M. Detyniecki, B. Bouchon-Meunier and R.Yager, "L'évaluation des nombres flous avec des a-poids," *Proceedings of the Journées francophones sur la Logique Floue et ses Applications - LFA'99,* Valenciennes, France, October, 1999.

L'évaluation des nombres flous avec des α-poids

Valuation of fuzzy numbers with α-weights

Marcin Detyniecki [1] Bernadette Bouchon-Meunier [1] Ronald R. Yager [2]

[1] LIP6 - CNRS - Université Paris VI
[2] Machine Intelligence Institute

4, place Jussieu
75230 Paris Cedex 05, France
Marcin.Detyniecki@lip6.fr

Résumé

Nous étudions principalement ici la méthode d'évaluation des nombres flous proposée par Yager et Filev dans [5]. Cette technique est basée sur l'espérance mathématique relative aux α-coupes et sur une fonction f. Dans cet article nous montrons que cette fonction n'est qu'une distribution de poids associés aux α-coupes. Cette interprétation nous permet de cerner les limites des techniques étudiées et de proposer une extension qui complète et généralise les propositions existantes.

Mots Clef

Evaluation, nombres flous, ordonnancement, α-poids.

Abstract

We studied here the valuation method for fuzzy numbers introduced by the Yager and Filev in [5]. This method is based upon the expected value relatively to the α-cuts and upon a function *f.* In this paper we show that this function is nothing but a weight distribution relatively to the α-cuts. This interpretation show us the limits of the existing method. We suggest then, a generalization that remedy this restrictions.

Keywords

Valuation, fuzzy numbers, ranking, α-weights.

1 Introduction

Dans beaucoup d'applications pratiques de la théorie des ensembles flous, par exemple dans l'aide à la décision, nous sommes confrontés au problème de choisir un élément parmi un ensemble de solutions possibles. En général, ce que nous voulons est tout simplement connaître la meilleur. Ce processus de sélection nous oblige à ordonner les nombres flous. Dans le cas de nombres réels le problème est trivial, mais ordonner des nombres flous n'est pas toujours évident.

Le problème de comparer et classer les sous ensembles flous a été étudié depuis les débuts de la théorie des ensembles flous [1] et s'est avéré un problème difficile. Les principales techniques sont présentes en [2,3]. Nous focalisons ici nos travaux sur une méthode particulière d'ordonnancement. Cette technique a été introduite par Yager [4] et s'appuie sur l'idée d'associer à chaque nombre flou un nombre réel, son évaluation. Puis nous utilisons cette évaluation pour comparer et ordonner les nombre flous. L'originalité de cette méthode réside dans son processus d'évaluation. Plus tard, Yager et Filev [5] ont proposé une méthode améliorée pour cette évaluation. Ils se sont inspirés alors d'évaluations basées sur l'espérance mathématique. Ils ont développé en [5] deux groupes de familles de fonctions d'évaluation paramétrées.

Dans cet article, nous concentrons notre attention sur le processus d'évaluation. Nous commençons par introduire la méthode d'évaluation proposée initialement par Yager. Puis nous étudions les différentes extensions apparues par la suite. Nous poursuivons en donnant une interprétation à la méthode d'évaluation, qui nous a permis par la suite d'en exhiber les défauts. Finalement, nous proposons une nouvelle qui relaxe les contraintes imposées a priori dans ses méthodes. Il est remarquable que cette nouvelle famille fait apparaître deux cas complémentaires de ceux proposés par Yager et Filev.

2 Méthode d'évaluation

Une des méthodes classiques pour s'attaquer au problème d'ordonnancement d'ensembles flous consiste à associer à chaque nombre flou F une valeur réelle représentative, $Val(F)$, puis à comparer les ensembles flous en utilisant simplement cette valeur réelle. Un exemple de cette approche a été introduit par Yager [4] dans lequel il propose :

$$Val(F) = \int_0^1 Moyenne\left(F_\alpha\right) \cdot d\alpha \tag{1}$$

où $F_\alpha = \{x | F(x) \geq \alpha\}$ est l'α-coupe de F et $Moyenne(F_\alpha)$ est la moyenne arithmétique des élément du sous-ensemble F_α.

Remarque: Si F n'est pas normalisé, nous intégrerons alors entre 0 et $\sup_x (F(x))$ et nous multiplierons l'intégrale par $1/\sup_x (F(x))$.

