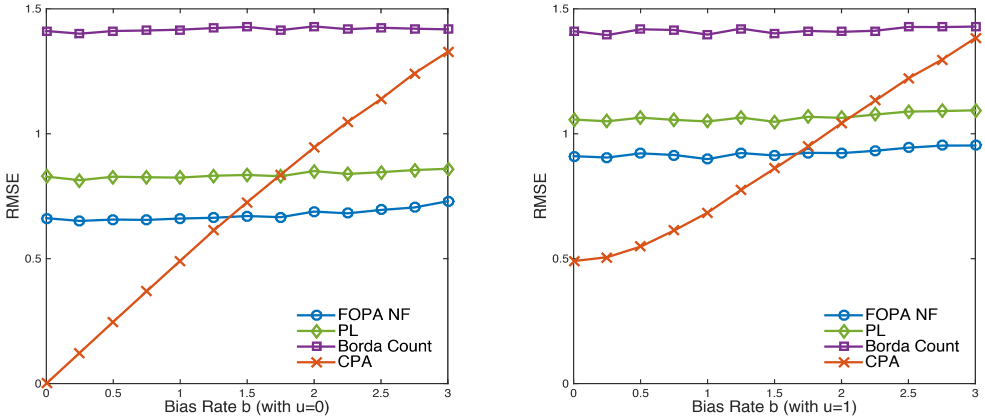

Figure 24. Performances of FOPA, PL and Borda against CPA in terms of RMSE (lower is better) when u=0 (left) and u=1 (right)

To provide a comprehensive view of the behavior of FOPA and CPA, Figure 25 shows the three-dimensional surfaces of the RMSE curves obtained ranging u and b from 0 to 3. Clearly the error level in FOPA mainly depends on the unreliability rate, while the error in CPA quite evenly depends on the unreliability and the bias rates. With medium-low bias and medium-high unreliability, CPA is a little better than FOPA. Conversely, with medium-high bias and medium-low unreliability, FOPA is quite better than CPA.

It is worth noting that CPA requires, by each assessor student, an amount of information significantly higher with respect to ordinal approaches. Given this complexity, as shown in section 4.2, in real contexts cardinal feedback is less reliable with respect to the ordinal one, even when assessors are at the same level of knowledge and experience. In light of this, the performed experiment ultimately benefits CPA because it assumes, for each iteration, the same level of bias and unreliability between cardinal and ordinal feedback. Nevertheless, the performances obtained by FOPA are comparable and in some cases better than those obtained by CPA.

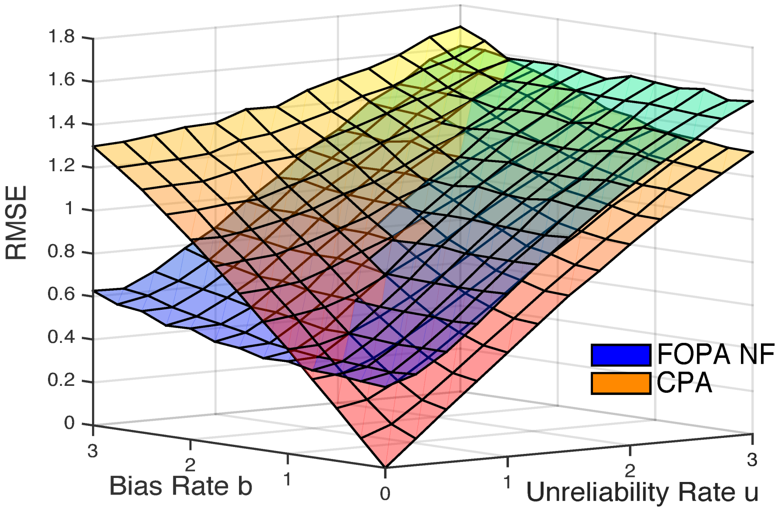

Figure 25. Performances of FOPA and CPA in terms of RMSE (lower is better) ranging both the bias and unreliability rates

6.3.4 Selection of the Number of Assessors

The number m of submissions that each student has to evaluate is one of the main parameters that must be defined to setup a peer assessment session. On one hand, this number should be kept as small as possible to avoid overloading the students, with the risk that they do not respond adequately to the exercise providing rough, partial or void estimations. On the other hand, according to the definition of assessment grid provided in section 4.3, this number corresponds to the number of assessors for each submission. In this respect, m should be kept as big as possible to have sufficient information to estimate the final ranking and grades.

To determine how the selection of m impacts on the performance of FOPA, we have executed the same experiment described so far with 20 and 200 students and a number of assignments to be evaluated by each student variable from 2 to 20. When generating perceived grades, we have set $b = 0$ (the previous experiments have shown that FOPA is insensitive to the bias) and u variable from 0 (perfect grading) to 3. For each setting we have repeated the experiment 1000 times and mediated the obtained results in terms of RMSE, calculated according to equation (55).

Figure 26 (left) plots the results obtained by FOPA (with *Net Flow*) with 20 students and m ranging from 2 to 20. A first thing to observe is that, while for high unreliability rates ($u \geq 2$) an increase of m always determines a decrease of the whole error level, for low unreliability rates ($u < 2$) an increase of m determines a decrease of the RMSE only until a given threshold. After the threshold, adding more assessors, results in an increase in the RMSE. This can be explained by the fact that, while using ranking strings for assessing the submissions, a noise is introduced in the model (in fact, ranking strings can be seen as approximated FPRs). Such noise increases when the strings length increases (so when m increases) but it is balanced by the additional information obtained with more assessors.

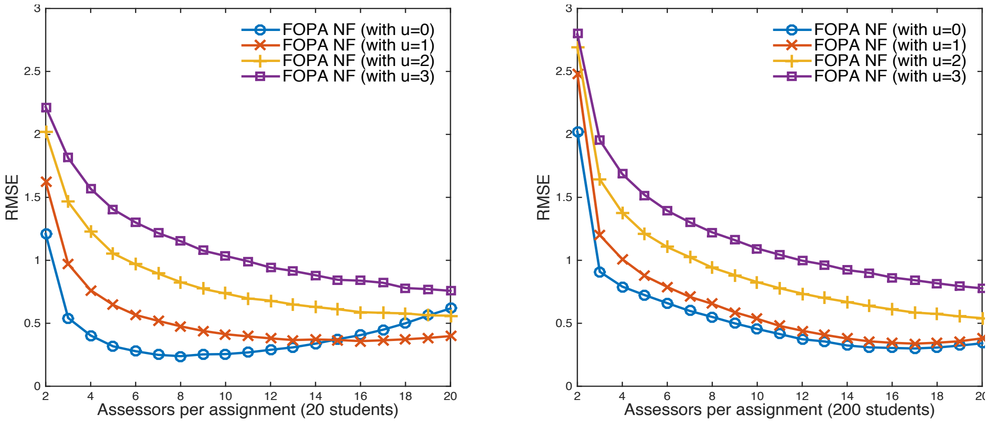

Figure 26. Performances of FOPA in terms of RMSE with different values for u, ranging m from 2 to 20, with n=20 (left) and n=200 (right)

In the (unrealistic) case of perfect grading (when $u = 0$), all assessors have exactly the same perception of the student grades so, after a given threshold, adding more assessors does not increase the quantity of available information until the extreme case of $m = n$, when all the assessor students provide exactly the same ranking string. So in these cases the noise introduced by ranking string approximation remains unbalanced and the error increases. This is evenly true in settings with low unreliability rates ($u < 2$) and with more students to evaluate (Figure 26, right) even if the threshold becomes higher and higher.

With respect to the selection of m, it should be noted that, apart the unrealistic case where $u = 0$, the curves plotted on the left and on the right side of Figure 26 have a similar trend. Regardless of the number of students and of the unreliability rate u, we notice a steep decrease of the RMSE while moving from two to three assessors and a smoother decrease for subsequent values of m. By looking at the right part of the figures we see that, when $u = 1$, the RMSE start to increase for $m > 16$ while, even for $u > 1$, the decrease in RMSE obtained adding a new assessor is less than 0.02. Such reflections suggest to select a number m of submissions to be assessed per student so that $3 \leq m \leq 16$ regardless of the total number of students involved and on the expected degree of unreliability and bias.

6.4 FOPA with Real Data

To evaluate the performance of FOPA and other peer assessment methods discussed in sections 4.2 and 4.6 in another context, we have experimented them within a course on *Computer Skills for Education* of a M.S. degree in Pedagogical Sciences at the University of Salerno. The experiment was aimed at measuring at what extent each model is able to estimate the grade assigned by the teacher to every student based on imprecise ordinal feedback provided by students themselves. In the next subsections, we describe the experimental setting and, then, we illustrate and analyse the collected data.

6.4.1 Experimental Setting

The experimental set was composed by first year students taking part in a 20 hours course on *Computer Skills for Education* aimed at developing basic competencies on computer architectures, computational thinking and coding. The course, that is part of a 5-year M.S. degree in Pedagogical Sciences, was held through traditional face-to-face lectures and exercises sessions.

