



**HAL**  
open science

## Comment former des groupes d'étudiants homogènes à partir des résultats de SELF ? Présentation d'un outil d'aide à la décision pour la création de groupes

Vincent Brault, Sylvain Coulange, Frédérique Letué, Marie-Pierre Jouannaud, Marie-José Martinez, Anne-Cécile Perret

### ► To cite this version:

Vincent Brault, Sylvain Coulange, Frédérique Letué, Marie-Pierre Jouannaud, Marie-José Martinez, et al.. Comment former des groupes d'étudiants homogènes à partir des résultats de SELF ? Présentation d'un outil d'aide à la décision pour la création de groupes. *Mediazioni. Rivista online du studi interdisciplinari su lingue e culture*, 2021, 32, pp.185-203. hal-03629447

**HAL Id: hal-03629447**

**<https://hal.science/hal-03629447>**

Submitted on 4 Apr 2022

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

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

# **Comment former des groupes d'étudiants homogènes à partir des résultats de SELF ? Présentation d'un outil d'aide à la décision pour la création de groupes**

Vincent Brault, Sylvain Coulange, Frédérique Letué, Marie-Pierre Jouannaud, Marie-José Martinez, Anne-Cécile Perret, Université Grenoble Alpes

---

Citation: Brault, Vincent, Sylvain Coulange, Frédérique Letué, Marie-Pierre Jouannaud, Marie-José Martinez, Anne-Cécile Perret (2021) "Comment former des groupes d'étudiants homogènes à partir des résultats de SELF ? Présentation d'un outil d'aide à la décision pour la création de groupes", in Monica Masperi, Cristiana Cervini, Yves Bardière (eds.) *Évaluation des acquisitions langagières : du formatif au certificatif*, *mediAzioni* 32: A185-A203, <http://www.mediazioni.sitlec.unibo.it>, ISSN 1974-4382.

---

## **1. Introduction**

La majorité des cursus universitaires en France impose l'apprentissage d'au moins une langue étrangère en parallèle de la spécialité choisie. Or, à leur arrivée à l'université, les étudiants possèdent des compétences en langues bien hétérogènes. En anglais, langue qu'ils ont tous en principe étudiée pendant leurs études secondaires, une minorité d'entre eux atteint le niveau B2 pourtant exigé en fin de lycée (Ministère de l'Education Nationale 2005), et les niveaux observés couvrent en réalité tout le spectre du CECRL (Cadre Européen Commun de Référence pour les Langues, Conseil de l'Europe 2001), de A1 (niveau élémentaire) à C2 (niveau expert). Dans les autres langues étrangères, les niveaux et les types de compétences acquises sont d'autant plus variés que les situations personnelles, les conditions et la durée d'apprentissage diffèrent selon les individus et les établissements.

Par ailleurs, les Centres Universitaires d'Études Françaises, regroupés au sein du réseau ADCUEFE<sup>1</sup>, accueillent en France des milliers d'étudiants chaque année venant de pays, cultures, et langues différentes. À l'Université Grenoble Alpes (UGA), en 2018/2019, 3200 étudiants de 116 nationalités différentes ont suivi des cours de Français Langue Étrangère au Centre Universitaire d'Études Françaises (CUEF) en intégrant des groupes de niveau A1 à C1. Tous ces étudiants présentent évidemment des niveaux plus hétérogènes encore, et ont également besoin d'être dirigés vers des groupes différents afin d'être placés dans un cours qui correspond à leur niveau et leur permet de progresser au mieux.

C'est dans cet objectif qu'a été développé le Système d'Évaluation en Langues à visée Formative (SELF<sup>2</sup>, Cervini *et al.* 2013), dans le cadre de l'IDEFI Innovalangues entre 2012 et 2020. Le test SELF évalue trois compétences : la compréhension de l'oral, de l'écrit et l'expression écrite courte et fournit à l'étudiant le niveau du groupe dans lequel il lui suggère de s'inscrire, en plus du niveau acquis dans chaque compétence. SELF est utilisé actuellement en six langues (anglais, italien, japonais, mandarin, espagnol et français). En 2020, SELF cumulait plus de 150 000 passations à travers une quarantaine d'établissements en France depuis les premiers déploiements en 2016. Il est actuellement testé dans plusieurs universités italiennes en vue de son internationalisation.