Yager et Filev [5], en s'appuyant sur leurs travaux sur la transformation d'un sous-ensemble flou en une distribution probabiliste associée [6], étendent la formulation (1) et développent une formule générale :

$$Val(F) = \frac{\int_0^1 Moyenne\left(F_\alpha\right) \cdot f(\alpha) \cdot d\alpha}{\int_0^1 f(\alpha) \cdot d\alpha} \qquad (2)$$

Dans ce qui précède f: $[0,1] \rightarrow [0,1]$.
Dans [5] Yager et Filev proposent deux familles complémentaires de fonctions paramétrées à utiliser dans les évaluations du type (2). L'une d'entre elles est croissante et l'autre décroissante. Etudions maintenant ces deux familles :

2.1 La famille croissante

$$f : \alpha \rightarrow f(\alpha) = \alpha^q \quad \text{avec} \quad q \geq 0 \qquad (3)$$

Nous avons alors deux cas *particuliers* intéressants (les cas limites) :

- pour $q = 0$, nous avons f constant égal à 1. Nous retrouvons ainsi la méthode d'évaluation originelle (1).
- pour $q \rightarrow \infty$, nous obtenons la fonction de Dirac translatée en 1. Nous obtenons alors une évaluation égale à $Val(F) = Moyenne\left(F_1\right)$. Il s'agit de la moyenne du noyau.

2.2 La famille décroissante

Il s'agit du cas complémentaire :

$$f : \alpha \rightarrow f(\alpha) = (1-\alpha)^q \quad \text{avec} \quad q \geq 0 \qquad (4)$$

Ici nous avons aussi deux cas *particuliers* intéressants (les cas limites):

- pour $q = 0$, f est constant égal à 1 et donc nous obtenons l'évaluation proposée originellement (1).
- pour $q \rightarrow \infty$, nous obtenons la fonction de Dirac. L'évaluation est donc égale à $Val(F) = Moyenne(F_0)$, c'est-à-dire la moyenne du support.

2.3 Interprétation

En regardant de plus près la formule (2) nous observons que f (α) est tout simplement le poids donné à la moyenne des éléments de l'α-coupe. En d'autres termes, si pour $\alpha \in [0,1]$ la valeur de f est élevée, ceci veut dire qu'on donne un poids important à cette α-coupe particulière. En gardant ceci à l'esprit, traçons sur un même graphique quelques fonctions de la famille de fonctions croissantes (3) pour différentes valeurs de q.

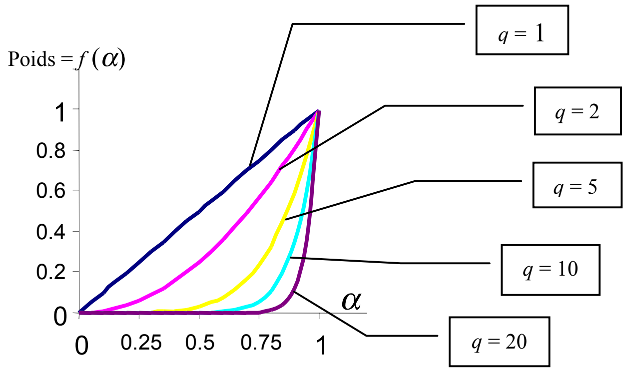
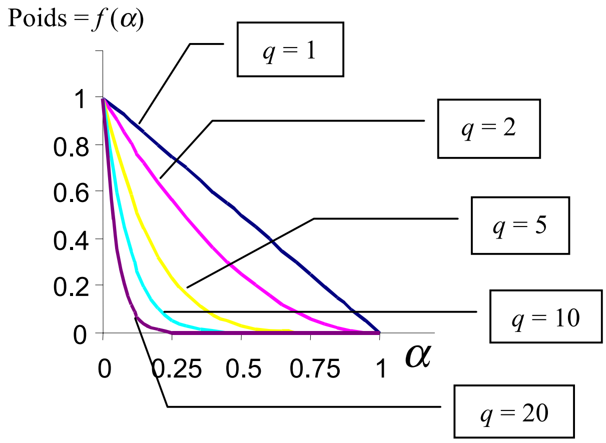


Figure 1: Fonction croissante f pour différents q

Nous observons ici que, l'augmentation de q réduit beaucoup plus les poids des coupes de faible α que ceux des coupes correspondant à α élevé. Nous observons un comportement analogue dans le cas complémentaire de la famille décroissante (4) :

Figure 2: Fonction décroissante f pour différents q

Nous observons dans ce cas un comportement symétrique du précédent. L'augmentation de q agit ici en réduisant



beaucoup plus les poids des coupes correspondant à α élevé que celui des α faibles.