The formative evaluation experiment was performed in two sessions, held in two different days of the same week, with 25 voluntary students. In the first session students have been asked to complete and submit a coding exercise while in the second session students have been asked to assess the submissions coming from a subset of their peers by providing a fuzzy ranking.

The peer grading task was performed in a blind mode in order that students do not know whom they are assessing. The same submissions have been also assessed by the course teacher to build the ground truth with which to compare the results coming from experimented peer assessment models.

6.4.2 Data Collection

A total of 11 students over 25 completed the first session by submitting a solution to the proposed exercise while the remaining 14 were not able to complete the task. For this reason, during the second session students were divided in two groups: the first including those that submitted their solution and the second including the remaining ones. Students of the first group (being considered more proficient) were asked to evaluate 5 submissions (over the 11 available) while students of the second group were asked to only evaluate 3 submissions.

To assign the submissions to assessors, two random assessment grids have been generated: the first 11×11 grid involved students from the first group both as assessors and as assessees while the second 11×14 grid involved students from the first group as assessees and students from the second group as assessors. In both cases, equation (51) was applied.

Student	Assessees	Fuzzy Rankings	True Grade (0-30)
s_1	$\{s_2, s_4, s_7, s_9, s_{11}\}$	$s_4 \geq s_{11} \geq s_9 \approx s_7 \approx s_2$	18
s_2	$\{s_3, s_5, s_6, s_8, s_{10}\}$	$s_3 \geq s_{10} \approx s_5 \gg s_8 \approx s_6$	10
s_3	$\{s_1, s_4, s_6, s_9, s_{11}\}$	$s_4 \gg s_{11} \geq s_9 \geq s_1 \geq s_6$	24
s_4	$\{s_1, s_3, s_5, s_8, s_{10}\}$	$s_{10} \geq s_3 > s_5 > s_1 > s_8$	30
s_5	$\{s_1, s_3, s_6, s_8, s_{11}\}$	$s_3 \gg s_{11} > s_8 > s_1 \gg s_6$	13
s_6	$\{s_2, s_4, s_7, s_9, s_{11}\}$	–	18
s_7	$\{s_1, s_2, s_4, s_6, s_9\}$	$s_4 \gg s_9 > s_1 > s_6 \geq s_2$	10
s_8	$\{s_2, s_5, s_6, s_7, s_{10}\}$	$s_{10} \geq s_5 > s_2 \geq s_7 \approx s_6$	11
s_9	$\{s_3, s_5, s_7, s_8, s_{10}\}$	$s_3 \gg s_{10} > s_8 \geq s_5 \approx s_7$	18
s_{10}	$\{s_2, s_4, s_7, s_9, s_{11}\}$	$s_4 \gg s_{11} > s_9 > s_7 \geq s_2$	28
s_{11}	$\{s_1, s_3, s_5, s_8, s_{10}\}$	$s_3 > s_{10} \gg s_8 \approx s_1 \approx s_5$	26
s_{12}	$\{s_4, s_9, s_{11}\}$	$s_4 \gg s_9 \geq s_{11}$	–
s_{13}	$\{s_4, s_5, s_{10}\}$	$s_4 \gg s_5 \approx s_{10}$	–
s_{14}	$\{s_1, s_5, s_{11}\}$	–	–
s_{15}	$\{s_2, s_6, s_7\}$	$s_7 \gg s_2 \approx s_6$	–
s_{16}	$\{s_1, s_3, s_8\}$	–	–
s_{17}	$\{s_2, s_7, s_{11}\}$	$s_{11} \gg s_7 > s_2$	–
s_{18}	$\{s_2, s_5, s_{10}\}$	$s_{10} \gg s_2 \geq s_5$	–
s_{19}	$\{s_4, s_6, s_9\}$	$s_4 \gg s_9 \geq s_6$	–
s_{20}	$\{s_3, s_8, s_{10}\}$	–	–
s_{21}	$\{s_4, s_8, s_9\}$	–	–
s_{22}	$\{s_3, s_5, s_{10}\}$	–	–
s_{23}	$\{s_2, s_7, s_{11}\}$	$s_{11} \geq s_2 > s_7$	–
s_{24}	$\{s_3, s_6, s_8\}$	–	–
s_{25}	$\{s_1, s_6, s_9\}$	–	–

Table 17. Students' proposed fuzzy rankings and teacher's assigned grades

Only 17 students over 25 completed the second session by providing a fuzzy ranking: 10 coming from the first group and 7 coming from the second one. All provided fuzzy rankings were complete i.e. all assigned submissions were covered by them. The 11 submissions were also evaluated by the teacher in the range [0,30]. The provided fuzzy rankings as well as teacher assigned grades (true grades) are summarized in Table 17.

6.4.3 Evaluating Peer Assessment Models

We have applied FOPA as well as the other ordinal peer assessment models described in section 4.2 on collected data to demonstrate the effectiveness of ordinal peer assessment in the estimation of student grades and to compare the results obtained by each model with respect to teacher assigned grades.

The Table 18 shows, for each student, the true grade, the grade estimated by FOPA, those estimated by the models of *Mallow* (MAL), *Score-Weighted Mallows* (MALS), *Bradley-Terry* (BT) and *Plackett-Luce* (PL) as defined in [81], and the grade obtained using the Borda count defined by equation (54). Equation (66) is used to obtain cardinal grades from the scores associated to each submission.

The performance of each model is measured both in terms of *Correctly Recovered Pairwise Relations* (PCRPR) and *Root Mean Square Error* (RMSE). With respect to PCRPR, as it can be seen in Table 18, all models rank the submissions in the same order reaching a 90% of similarity to the ranking made by considering teacher assigned grades. With respect to RMSE, the models behaviour ranges from a minimum error of 2.4, obtained by FOPA, to a maximum error of 2.9, obtained by Borda.

According to such results, we can assert that ordinal peer assessment is a valuable approach to support formative evaluation and is capable of estimating quite accurately teacher assigned grades, at least in the considered sample. Only small differences can be appreciated with respect to the selected model. In particular, FOPA presents the minimum error but it slightly increases the mean grade of the class with respect to teacher assigned grades.

Instead, PL shows a slightly greater error rate but it maintains a greater fidelity with respect to the mean grade.

Student	True Grade	FOPA	MAL	MALS	BT	PL	Borda
s_1	18.0	15.7	16.0	14.6	14.7	14.3	12.0
s_2	10.0	9.8	9.0	11.1	11.5	10.8	14.0
s_3	24.0	28.0	27.7	26.9	27.1	26.6	25.0
s_4	30.0	30.0	30.0	30.0	30.0	30.0	30.0
s_5	13.0	15.6	18.3	15.9	16.2	15.1	17.0
s_6	18.0	9.0	9.0	9.0	9.0	9.0	9.0
s_7	10.0	11.0	11.3	11.9	12.2	11.7	14.0
s_8	11.0	14.1	13.7	14.9	15.1	14.1	13.0
s_9	18.0	19.6	20.7	19.7	20.3	19.9	18.0
s_{10}	28.0	24.1	25.3	23.6	24.4	23.7	27.0
s_{11}	26.0	23.5	23.0	22.0	22.5	22.1	24.0
Mean	17.9	18.2	18.5	18.2	18.5	17.9	18.5
PCRPR		0.9	0.9	0.9	0.9	0.9	0.9
RMSE		2.4	2.7	2.8	2.8	2.6	2.9

Table 18. True grades and grades obtained with peer assessment methods

6.4.4 Additional Experiments

It should be noted that, while FOPA is able to fully interpret collected fuzzy rankings, the other models need to translate them into ordinal rankings before use. In particular, while Borda just interprets the $>$ symbol, MAL, MALS, BT and PT can also interpret the $\approx$ symbol (i.e. they admit ties). The symbols $\geq$ and $\gg$ within fuzzy rankings are so translated in the symbol $>$ before using them with methods different from FOPA. The $\approx$ symbol is also removed with Borda and an artificial random order is introduced between the adjacent symbols.

Given this difference, an additional experiment has been performed to investigate the behavior of FOPA when put under the same conditions of the other methods i.e. when using modified fuzzy rankings rather than the original ones. In such conditions, FOPA ended up with a 2.7 RMSE (with 0.9 PCRPR) so 0.3 points are lost with respect to the preceding settings. So, we can conclude that the contribution of fuzzy symbols is remarkable but not decisive in the estimation of teacher assigned grades.

Two additional experiments have been performed to evaluate how the models under examination perform with a reduced set of ranking strings. As said, students have been assigned to two groups, a first group including "more proficient" students and a second group made of "less proficient" ones.

The rows 1-3 of Table 19 show the results obtained by all peer assessment models by considering only fuzzy rankings coming from the group of "more proficient" students. With a lower amount of data available, all the models result in slightly higher error rates, while keeping the adherence to the teacher ranking almost unaltered. The consideration that can be drawn is that adding evaluations improve the peer grading process even in case of dubious reliability of the new evaluations.