Chaque test SELF a été conçu selon une méthodologie de conception commune aux six langues, issue de la recherche sur l'évaluation en langues (North *et al.* 2009 ; Chapelle *et al.* 2007 ; Alderson 2005 ; Bachman 2004) et a bénéficié de l'expertise de grands acteurs de l'évaluation tels que *the Association of Language Testers in Europe* (ALTE), France Éducation International (FEI, anciennement CIEP) ou *Cito International*.

---

<sup>1</sup> Association des directeurs de centres universitaires d'études françaises pour étrangers.

<sup>2</sup> Pour en savoir plus : <https://innovalangues.univ-grenoble-alpes.fr/evaluer-pour-former/>

Toutefois, SELF ne permet pas pour l'instant de définir précisément le profil de compétences langagières de l'étudiant au-delà d'un niveau par activité langagière du CECRL. Un profilage plus précis reste compliqué dans la mesure où l'étudiant est soumis à une quantité limitée d'items, afin de réduire le plus possible la durée de passation du test. Cependant il est possible de regrouper les étudiants en fonction de leurs tendances de réponses aux items. L'observation de ces tendances peut potentiellement donner des indices sur le profil de compétence des étudiants, et c'est ce que nous cherchons à faire ici en croisant les résultats des étudiants avec le détail de leurs réponses, mais également avec des paramètres indépendants du test tels que leur(s) langue(s) maternelle(s) ou leur âge.

La mise en groupes des étudiants par SELF selon les niveaux du CECRL a donc été comparée à un autre découpage effectué par des algorithmes de co-clustering automatique. Ces algorithmes consistent en une classification jointe des participants et des items dans le but de construire des groupes homogènes respectivement d'étudiants et d'items. Pour faire cette co-classification, nous nous appuyons sur des modèles de blocs latents (Govaert, Nadif 2003) et leur extension incluant des classes de bruit<sup>3</sup> (Laclau, Brault 2018) permettant de mettre en évidence des items peu discriminants dans la classification des étudiants.

Nous avons développé une interface graphique permettant d'automatiser la procédure allant de la récupération des données en sortie de la plateforme SELF, jusqu'à la visualisation des résultats des méthodes de classement des étudiants en passant par le nettoyage des données et leur traitement statistique. La visualisation des résultats permet entre autres :

- la mise en évidence de la réussite aux items ;
- la création de groupes homogènes d'étudiants ayant réussi les mêmes groupes d'items ;

---

<sup>3</sup> Nous entendons par classe de bruit le sous-ensemble des items pour lesquels le taux de réussite est indépendant des groupes d'étudiants (ces items ne sont pas utiles pour la classification des étudiants parce qu'aucune tendance ne peut être décelée dans les réponses).

- la recherche de la partition optimale des étudiants et des items ;
- la détermination du profil type des groupes d'étudiants par rapport au test proposé (par exemple, ont-ils mieux réussi les items de CO ou de CE ?).

## 2. Présentation de SELF

Les étapes de conception d'un test SELF comprennent une succession de phases de recherches, d'analyses quantitatives et qualitatives mêlant analyses psychométriques (statistiques) et évaluation d'experts indépendants de la conception des items. Ces étapes comprennent :

- 1 la définition du construit de compétence en langue étrangère et de chacune des sous-habiletés testées ;
- 2 la recherche ou la conception (selon les langues) de référentiels par niveaux ;
- 3 la mise en place d'un cahier des charges de conception des items ;
- 4 la conception puis la révision des items par des experts ;
- 5 l'expérimentation à petite échelle de groupes d'items (pilotage) suivie d'analyses psychométriques (Théorie Classique des Tests) pour sélectionner les items les plus discriminants et efficaces pour positionner les étudiants ;
- 6 l'expérimentation à grande échelle (prétest), tous niveaux confondus, suivie d'analyses psychométriques (modèles de réponse à l'item, en particulier le modèle de Rasch, Laveault et Grégoire 2014) pour identifier précisément la difficulté et le comportement de chaque item indépendamment de l'échantillon d'étudiants (calibrage et ajustement au modèle) ;
- 7 la détermination des scores de césure, ou *standard setting*, qui permet, avec l'aide d'un groupe d'experts étrangers à la conception, d'identifier consensuellement le niveau de chaque item en fonction des référentiels choisis ;
- 8 et enfin l'assemblage du test final à partir de la banque d'items validés.