3 Extension pour les fonctions poids

3.1 Introduction

En regardant à nouveau la Figure 1 et la Figure 2, nous remarquons que les fonctions de poids proposées par Yager et Filev fixent le poids du noyau et du support. En effet, dans le cas décroissant le poids associé au support est toujours égal à 1 et le poids associé au noyau égal à 0. Dans le cas croissant nous avons l'inverse, c'est-à-dire 0 pour le noyau et 1 pour le support. Autrement dit, le poids du noyau et du support ne peuvent être que 0 ou 1. Dans cette partie, nous présentons une forme générale de fonction de poids avec laquelle nous pouvons faire varier le poids du noyau et celui du support. Ensuite nous étudions les conséquences d'une telle extension, faisant apparaître quelques nouvelles fonctions intéressantes.

3.2 Définition des fonctions pro-support et pro-noyau

Soit A le poids associé au noyau et B le poids associé au support. Nous proposons les fonctions:

$$f_s(\alpha) = A_s + \left(B_s - A_s\right) \cdot \alpha^q \qquad (5)$$

$$f_n(\alpha) = B_n + (A_n - B_n) \cdot (1-\alpha)^q \qquad (6)$$

où $q \in \mathrm{N}$.

Nous allons appeler la première fonction (5) la famille pro-support et la deuxième (6) la famille pro-noyau. Ces nommes seront expliqués dans les paragraphes qui suivent.

Nous remarquons que, dans le deux cas (pro-support (5) et pro-noyau (6)), nous avons **pour tout *q*** un poids du noyau égal à B_i (i étant n ou s) et un poids du support égal à A_i , c'est qui est exactement ce que nous souhaitions. On aimerait insister ici sur le fait que ces poids ne changent pas avec les variations de q (voir Figure 4 et Figure 4).

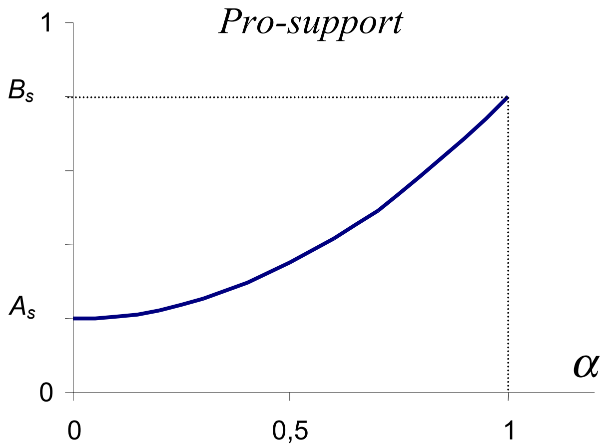


Figure 3: Exemples d'une fonctions pro-support

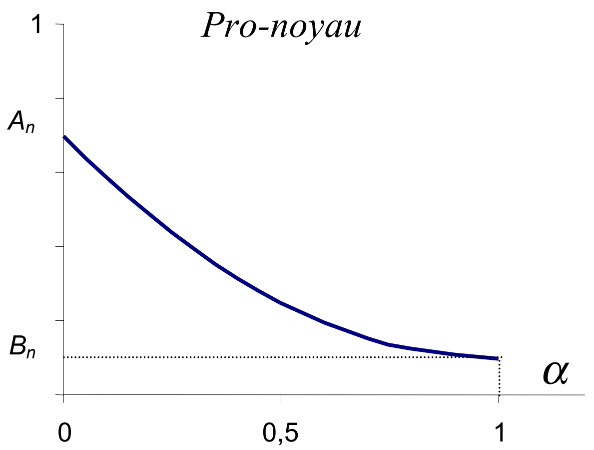


Figure 4: Exemples d'une fonctions pro-noyau

Un autre résultat intéressant est que nous obtenons la famille croissante avec la famille pro-support pour A_s=0 et B_s=1. De même nous obtenons la famille décroissante avec la famille pro-noyau pour A_n=1 et B_n=0. Remarquons que la croissance ou décroissance des fonctions pro-support ou pro-noyau dépendent des valeurs relatives de A_i et B_i .

3.3 Comportement des fonctions pro-support et pro-noyau

Essayons de comprendre le comportement de familles pro-noyau et pro-support dans la Figure 5 et Figure 6 pour différentes valeurs de q.