Group	Measure	FOPA	MAL	MALS	BT	PL	Borda
1	Mean	19.1	17.6	18.7	18.9	18.7	17.9
	PCRPR	0.9	0.9	0.9	0.9	0.9	0.8
	RMSE	2.9	3.0	3.0	3.0	2.9	3.6
2	Mean	16.2	18.3	17.3	17.6	17.6	17.9
	PCRPR	0.8	0.6	0.7	0.8	0.8	0.6
	RMSE	4.7	7.8	4.8	4.7	4.8	8.7

Table 19. Performance considering a subset of available fuzzy rankings

The rows 4-6 of Table 19 show the results obtained by considering only fuzzy rankings coming from the group of "less proficient" students. As it can be seen, basing on a lower amount of data that, in addition, is of a worst quality, all models result in significantly higher error rates. In particular, Borda and MAL show the higher increase in RMSE (+5.8 for Borda, +5.1

for MAL) while BT shows the lowest one (+1.9). The adherence to the teacher's ranking also lowers drastically with values ranging from 60% to 80%. Nevertheless, also in this case FOPA shows the best performance.

6.5 FOPA vs. GMPA with Real Data

Formative assessment is a teaching method where *evidence about student achievement is elicited, interpreted, and used by teachers, learners, or their peers, to make decisions about the next steps in instruction that are likely to be better, or better founded, than the decisions they would have taken in the absence of the evidence* [142]. An important function of formative assessment is providing students with *continuous feedback*, meaning that opportunities for feedback should occur continuously, but not intrusively, as a part of instruction [143].

In this experiment we evaluate the capability of both FOPA and GMPA peer assessment models defined in sections 4.5 and 4.6 to support formative assessment within a University course on Linear Algebra. In particular, the experiment was aimed at answering the following questions:

1. at what extent peer assessment methods are valuable tools to support formative assessment?
2. at what extent peer assessment methods are also capable of improving students' learning outcomes?

In the next subsections, we describe the experiment setting, report about collected data and analyse such data with the aim of providing an answer to the experimental questions here reported.

6.5.1 Experimental Setting

The experimental set was composed by first year students taking part in a six-monthly intensive module of mathematics within a 3-year B.Sc. degree in Computer Engineering at the University of Salerno. In particular, the focus was on the second module, which concerned linear algebra topics.

The module was made of eight hours per week in face-to-face traditional lectures/exercises sessions, supported by an on-line learning system based on *Moodle*[8] which provided the students with additional learning resources and communication tools. The experiment was held with voluntary students. In particular 43 students over about 200 decided to participate.

The peer assessment exercise was implemented through the *workshop* component of Moodle allowing students to submit and evaluate each other's submissions according to a teacher's assignment. The workflow implemented by the workshop component consisted of the following phases (summarized in Figure 27):

- *planning:* the teacher decides the grading strategy and the assignment allocation method (in case of multiple assignments);
- *setup*: the teacher creates the assessment forms and specifies instructions and configures settings;
- *submission*: the students submit their own work and submissions are allocated to assessor students;
- *assessment*: the students review each other's work according to the criteria established by the teacher;
- *grading evaluation*: student grades are calculated by mediating grades obtained by peers according to equation (52);
- *closing*: the students can see their final grades, the single grades obtained by peers and the related feedback.

6.5.2 Data Collection

A set of 43 students participated in the experiment providing a submission for the 4 questions making up the assignment. Then, 3 submissions to be graded were assigned to each student through a random assessment grid filled according to equation (51). The exercise was performed in a blind mode in order that students did not know whom they were assessing.

[8] https://moodle.org/

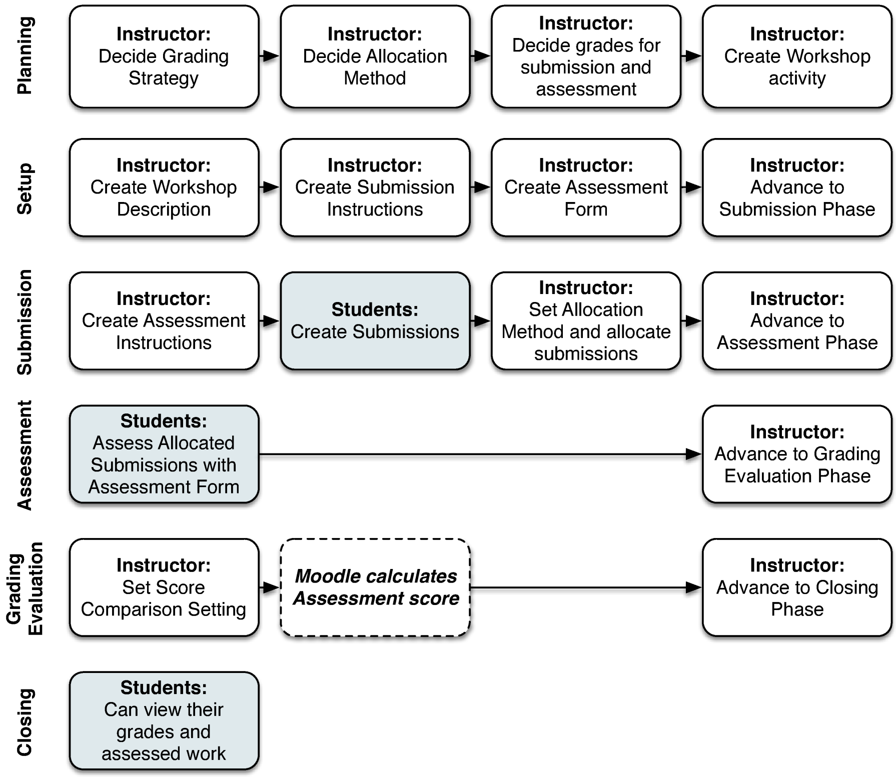
Planning
Instructor: Decide Grading Strategy
Instructor: Decide Allocation Method
Instructor: Decide grades for submission and assessment
Instructor: Create Workshop activity
Setup
Instructor: Create Workshop Description
Instructor: Create Submission Instructions
Instructor: Create Assessment Form
Instructor: Advance to Submission Phase
Submission
Instructor: Create Assessment Instructions
Students: Create Submissions
Instructor: Set Allocation Method and allocate submissions
Instructor: Advance to Assessment Phase
Assessment
Students: Assess Allocated Submissions with Assessment Form
Instructor: Advance to Grading Evaluation Phase
Grading Evaluation
Instructor: Set Score Comparison Setting
Moodle calculates Assessment score
Instructor: Advance to Closing Phase
Closing
Students: Can view their grades and assessed work

Figure 27. The workflow implemented by Moodle workshop component adopted by this experiment.

A subset of 26 students over 43 decided to also take part to the (optional) assessment step. Among them, 24 students completed such task for all the 4 questions of the 3 assigned submissions while 2 students provided only partial marks. This resulted in a total of 304 assigned grades (ranging from 0 to 10) with an average of 1.8 grades per question.

When using peer assigned grades with cardinal models defined in section 4, we had to take into account that, while some students received all expected evaluations, 3 students did not receive any evaluation at all. This singularity had little impact on the *Average* method, provided that m in equation (52) is settled to the number of available votes for each submission rather than to the number of expected ones.

Conversely, the impact on GMPA methods described in section 4.5 is higher. Such methods, in fact, weight the grades provided by each assessor by her own grade. So, grades provided by ungraded students have no value at all. This impacts recursively on the grades of the assessed students and on those of the students assessed by them. To avoid this problem, we have assigned dummy grades to ungraded students and used them throughout the algorithm iterations. Dummy grades, initially set to the average grade of the class, have been removed after all class grades have been calculated.

To use ordinal methods like *Borda* (described in section 4.3) and *FOPA* (described in section 4.6) on students' cardinal input we have defined the fuzzy ranking R_j for each assessor student $s_j \in S$ through equations (36)-(37) by setting $\phi(s_i) = g_{ij}$ for each $s_i \in S_j$ (where g_{ij} corresponds to the grade assigned by s_j to s_i).

To evaluate the effectiveness of peer grading as formative assessment tool, we have also asked the teacher to provide her grades for all the available submissions. Teacher grades was collected separately and did not affected the peer grading process.

6.5.3 Performance on Formative Assessment

To evaluate the effectiveness of peer assessment as a formative assessment tool, we have applied the methods described in sections 4.2 and 4.6 to the data collected and adapted as explained in section 6.5.2 and have compared the obtained final grades to those calculated by *Moodle* (adopting a standard *Average* rule) as well as to those assigned by the teacher.

Table 20 compares the results obtained with the standard *Average* (AVG) rule described by equation (52), with the *PeerRank* rule (PR) described by equation (60), with the *F-PeerRank* the rule described by equations (61)-(62) selecting the functions $f(x) = x^2$ (*PowPeerRank* reported as PPR) and with $f(x) = e^x$ (*ExpPeerRank* reported as EPR), with the *BestPeer* rule (BP) described by equation (63) using *PowPeerRank* to obtain a first estimation of student grades, with the *Borda* count defined by equation (54) and with FOPA as described in section 4.6. Performances have been measured in terms of RMSE, through equation (55), between the grades estimated with each method and the grades assigned by the teacher.