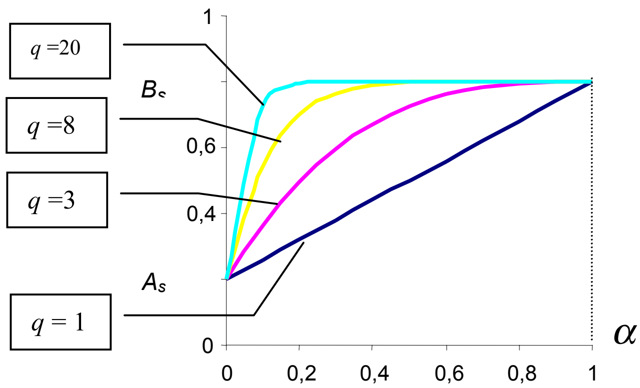


Figure 5: Fonctions pro-support pour A= 0.2 , B= 0.8 et différents q

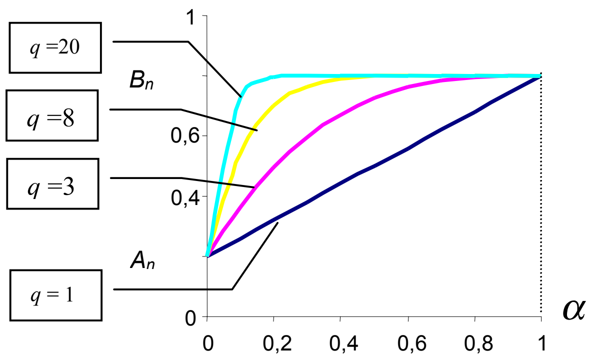


Figure 6: Fonctions pro-noyau pour A= 0.2 , B= 0.8 et différents q

Les fonctions pro-noyau tendent, quand q augmente, vers la constante B_n , qui est le poids associé au noyau. C'est pourquoi nous avons appelé ce groupe de fonctions la famille pro-noyau.

De manière analogue, les fonctions pro-support tendent, quand q augmente, vers le poids A_s du support. Plus formellement, nous avons :

- Pour la famille pro-support :

Pour $\alpha \neq 0$, (7)

$$f_s(\alpha) = A_s + (B_s - A_s) \cdot \alpha^q \xrightarrow[q \to \infty]{} A_s$$

- Pour la famille pro-noyau :

Pour $\alpha \neq 1$, (8)

$$f_n(\alpha) = B_n + (A_n - B_n) \cdot (1-\alpha)^q \xrightarrow[q \to \infty]{} B_n$$

3.4 Quelques cas particuliers intéressants

Yager et Filev ont proposés deux cas particuliers: d'abord la famille pro-support avec A_s=0 et B_s=1, qui correspond à la famille croissante (3) et ensuite la famille pro-noyau avec A_n=1 et B_n=0, qui correspond à la famille décroissante (4). On voit tout naturellement apparaître deux familles symétriques des précédentes: d'abord les fonctions pro-support avec A_s=1 et B_s=0 et puis les fonctions pro-noyau avec A_n=0 et B_n=1:

- ces fonctions pro-support sont des fonctions décroissantes, où l'augmentation de q fait tendre les poids vers celui du support. Elles ne coïncident avec la famille décroissante de Yager et Filev que pour q=1.
- ces fonctions pro-noyau sont de fonctions croissantes, pour lesquelles l'augmentation de q fait tendre les poids vers celui du noyau. Elles ne coïncident avec la famille croissante de Yager et Filev que pour q =1.

Un autre cas particulier remarquable correspond à $A_i = B_i = k \neq 0$. Nous obtenons alors la fonction originale proposée par Yager (1). Un autre cas où nous obtenons l'évaluation originale de Yager (1) correspond à q =0 pour n'importe quel A_i ou B_i. En d'autres termes, nous obtenons l'évaluation classique si A_i égale B_i ou bien comme cas limite pour q égal à zéro.

4 L'interprétation

Remarquons tout d'abord que les processus d'évaluation peuvent aussi être présentés comme des méthodes de défuzzification [7] et que défuzzification et ordonnancement sont des techniques centrales dans la modélisation de systèmes de commande floue [8], d'aide à la décision, et de recherche d'information.

Les techniques d'évaluation présentées précédemment semblent très intéressantes pour chacune de ces applications, car elle permettent d'introduire lors du processus de défuzzification l'importance (poids) des différentes α-coupes. Autrement dit, on peut pondérer le zones incertaines ou certaines de chaque nombre flou. De cette manière on voit apparaître différentes attitudes lors du processus d'évaluation, ce qui donne sur les 4 familles remarquables:

- Pro-support croissant (A_s=0 ; B_s=1) : Cette famille traduit une attitude très pessimiste. On donne plus de poids au coupes de coefficient dont α est élevé et, plus q augmente, plus les seuls α très élevés gardent un poids élevé. On est donc de plus en plus pessimiste. Dans le cas limite ($q\to\infty$) on obtient comme évaluation la moyenne de ce dont on est certain (le noyau).
- Pro-noyau croissant (A_n=0 ; B_n=1) : Cette famille est aussi pessimiste, mais moins que celle qui précède, car on associe toujours des poids plus élevées aux valeurs élevées du coefficient α. Cependant, l'attitude est différente lorsque q augmente. Dans ce cas, les seuls α très faibles gardent un poids faible. On est donc de moins en moins pessimiste. Dans le cas limite ($q\to\infty$) on obtient une évaluation indifférente où tous les poids sont les mêmes (évaluation classique).
- Pro-noyau décroissant (A_n=1 ; B_n=0) : Cette famille est optimiste, car on donne des poids élevées aux coefficients α faibles, c'est-à-dire qu'on a confiance dans les niveaux de grande incertitude. L'augmentation de q donne une attitude de moins en moins optimiste, en effet les seuls poids des coefficients α très élevés restent faible. Dans le cas limite ($q\to\infty$) on obtient bien évidemment l'évaluation indifférente où tous les poids sont les mêmes (évaluation classique).
- Pro-support décroissant (A_s=1 ; B_s=0): De manière analogue aux cas précédents, on remarque que cette famille a une attitude optimiste et qu'elle l'est de plus en plus lorsque q augmente. Dans le cas limite ($q\to\infty$) on obtient comme évaluation la moyenne du support.

Nous remarquons que le cas limite q = 0, donne l'attitude d'indifférence (i.e. le même poids pour toutes les coupes). Nous signalons aussi que si A ou B ne prennent pas les valeurs 0 ou 1, alors les remarques précédentes restent valables mise à part les cas limites ($q\to\infty$) qui donne alors toujours l'évaluation classique. Nous observons alors que plus l'écart entre A et B est grand plus le caractère pessimiste ou optimiste est fort.

5 Conclusions

Dans cet article, nous étudions les processus d'évaluation de nombres flous. Il s'agit d'une étape indispensable dans les techniques de classement. Pour plus de détails sur les interactions entre ces méthodes et le classement obtenu, voir [9].

Nous avons donc commencé par présenter la méthode générale d'évaluation (2) introduite par Yager et Filev dans [5] et ses deux familles de fonctions paramétrées. Nous poursuivons en donnant une interprétation à ces fonctions. Ceci nous a permis de remarquer que les fonctions proposées par Yager et Filev fixent les poids du noyau et du support à 0 ou 1. Nous proposons alors une façon d'étendre ces deux familles et nous voyons apparaître les familles pro-noyau et pro-support. Nous étudions l'effet du paramètre q sur leur comportement. Ceci nous permet de comprendre et de donner une interprétation pour ces deux familles. Nous constatons alors que les fonctions pro-noyau et pro-support sont suffisamment générales pour inclure les familles proposées par Yager et Filev et pour faire apparaître deux familles complémentaires.

Nous finissons en présentant une interprétation pour les méthodes d'évaluation présentées.

Références

[1]. Baas, S.M. et Kwakernaak, H., Rating and ranking multiple-aspect alternatives using fuzzy sets, in *Automatica*, 13 , 47-58, 1977.

[2]. Bortolan, G. et Degani, R., A review of some methods for ranking fuzzy subsets, in *Fuzzy Sets and Systems*, 15, 1-19, 1985.

[3]. Facchinetti, G., Ricci, R.G. et Muzzioli, S., Note on ranking fuzzy triangular numbers, in *International Journal of Intelligent Systems*, 13, 613-622, 1998.

[4]. Yager, R.R., A procedure for ordering fuzzy subsets of the unit interval, in *Information Sciences*, 24 , 143-161, 1981.

[5]. Yager, R.R., On ranking fuzzy numbers using évaluations, in *Technical Report* MII-1901, Machine Intelligence Institute, Iona College, New Rochelle, NY, 1998.

[6]. Yager, R.R. et Filev D.P., On the instantiation of possibility distributions, in *Technical Report* MII-1817, Machine Intelligence Institute, Iona College, New Rochelle, NY, 1998.

[7]. Yager, R.R. et Filev D.P., On the issue of defuzzification and selection based on a fuzzy set, in *Fuzzy Sets and Systems*, 55, 255-272, 1993.

[8]. Yager, R.R. et Filev D.P., Essentials of fuzzy modeling and control, John Wiley: New York, 1994.

[9]. Detyniecki, M. et Yager, R.R., A Note On Ranking Fuzzy Numbers Using α-Weighted Valuations, in *Technical Report* MII-1913, Machine Intelligence Institute, Iona College, New Rochelle, NY, 1998.