Question	RMSE per Method						
	AVG	PR	PPR	EPR	BP	Borda	FOPA
1	3.73	3.65	3.65	3.64	3.73	4.20	3.70
2	4.54	4.04	4.04	4.00	4.52	3.92	4.14
3	4.04	3.22	3.22	3.18	4.03	3.27	2.95
4	4.19	3.80	3.84	3.71	4.00	3.95	3.92
Mean	4.12	3.68	3.69	3.63	4.07	3.84	3.68

Table 20. Performance obtained on experimental data

The first thing that can be noted is that grades coming from students are very unreliable if compared with grades assigned by the teacher. This may be due to the fact that the data comes from the first experience of the class with a peer-grading exercise and it has been performed at the very beginning of the course. Moreover, only about the 60% of all students have participated in the assessment step resulting in a lack of data for the aggregation step.

The positive thing is that any of the proposed alternative methods reach a lower RMSE with respect to the baseline *Average* method provided by *Moodle*. In particular the *ExpPeerRank* rule outperforms the other methods on average and in almost all the single questions. It is also notable that *FOPA* reaches similar results by relying only on a subset of the information used by *ExpPeerRank* (just the obtained fuzzy ranking of submissions is used rather than the assigned ordinal grades).

6.5.4 Performance Injecting Teachers' Grades

Given the small participation rate, two additional analyses was performed on collected data to evaluate the behaviour of grading methods when the amount of available information increases. Given the availability of teacher's grades for all submissions, we have measured how the performance of all the methods changes by considering, in addition to grades coming from assessor students, an increasingly large subset of grades coming from the teacher.

Both analyses were made in 43 steps (one for each submission). At each step, 4 additional grades coming from the teacher were considered, one for each question of a new submission (the priority was given to submissions with the fewer amount of available evaluations).

In the first analysis, the teacher was considered as a common student evaluating some of the available submissions. For each question, a new column filled of 0 has been so added to both the assessment grid and the grades matrix. At each step an element i of this row was turned to 1 in the assessment grid and the corresponding element of the grades matrix was set as the grade assigned by the teacher to the i-th submission.

An additional row was also added to both matrices to set dummy grades assigned by other students to the teacher (used by *PeerRank*, *PowPeerRank*, *ExpPeerRank* and *BestPeer* methods). In particular, the new row has been filled of 1 (apart for the last element, set to 0) in the assessment grid and filled of 10 (apart the last element, set to 0) in the grades matrix. The teacher is so considered as graded 10 by all other students.

Figure 28 shows how the RMSE of the proposed methods changes while adding new grades from the teacher. As it can be seen, *BestPeer* and *FOPA* obtain the best performance while *ExpPeerRank* shows an error which is always below than that made by the *Average* method. The *PeerRank* rule is better than the *Average* one until 17 added grades, then it results to be a bit worse. *Borda* is quite better than *Average* until 11 added grades, then it becomes quite worse.

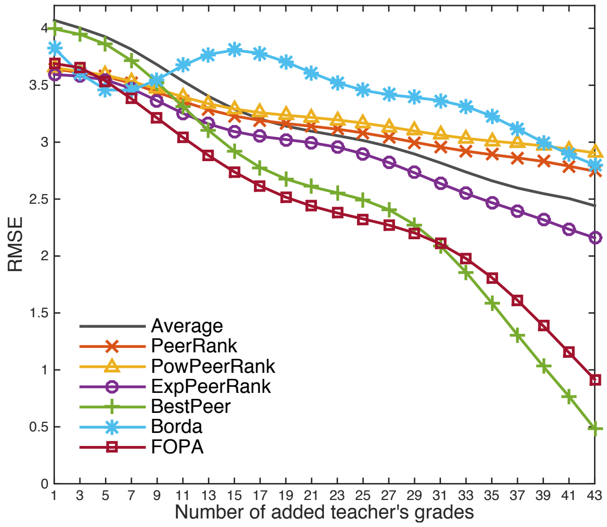

Figure 28. Performance considering increasingly large subsets of grades coming from the teacher (case 1).

Although *BestPeer* and *FOPA* seem to show a similar behaviour, it should be noted that the performance of *BestPeer* is boosted by the dummy grade of 10 assigned to the teacher. Given that it returns the grade assigned by the best grader, in almost all cases, when available, it returns the grade assigned by the teacher. Instead *FOPA* makes no assumption on the grades obtained by graders so it can be considered as the most reliable rule among those experimented.

It should be also noted that the results of *Borda* are quite penalised by the fact that, to uniform scores, they have been normalized by the total number of assessment made by each assessor. So, while the number of teacher's grades increases, their weight with respect to the other decreases.

The second analysis is similar to the first one, except that the teacher is considered as a "super" student, whose grades, if available, are preferred over the grades provided by common students. In fact, while the first analysis is aimed at determining how the described methods behave with additional available grades, the second one is aimed at determining if they can reach even better performances by asking to the teacher to fill the gaps in the data.

Figure 29 shows how the RMSE of the proposed methods changes while adding new grades from the teacher. Also in this case *BestPeer* and *FOPA* show the best performances: *FOPA* is better until 33 added grades, then *BestPeer* wins. In this case, the differences among methods remains almost constant while in the previous case they increase with the number of available grades. Also in this case, the results of *Borda* are penalised for the same reasons explained above.

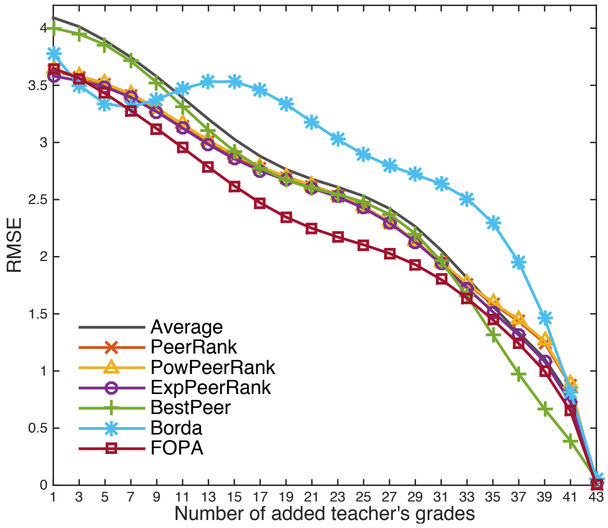

Figure 29. Performance considering increasingly large subsets of grades coming from the teacher (case 2).

Based on experimental data we can affirm that it is possible to improve the results of peer grading through the application of alternative methods with respect to the standard *Average* rule. In particular, *FOPA* is the method that is able to provide the best results with less information (a ranking is needed rather than ordinal grades). Moreover, the results of *FOPA* improve more than the others with increasing amount of information available.

When only few unreliable evaluations are available (as in the analysed case) the use of peer grading as a formative assessment tool is questionable. The results obtained, even when corrective algorithms are used, are quite far from grades assigned by the teacher. Nevertheless, as seen, such results can be improved by asking the teacher to fill the gaps in the data.

6.5.5 Qualitative Evaluation

To evaluate the effectiveness of peer assessment as a tool for improving the learning outcomes, we have had an open interview with the tutor that has oversaw the online activities of the students. She sees *peer assessment as a good strategy for filling knowledge gaps through a different perspective* and suggests its application also on the subsequent topics of the course. Formative assessment is in fact a process observable over a long period of time and the proposed methodology is capable of catching information over time.

The involved tutor also thinks that the method enables to review learnt topics in a collaborative way. In fact, peer grading sees an involvements of students both as assessors of their own learning and as resources to other students. One of the key components of engaging students in the assessment of their own learning is providing them with descriptive feedback as they learn. Descriptive feedback provides students with an understanding of what they are doing well, links to classroom learning, and gives specific input on how to reach the next step in the learning progression.

Apart from formative assessment, peer assessment has resulted capable of developing students' *argumentation skills* where argumentation is defined as the intentional explication of the reasoning used during the development of a given task [144]. In fact, it encourages students to clarify, review and edit their ideas, through the focus of peer feedback. At the same time, it requires students to provide either feedback and grades to their peers based on the criteria of excellence they perceive.

Final Remarks

In this Ph.D. thesis, several fuzzy models for GDM, aimed at improving both preferences expression and aggregation, have been defined and validated in two applicative contexts: e-Learning and Recommender Systems. First of all, a preference model named *Fuzzy Ranking*, combining the user-friendliness of ordinal ranking with the expressive capability of FPR, has been defined. Like FPRs, fuzzy rankings allow decision makers to focus on two alternatives at a time. Differently from FPRs it is not needed to assess preference degrees for any pair of alternatives (resulting in n^2 comparisons with n alternatives) but just for adjacent alternatives in the defined ranking (resulting in $n-1$ comparisons).

Fuzzy rankings offer a compact notation that does not oblige experts to be unnecessarily precise in preference definition. In this way it is very unlikely to introduce inconsistencies in the GDM process while allowing to reason by approximation. The impossibility to evaluate alternatives is supported with *partial* fuzzy rankings while *multiple* fuzzy rankings support incomparability between alternatives. To let use standard GDM methods and tools when preferences are expressed with fuzzy rankings, translation methods to and from FPRs have been provided as well as similarity measures to assess the convergence of experts' opinions.

A possible extension of fuzzy rankings, to be studied in future works, is the adoption of linguistic labels (mapped on fuzzy numbers) to specify the gap between two subsequent elements in the ranking. This would make the model more complex but, at the same time, enable a better representation of the vagueness inherent in the subjective evaluation of alternatives made by experts. Specific GDM approaches based on linguistic assessment, like in [145], could be adapted to deal with this case.

Once defined, the fuzzy ranking model has been used as a building block for a complete GDM model able to consider *social influence* between experts and to estimate how experts' opinions change according to its effects. If fact, despite its prominent role in opinion formation, social influence seems to be almost disregarded by current GDM models. Aiming at filling this gap, the proposed model links the concept of social influence to that of interpersonal trust according to the intuition that the more an expert trusts in another, the more her opinion is influenced by the trusted expert, especially when she is unable to express an opinion on some alternatives.

Fuzzy rankings are used to represent experts' opinions regarding the set of alternatives as well as their trust on other experts. Defined rankings are then used to determine the structure and the level of experts' interpersonal influence used, in turn, to estimate missing preferences and to let them evolve simulating the effects of experts' interaction. The defined model leads to a more accurate representation of the GDM process by formalizing important aspects that are usually disregarded by other models. A future extension of such model could be directed toward multi-criteria (or multi-attribute) GDM that deals with problems where the alternatives are characterized in terms of multiple, usually conflicting, attributes. In such cases, the experts should first of all reach an agreement on the priority of each attribute and, then, on the priority of each alternative with respect to each attribute.

As said, defined models have been specialized in two applicative contexts. With respect to the e-Learning context, a model for ordinal peer assessment, named *FOPA*, has been defined. With FOPA each student is asked to define a fuzzy ranking among some submissions of other students for a given assignment. Students' provided rankings are transformed in FPRs, expanded to estimate missing values and aggregated. The aggregated relation is then used to generate a global ranking between the submissions and to estimate their absolute grades. FOPA has been compared with existing ordinal and cardinal peer assessment models and has shown better performances in several in silico experiments both in the reconstruction of the student ranking and in the estimation of students' grades. Additional experiments with real University students have confirmed the former results.

Despite that it has been conceived for peer assessment, FOPA can be easily adapted in other contexts where several alternatives must be evaluated taking into account the opinion of many assessors but when each assessor has only a partial view of the whole picture. For example, in a *Conference Review Process* many submissions must be ranked (to choose the best ones to invite for presentation and/or to be awarded) basing on a set of (possibly unreliable) experts, each reviewing a relatively small number of works. Another example is the *Employee Reward and Recognition Systems* set up by companies to motivate their employees. Here employee performances are ranked according to suggestions coming from managers, each of them evaluating just the subset of employees involved in the projects she manages.

To validate FOPA and pave the way for future extensions, we have also defined additional peer-assessment models based on graph mining techniques. The assumption of these models, confirmed by some existing studies, is that the grade obtained by a student on a given subject is correlated to her ability as assessor on the same subject. Experimental results with synthetic and real data show that such methods outperforms other existing methods in most configurations even if they show worse results than FOPA. Nevertheless, a possible extension of FOPA is to integrate the same techniques to detect the assessors' reliability and use this information to weight the feedback provided in the aggregation step. To this purpose, preference aggregators that takes experts' importance into account (like I-IOWA) should be preferred to OWA. In addition, to support assessment rubrics, it would be possible to extend the underlying model to multi-criteria GDM.

Finally, with respect to applicative context of Recommender Systems, the group recommendation problem has been tackled. While in GDM, a group of decision makers evaluate a set of alternatives with the aim of selecting the best one to adopt, in GRS the system selects, from a given catalogue, the set of items that best fit the preferences of all (or the majority of) members belonging to a group of users. While the majority of existing GRS approaches just use individual users' preferences to estimate those of the whole group, the proposed approach, based on the defined GDM models, also considers the personality of group members, their interpersonal trust and social influence. Taking such factor into account allows to define a more accurate model that is capable of reaching good recommending performances.

The proposed model is able to build a social influence network starting from information about interpersonal trust and users' personality traits. The network is used, in turn, to evolve users' preferences toward a shared solution. An evolution of the proposed approach is to directly obtain information about interpersonal trust by analyzing implicit data contained in social networks according to the models defined in [118, 119]. In addition, personality traits can be predicted by analyzing the language used in social media according to models defined in [124, 125, 146]. This will make the process transparent to users without the need to fill long questionnaires before system use.

It is worth noting that some of the preliminary results obtained by the candidate during the three-years PhD program (subsequently extended and systematized in this thesis), have been already submitted and accepted for publication and presentation on international conferences and journals. In particular, a first version of the fuzzy GDM model guided by social influence described in chapter 3 has been published in [147], with an embryonic version of the fuzzy ranking model defined in chapter 2. A preliminary version of the FOPA model described in chapter 4 has been published in [148] while the peer assessment methods based on graph mining (described in section 4.5) have been also published in [149, 150]. Some of the experiments reported in chapter 6 have been published in [151, 152] while additional works on fuzzy rankings and GDM-based group recommendations are in preparation.

References

[1] J. Lu, G. Zhang, D. Ruan and F. Wu, Multi-Objective Group Decision Making, Methods, Software and Applications with Fuzzy Set Techniques, World Scientific, 2007.

[2] W. Pedrycz, P. Ekel and R. Parreiras, Fuzzy Multicriteria Decision-Making: Models, Methods and Applications, John Wiley & Sons, 2010.

[3] H. A. Simon, The New Science of Management Decision, Prentice Hall, 1977.

[4] O. I. Larichev, "Psychological validation of decision methods," *Journal of Applied Systems Analysis,* vol. 11, no. 1, pp. 37-46, 1984.

[5] B. C. Y. Tan, H. H. Teo and K. K. Wei, "Promoting consensus in small decision-making groups," *Information and Management,* vol. 28, no. 4, pp. 251-259, 1995.

[6] R. O. Parreiras, P. Y. Ekel, M. J. S. C. and R. M. Palhares, "A flexible consensus scheme for multicriteria group decision-making under linguistic assessments," *Information Sciences,* vol. 180, no. 7, pp. 1075-1089, 2010.

[7] W. P. van Ginkel and D. van Knippenberg, "Knowledge about the distribution of information and group decision-making: when and why does it work?," *Organizational Behavior and Human Decision Processes,* vol. 108, no. 2, pp. 218-229, 2009.

[8] B. L. Bonner, M. R. Baumannb and R. S. Dalal, "The effects of member expertise on group decision-making and performance," *Organizational Behavior and Human Decision Processes,* vol. 88, no. 2, pp. 719-736, 2002.

[9] T. L. Griffith, M. A. Fuller and G. B. Northcraft, "Facilitator influence in group support systems: intended and unintended effects," *Information Systems Research,* vol. 9, no. 1, pp. 20-36, 1998.

[10] O. K. Ngwenyama, N. Bryson and A. Mobolurin, "Supporting facilitation in group support systems: techniques for analyzing consensus relevant data," *Decision Support Systems,* vol. 16, no. 2, pp. 155-168, 1996.

[11] Z. Wong and M. Aiken, "Automated facilitation of electronic meetings," *Information and Management,* vol. 41, no. 2, pp. 125-134, 2003.

[12] J. Marold, R. Wagner, M. Schöbel and D. Manzey, "Decision-making in groups under uncertainty," *Les Cahiers de la Sécurité Industrielle,* no. 5, 2012.

[13] R. Lipshitz and O. Strauss, "Coping with uncertainty: A naturalistic decision-making analysis," *Organizational Behavior and Human Decision Processes,* vol. 69, no. 2, pp. 149-163, 1997.

[14] G. B. Dantzig, "Linear programming under uncertainty," *Management Science,* vol. 1, no. 2, pp. 197-207, 1955.

[15] W. K. Grassman, Stochastic Systems for Management, North-Holland, 1981.

[16] L. S. Belyaev and L. A. Krumm, "Applicability of probabilistic methods in energy calculations," *Power Engineering,* vol. 21, no. 2, pp. 3-10, 1983.

[17] L. A. Zadeh, "Fuzzy sets," *Information and Control,* vol. 8, no. 3, pp. 338-353, 1965.

[18] H. J. Zimmermann, Fuzzy Set Theory and Its Applications, 4th ed., Springer, 2001.

[19] D. Dubois and H. Prade, Fuzzy Sets and Systems: Theory and Applications, Academic Press, 1980.

[20] Q. Zhang, J. C. H. Chen and P. P. Chong, "Decision consolidation: criteria weight determination using multiple preference formats," *Decision Support Systems,* vol. 38, no. 2, pp. 247-258, 2004.

[21] F. Seo and M. Sakawa, "Fuzzy multiattribute utility analysis for collective choice," *IEEE Transactions on Systems, Man and Cybernetics,* vol. 15, no. 1, pp. 45-53, 1985.

[22] F. Chiclana, F. Herrera and E. Herrera-Viedma, "Integrating three representation models in fuzzy multipurpose decision making based on fuzzy preference relations," *Fuzzy Sets and Systems,* vol. 97, no. 1, pp. 33-48, 1998.

[23] V. Belton, "Multiple-criteria problem structuring and analysis in a value theory framework," in *Multicriteria Decision Making: Advances in MCDM Models, Algorithms, Theory,* Kluwer, 1999, pp. 2-29.

[24] R. J. Li, "Fuzzy method in group decision making," *Computers & Mathematics with Applications,* vol. 38, no. 1, pp. 91-101, 1999.

[25] T. Saaty, The Analytic Hierarchy Process, McGraw-Hill, 1980.

[26] X. Zeshui and W. Cuiping, "A consistency improving method in the analytic hierarchy process," *European Journal of Operational Research,* vol. 116, no. 2, pp. 443-449, 1999.

[27] Y. M. Wang and Z. P. Fan, "Fuzzy preference relations: Aggregation and weight determination," *Computers & Industrial Engineering,* vol. 53, no. 1, pp. 163-172, 2007.

[28] J. Ma, Z. P. Fan, Y. P. Jiang, J. Y. Mao and L. Ma, "A method for repairing the inconsistency of fuzzy preference relations," *Fuzzy Sets and Systems,* vol. 157, no. 1, pp. 20-33, 2006.

[29] E. Herrera-Viedma, F. Herrera, F. Chiclana and M. Luque, "Some issues on consistency of fuzzy preference relations," *European Journal of Operational Research,* vol. 154, no. 1, pp. 98-109, 2004.

[30] F. Chiclana, E. Herrera-Viedma, S. Alonso and F. Herrera, "A note on the estimation of missing pairwise preference values: A uninorm consistency based method," *International Journal of Uncertainty, Fuzziness and Knowlege-Based Systems ,* vol. 16, no. Suppl. 2, pp. 19-32, 2008.

[31] V. Peneva and I. Popchev, "Properties of the aggregation operators related with fuzzy relations," *Fuzzy Sets and Systems,* vol. 139, no. 3, pp. 615-633, 2003.

[32] R. Boukezzoula, S. Galichet and L. Foulloy, "MIN and MAX Operators for Fuzzy Intervals and Their Potential Use in Aggregation Operators," *IEEE Transactions on Fuzzy Systems,* vol. 15, no. 6, pp. 1135-1144, 2007.

[33] R. R. Yager, "Families of OWA operators," *Fuzzy Sets and Systems,* vol. 59, no. 2, pp. 125-148, 1993.

[34] L. A. Zadeh, "A Computational Approach to Fuzzy Quantifiers in Natural Languages," *Computers and Mathematics with Applications,* vol. 9, pp. 149-184, 1983.

[35] R. R. Yager and D. Filev, "Induced ordered weighted averaging operators," *IEEE Transactions on Systems, Man and Cybernetics,* vol. 29, pp. 141-150, 1999.

[36] F. Chiclana, E. Herrera-Viedma, F. Herrera and S. Alonso, "Some induced ordered weighted averaging operators and their use for solving group decision-making problems based on fuzzy preference relations," *European Journal of Operational Research,* vol. 182, no. 1, pp. 383-399, 2007.

[37] S. A. Orlovsky, "Decision making with a fuzzy preference relation," *Fuzzy Sets and Systems,* vol. 1, no. 3, pp. 155-167, 1978.

[38] A. Banerjee, "Rational choice under fuzzy preferences: the Orlovsky choice function," *Fuzzy Sets and Systems,* vol. 54, no. 3, pp. 295-299, 1993.

[39] F. Chiclana, F. Herrera, E. Herrera-Viedma and M. C. Poyatos, "A classification method of alternatives for multiple preference ordering criteria based on fuzzy majority," *Journal of Fuzzy Mathematics,* vol. 4, pp. 801-813, 1996.

[40] S. Alonso, F. Chiclana, F. Herrera, E. Herrera-Viedma, J. Alcala-Fdez and C. Porcel, "A consistency-based procedure to estimate missing pairwise preference values," *International Journal of Intelligent Systems,* vol. 23, no. 1, pp. 155-175, 2008.

[41] F. Chiclana, E. Herrera-Viedma, S. Alonso and F. Herrera, "Cardinal consistency of reciprocal preference relations: A characterization of multiplicative transitivity," *IEEE Transactions on Fuzzy Systems,* vol. 17, no. 1, pp. 14-23, 2009.

[42] S. Alonso, E. Herrera-Viedma, F. Chiclana and F. Herrera, "Individual and Social Strategies to Deal with Ignorance Situations in Multi-Person Decision Making," *International Journal of Information Technology and Decision Making,* vol. 8, no. 2, pp. 313-333, 2009.

[43] J. Mazurek, "Evaluation of ranking similarity in ordinal ranking problems," *Acta Academica Karviniensia,* vol. 2, pp. 119-128, 2011.

[44] M. Kendall, "A New Measure of Rank Correlation," *Biometrika,* vol. 30, no. 1-2, pp. 81-89, 1938.

[45] L. A. Goodman and W. H. Kruskal, "Measures of Association for Cross Classifications," *Journal of the American Statistical Association,* vol. 49, no. 268, pp. 732-764, 1954.

[46] C. Spearman, "The proof and measurement of association between two things," *American Journal of Psychology,* vol. 15, pp. 72-101, 1904.

[47] G. Bordogna, M. Fedrizzi and G. Passi, "A Linguistic Modelling of Consensus in Group Decision Making Based on OWA Operators," *IEEE Transactions on Systems, Man and Cybernetics,* vol. 27, pp. 126-132, 1997.

[48] R. Sheldon, Introductory Statistics, 3rd ed., Elsevier, 2010.

[49] J. Mazurek, "Fuzzy Rankings: Properties and Applications," Cornell University Library - arXiv, 2017.

[50] F. Herrera, E. Herrera-Viedma and J. L. Verdegay, "A sequential selection process in group decision making with linguistic assessment," *Information Sciences,* vol. 85, pp. 223-239, 1995.

[51] F. Herrera and E. Herrera-Viedma, "Choice functions and mechanisms for linguistic preference relations," *European Journal of Operational Research,* vol. 120, pp. 144-161, 2000.

[52] S. Abbasbandy, "Ranking of Fuzzy Numbers, Some Recent and New Formulas," in *Proceedings of the Joint International Fuzzy Systems Association World Congress and European Society of Fuzzy Logic and Technology Conference*, Lisbon, Portugal, 2009.

[53] J. Galindo, A. Urrutia and M. Piattini, Fuzzy Databases: Modeling, Design and Implementation, IGI Global, 2005.

[54] J. Galindo, "New Characteristics in FSQL, a Fuzzy SQL for Fuzzy Databases," *WSEAS Transactions on Information Science and Applications,* vol. 2, no. 2, pp. 161-169, 2005.

[55] D. Artz and Y. Gil, "A survey of trust in computer science and the Semantic Web," *Journal of Web Semantics: Science, Services and Agents on the World Wide Web,* vol. 5, no. 2, pp. 58-71, 2007.

[56] M. H. De Groot, "Reaching a consensus," *Journal of American Statistical Association,* vol. 69, pp. 118-121, 1974.

[57] N. Friedkin and E. Johnsen, "Social Influence Networks and Opinion Change," *Advances in Group Processes,* vol. 16, no. 1, pp. 1-29, 1999.

[58] L. G. Pérez, F. Mata, F. Chiclana, G. Kou and E. Herrera-Viedma, "Modelling influence in group decision making," *Soft Computing,* vol. 20, no. 4, pp. 1653-1665, 2016.

[59] Q. Liang, X. Liao and J. Liu, "A social ties-based approach for group decision-making problems with incomplete additive preference relations," *Knowledge-Based Systems,* vol. 119, pp. 68-86, 2017.

[60] J. L. Doob, Stochastic Processes, John Wiley & Sons, 1990.

[61] J. Wu and F. Chiclana, "A social network analysis trust–consensus based approach to group decision-making problems with interval-valued fuzzy reciprocal preference relations," *Knowledge-Based Systems,* vol. 59, pp. 97-107, 2014.

[62] J. Wu, F. Chiclana and E. Herrera-Viedma, "Trust Based Consensus Model for Social Network in an Incomplete Linguistic Information Context," *Applied Soft Computing,* vol. 35, pp. 827-839, 2015.

[63] C. A. Grady, X. He and S. Peeta, "Integrating social network analysis with analytic network process for international development project selection," *Expert Systems with Applications,* vol. 42, pp. 5128-5138, 2015.

[64] T. H. Duong, N. T. Nguyen, H. B. Truong and V. H. Nguyen, "A collaborative algorithm for semantic video annotation using a consensus-based social network analysis," *Expert Systems with Applications,* vol. 42, pp. 246-258, 2015.

[65] N. Dabbagh, A. D. Benson, A. Denham, R. Joseph, M. Al-Freih, G. Zgheib, H. Fake and G. Zhetao, Learning Technologies and Globalization, Springer, 2016.

[66] D. Shah, "By The Numbers: MOOCS in 2016," in *Monetization over Massiveness: A Review of MOOC Stats and Trends in 2016*, Class Central, 2016.

[67] G. Siemens, "Massive Open Online Courses: Innovation in Education?," in *Open Educational Resources: Innovation, Research and Practice*, Commonwealth of Learning, 2013, pp. 5-15.

[68] O. Rodriguez, "The concept of openness behind c and x-MOOCs (Massive Open Online Courses)," *Journal of Chemical Education,* vol. 89, pp. 1133-1137, 2013.

[69] D. G. Glance, M. Forsey and M. Riley, "The pedagogical foundations of massive open online courses," *First Monday,* vol. 18, no. 5, 2013.

[70] T. Daradoumis, R. Bassi, F. Xhafa and S. Caballé, "A review on massive e-learning (MOOC) design, delivery and assessment," in *Proceedings of teh 8th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing*, 2013.

[71] N. Capuano and R. King, "Knowledge-based assessment in serious games: an experience on emergency training," *Journal of e-Learning and Knowledge Society,* vol. 11, no. 3, pp. 117-132, 2015.

[72] I. Caragiannis, A. Krimpas and A. A. Voudouris, "Aggregating partial rankings with applications to peer grading in massive online open courses," in *Proceedings of the International Conference on Autonomous Agents and Multiagent Systems*, Istanbul, 2015.

[73] L. Bouzidi and A. Jaillet, "Can online peer assessment be trusted?," *Educational Technology & Society,* vol. 12, no. 4, pp. 257-268, 2009.

[74] P. A. Carlson and F. C. Berry, "Calibrated Peer Review™ and Assessing Learning Outcomes," in *Proceedings of the 33rd International Conference Frontiers in Education*, 2003.

[75] C. Piech, J. Huang, Z. Chen, C. Do, A. Ng and D. Koller, "Tuned models of peer assessment in MOOCs," in *Proceedings of the 6th International Conference on Educational Data Mining*, 2013.

[76] I. M. Goldin, "Accounting for Peer Reviewer Bias with Bayesian Models," in *Proceedings of the 11th International Conference on Intelligent Tutoring Systems*, 2012.

[77] M. Uto and M. Ueno, "Item Response Theory for Peer Assessment," *IEEE Transactions on Learning Technologies,* vol. 9, no. 2, pp. 157-160, 2016.

[78] L. De Alfaro and M. Shavlovsky, "Crowdgrader: Crowdsourcing the evaluation of homework assignments," in *Proceedings of the 45th ACM Technical Symposium on Computer Science Education*, 2014.

[79] T. Walsh, "The PeerRank Method for Peer Assessment," in *Proceedings of the 21st European Conference on Artificial Intelligence*, 2014.

[80] L. Page, B. Brin, R. Motwani and T. Winograd, "The PageRank citation ranking: Bringing order to the Web”," Stanford InfoLab, 1999.

[81] K. Raman and T. Joachims, "Methods for ordinal peer grading," in *Proceedings of the 20th SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2014.

[82] J. Ma and D. Zhou, "Fuzzy set approach to the assessment of student-centred learning," *IEEE Transactions on Education,* vol. 43, no. 2, pp. 237-241, 2000.

[83] C. H. Lan, S. Graf, K. R. Lai and Kinshuk, "Enrichment of Peer Assessment with Agent Negotiation," *IEEE Transactions on Learning Technologies,* vol. 4, no. 1, pp. 35-46, 2011.

[84] K. C. Chai, K. M. Tay and C. P. Lim, "A new fuzzy peer assessment methodology for cooperative learning of students," *Applied Soft Computing,* vol. 32, pp. 468-480, 2015.

[85] J. C. Borda, "Memoire sur les elections au scrutin," *Histoire de l'Académie Royale des Sciences,* 1781.

[86] A. Abdulkadiroglu and T. Sonmez, "Random Serial Dictatorship and the Core from Random Endowments in House Allocation Problems," *Econometrica,* vol. 66, no. 3, p. 689, 1998.

[87] C. Dwork, R. Kumar, M. Naor and D. Sivakumar, "Rank aggregation methods for the Web," in *Proceedings of the 10th International Conference on World Wide Web*, 2001.

[88] C. L. Mallows, "Non-Null Ranking Models. I," *Biometrika,* vol. 44, no. 1, p. 114, 1957.

[89] R. A. Bradley and M. E. Terry, "Rank Analysis of Incomplete Block Designs: I. The Method of Paired Comparisons," *Biometrika,* vol. 39, no. 3, p. 324, 1952.

[90] R. L. Plackett, "The Analysis of Permutations," *Applied Statistics,* vol. 24, no. 2, p. 193, 1975.

[91] R. J. Hyndman and A. B. Koehler, "Another look at measures of forecast accuracy," *International Journal of Forecasting,* vol. 22, no. 4, pp. 679-688, 2006.

[92] C. Willmott and K. Matsuura, "On the use of dimensioned measures of error to evaluate the performance of spatial interpolators," *International Journal of Geographic Information Science,* vol. 20, pp. 89-102, 2006.

[93] S. P. Borgatti, "Centrality and Network Flow," *Social Networks,* vol. 27, pp. 55-71, 2005.

[94] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions," *IEEE Transactions on Knowledge and Data Engineering,* vol. 17, no. 6, pp. 734-749, 2005.

[95] C. C. Aggarwal, Recommender Systems: The Textbook, Springer, 2016.

[96] J. A. Konstan, B. N. Miller, D. Maltz, J. L. Herlocker, L. R. Gordon and J. Riedl, "Grouplens: applying collaborative filtering to usenet news," *Communications of ACM,* vol. 40, no. 3, pp. 77-87, 1997.

[97] G. Linden, B. Smith and J. York, "Amazon.com recommendations: Item-to-item collaborative filtering," *IEEE Internet Computing,* vol. 7, no. 1, pp. 76-80, 2003.

[98] C. Desrosiers and G. Karypis, "A comprehensive survey of neighborhood-based recommendation methods," in *Recommenders Systems Handbook*, Springer, 2011, pp. 107-144.

[99] L. Ungar and D. Foster, "A Formal Statistical Approach to Collaborative Filtering," in *Proceedings of the Conference on Automated Learning and Discovery*, Pittsburgh, USA, 1998.

[100] Y. Koren, R. Bell and C. Volinsky, "Matrix Factorization Techniques for Recommender Systems," *Computer,* vol. 42, no. 8, pp. 30-37, 2009.

[101] Y. Koren, "Factorization Meets the Neighbourhood: A Multifaceted Collaborative Filtering Model," in *Proceendings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2008.

[102] K. G. Sarwar B M, J. Konstan and J. Riedl, "Application of Dimensionality Reduction in Recommender System - A Case Study," in *Proceedings of the KDD Workshop on Web Mining for e-Commerce: Challenges and Opportunities*, 2000.

[103] G. Shani and A. Gunawardana, "Evaluating Recommendation Systems," in *Recommender Systems Handbook*, Springer, 2011, pp. 257-297.

[104] C. J. Van Rijsbergen, Information Retrieval, Butterworth-Heinemann, 1979.

[105] J. S. Breese, D. Heckerman and C. Kadie, "Empirical analysis of predictive algorithms for collaborative filtering," in *Proceedings of the 14th Conference on Uncertainty in artificial intelligence*, Madison, Wisconsin, USA, 1998.

[106] K. Jarvelin and J. Kekalainen, "Cumulated gain-based evaluation of IR techniques," *ACM Transactions on Information Systems,* vol. 20, no. 4, pp. 422-446, 2002.

[107] L. Baltrunas, T. Makcinskas and F. Ricci, "Group recommendations with rank aggregation and collaborative filtering," in *Proceedings of the 4th ACM conference on Recommender Systems*, New York, 2010.

[108] J. L. Herlocker, J. A. Konstan, T. L. G and J. T. Riedl, "Evaluating collaborative filtering recommender systems," *ACM Transactions on Information Systems,* vol. 22, no. 1, pp. 5-53, 2004.

[109] B. Smyth and P. McClave, "Similarity vs. diversity," in *Proceedings of the 4th International Conference on Case-Based Reasoning*, Vancouver, Canada, 2001.

[110] D. Kotkov, S. Wang and J. Veijalainen, "A survey of serendipity in recommender systems," *Knowledge-Based Systems,* vol. 111, pp. 180-192, 2016.

[111] S. Vargas and P. Castells, "Rank and relevance in novelty and diversity metrics for recommender systems," in *Proceedings of the 5th ACM Conference on Recommender Systems*, New York, USA, 2011.

[112] J. Masthoff, "Group Recommender Systems: Combining Individual Models," in *Recommender Systems Handbook*, Springer, 2011, pp. 677-702.

[113] T. De Pessemier, S. Dooms and L. Martens, "Comparison of group recommendation algorithms," *Multimedia Tools and Applications,* vol. 72, no. 3, pp. 2497-2541, 2014.

[114] J. L. Herlocker, J. A. Konstan and J. Riedl, "Explaining collaborative filtering recommendations," in *Proceedings of the 2000 ACM conference on computer supported cooperative work*, New York, USA, 2000.

[115] J. Masthoff, "Group modeling: selecting a sequence of television items to suit a group of viewers," *User Modeling and User-Adapted Interaction,* vol. 14, no. 1, pp. 37-85, 2004.

[116] J. Masthoff and A. Gatt, "In Pursuit of Satisfaction and the Prevention of Embarrassment: Affective state in Group Recommender Systems," *User Modeling and User-Adapted Interaction,* vol. 16, no. 3-4, pp. 281-319, 2006.

[117] D. Z. Levin, R. Corss and L. C. Abrams, "The strength of weak ties you can trust: the mediating role of trust in effective knowledge transfer," *Management Science,* vol. 50, pp. 1477-1490, 2004.

[118] E. Gilbert and K. Karahalios, "Predicting tie strength with social media," in *Proceedings of the 27th International Conference on Human Factors in Computing Systems*, Boston, USA, 2009.

[119] L. Quijano-Sanchez, J. A. Recio-Garcia, B. Diaz-Agudo and G. Jimenez-Diaz, "Social factors in group recommender systems," *ACM Transactions on Intelligent Systems and Technology,* vol. 4, no. 1, pp. 8:1-8:30, 2013.

[120] K. Thomas and R. Kilmann, Thomas-Kilmann Conflict Mode Instrument, New York, USA: Tuxedo, 1974.

[121] J. E. Barbuto, K. A. Phipps and Y. Xu, "Testing relationships between personality, conflict styles and effectiveness," *International Journal of Conflict Management,* vol. 21, no. 4, pp. 434-447, 2010.

[122] L. C. Messarra, S. Karkoulian and A. N. El-Kassar, "Conflict resolution styles and personality: The moderating effect of generation X and Y in a non-Western context," *International Journal of Productivity and Performance Management,* vol. 65, no. 6, pp. 792-810, 2016.

[123] F. De Fruyt, R. R. McCrae, Z. Szirmák and J. Nagy, "The Five-Factor personality inventory as a measure of the Five-Factor Model: Belgian, American, and Hungarian comparisons with the NEO-PI-R," *Assessment,* vol. 11, no. 3, pp. 207-215, 2004.

[124] H. A. Schwartz, J. C. Eichstaedt, M. L. Kern, L. Dziurzynski, S. M. Ramones, M. Agrawal, A. Shah, M. Kosinski, D. Stillwell, S. M. E. P and U. L. H, "Personality, Gender, and Age in the Language of Social Media: The Open-Vocabulary Approach," *PLoS ONE,* vol. 8, no. 9, 2013.

[125] M. Kosinski, D. Stillwell and T. Graepel, "Private traits and attributes are predictable from digital records of human behavior," *Proceedings of the National Academy of Sciences of the United States of America,* vol. 110, no. 15, pp. 5802-5805, 2013.

[126] J. McCarthy and T. Anagnost, "MusicFX: an arbiter of group preferences for computer supported collaborative workouts," in *Proceedings of the ACM conference on Computer Supported Cooperative Work*, New York, USA, 1998.

[127] A. Crossen, J. Budzik and K. J. Hammond, "Flytrap: intelligent group music recommendation," in *Proceedings of the 7th international conference on Intelligent User Interfaces*, New York, USA, 2002.

[128] M. O'Connor, D. Cosley, J. A. Konstan and J. Riedl, "Polylens: a recommender system for groups of users," in *Proceedings of the 7th conference on European conference on computer supported cooperative work*, Norwell, USA, 2001.

[129] Z. Yu, X. Zhou, Y. Hao and J. Gu, "TV Program Recommendation for Multiple Viewers Based on user Profile Merging," *User Modeling and User-Adaped Interaction,* vol. 16, no. 1, pp. 63-82, 2006.

[130] D. Goren-Bar and O. Glinansky, "Family stereotyping-a model to filter tv programs for multiple viewers," in *Proceedings of the 2nd workshop on personalization in future tv*, Malaga, Spain, 2002.

[131] A. Jameson, S. Baldes and T. Kleinbauer, "Two methods for enhancing mutual awareness in a group recommender system," in *Proceedings of the working conference on Advanced Visual Interfaces*, New York, USA, 2004.

[132] K. McCarthy, M. Salamo, L. Coyle, L. McGinty, B. Smyth and P. Nixon, "Cats: a synchronous approach to collaborative group recommendation," in *Florida Artificial Intelligence Research Society Conference*, 2006.

[133] J. McCarthy, "Pocket restaurant finder: a situated recommender system for groups," in *Proceedings of the ACM Conference on Human Factors in Computing Systems*, Minneaoplis, USA, 2002.

[134] L. Ardissono, A. Goy, G. Petrone, M. Segnan and P. Torasso, "Tailoring the recommendation of tourist information to heterogeneous user groups," *Lecture Notes in Computer Science,* vol. 2266, pp. 228-231, 2002.

[135] L. Quijano-Sanchez, J. A. Recio-Garcia and B. Diaz-Agudo, "Personality and social trust in group recommendations," in *Proceedings of the 22nd IEEE International Conference on Tools with Artificial Intelligence*, Washington, DC, USA, 2010.

[136] R. R. Sinha and K. Swearingen, "Comparing recommendations made by online systems and friends," in *Proceedings of the DELOS Workshop: Personalisation and Recommender Systems in Digital Libraries*, 2001.

[137] J. Golbeck, "Generating predictive movie recommendations from trust in social networks," in *Proceedings of the 4th International Conference on Trust Management*, 2006.

[138] J. Golbeck, "Combining provenance with trust in social networks for semantic web content filtering," *Lecture Notes in Computer Science,* vol. 4145, pp. 101-108, 2006.

[139] J. Castro, F. J. Quesada, I. Palomares and L. Martinez, "A Consensus-Driven Group RecommenderSystem," *International Journal of Intelligent Systems,* vol. 30, pp. 887-906, 2015.

[140] J. Castro, O. Cordon and L. Martinez, "CLG-REJA: A Consensus Location-aware group recommender system for Restaurants," in *9th ACM Conference on Recommender Systems*, Vienna, Austria, 2015.

[141] J. Miguel, S. Caballé, F. Xhafa and J. Prieto, "Security in online Web learning assessment," *World Wide Web,* vol. 18, no. 6, pp. 1655-1676, 2015.

[142] P. Black and D. Wiliam, "Assessment for learning in the classroom," in *Assessment and learning*, SAGE Publications, 2006, pp. 9-15.

[143] J. D. Bransford, A. Brown and R. Cocking, How People Learn: Mind, Brain, Experience and School, National Academy Press, 2000.

[144] N. Seery, D. Canty and P. Phelan, "The validity and value of peer assessment using adaptive comparative judgment in design driven practical education," *International Journal of Technology & Design Education,* vol. 22, no. 2, pp. 205-226, 2012.

[145] F. Herrera, S. Alonso, F. Chiclana and E. Herrera-Viedma, "Computing with Words in Decision Making: Foundations, Trends and Prospects," *Fuzzy Optimization and Decision Making,* vol. 8, no. 4, pp. 337-364, 2009.

[146] N. Capuano, G. D'Aniello, A. Gaeta and S. Miranda, "A personality based adaptive approach for information systems," *Computers in Human Behavior,* vol. 44, pp. 156-165, 2015.

[147] 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 of Fuzzy Systems,* vol. in press, 2017.

[148] N. Capuano, V. Loia and F. Orciuoli, "A Fuzzy Group Decision Making Model for Ordinal Peer Assessment," *IEEE Transactions on Learning Technology,* vol. 10, no. 2, pp. 247-259, 2017.

[149] N. Capuano and S. Caballé, "Towards Adaptive Peer Assessment for MOOCs," in *Proceedings of the 10th International Conference on P2P, Parallel, GRID, Cloud and Internet Computing*, Krakow, Poland, 2015.

[150] N. Capuano, S. Caballé and J. Miguel, "Improving Peer Grading Reliability with Graph Mining Techniques," *International Journal of Emerging Technologies in Learning,* vol. 11, no. 7, pp. 24-33, 2016.

[151] N. Capuano and F. Orciuoli, "Application of Fuzzy Ordinal Peer Assessment in Formative Evaluation," in *Proceedings of the 12th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing*, Barcelona, Spain, 2017.

[152] G. Albano, N. Capuano and A. Pierri, "Adaptive Peer Grading and Formative Assessment," *Journal of e-Learning and Knowledge Society,* vol. 13, no. 1, pp. 147-161, 2017.