Suite aux premières administrations une fois le test déployé, de nouvelles analyses psychométriques sont effectuées afin de vérifier que le test se comporte comme prévu, et d'effectuer des ajustements le cas échéant.

SELF est doté d'un algorithme de type semi-adaptatif, permettant de réduire le nombre d'items soumis à l'étudiant – et donc le temps de passation – sans compromettre la précision d'évaluation (Bachman 2004). Le test propose dans un premier temps une même sélection d'items à tous les étudiants (minitest initial), qui sont ensuite dirigés automatiquement vers une deuxième sélection d'items adaptés à leur niveau (étape 1, étape 2 ou étape 3) en fonction de leurs résultats au minitest initial (voir un exemple pour le japonais, fig. 1). À l'issue du test, chaque étudiant se voit attribuer un score agrégé, qui correspond au niveau du cours dans lequel SELF lui suggère de s'inscrire, ainsi que le niveau estimé acquis dans chacune des trois sous-habilités évaluées.

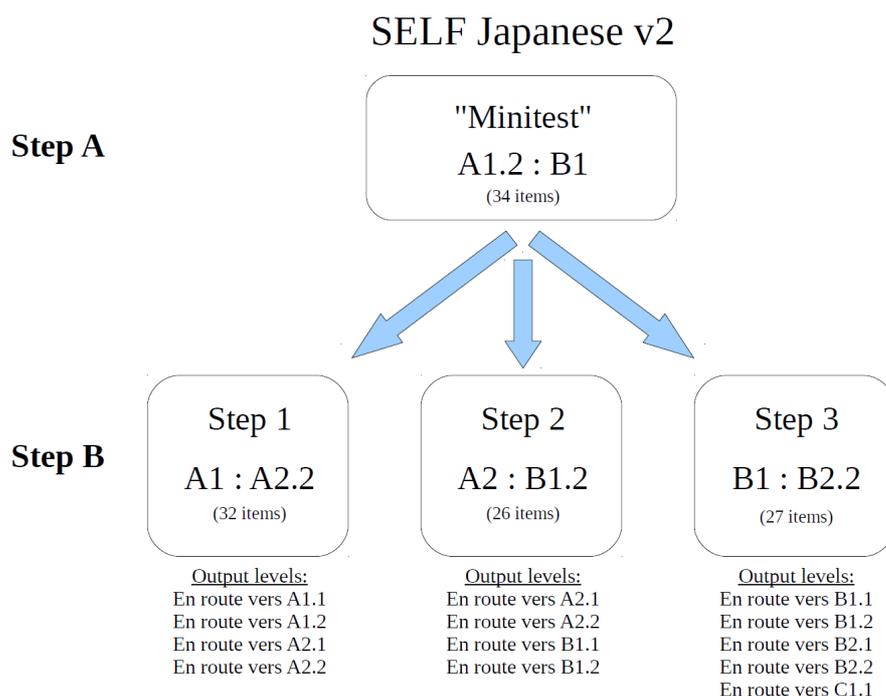


Fig. 1 Structuration des étapes du test SELF en japonais (similaire dans les 6 langues).

Des scores de césure ont été déterminés pour le passage de la première à la deuxième partie du test, et pour le calcul final des scores. Ces scores ont été estimés avec des règles proportionnelles sur les items du test : dans le cas du japonais, un étudiant est dirigé vers l'étape 2 s'il a réussi au moins un tiers des items du minitest dont au moins 5 items de niveau A2.1 ; et il est dirigé vers

l'étape 3 s'il a réussi au moins les deux tiers des items et au moins 4 items de niveau B1.1.

### 3. Limites du test de positionnement

Avec un algorithme comme celui de SELF, basé sur le nombre de bonnes réponses, deux étudiants peuvent obtenir le même score et avoir pourtant des profils très différents (fig. 2). Il pourrait s'avérer pertinent, dans certains contextes de formation, d'obtenir des groupes d'étudiants aux profils similaires, même si leurs scores varient légèrement.

	Q1	Q2	Q3	Q4	Q5	Q6	Q7
Etudiant 1	■			■		■	
Etudiant 2		■	■				■

Note 1 : 4/7  
Note 2 : 4/7

Fig. 2 Deux étudiants avec le même score, mais des profils très différents.

C'est ce que font les modèles des blocs latents (fig. 3). A partir d'une matrice de résultats (tableau avant réorganisation), où chaque ligne correspond à un étudiant et chaque colonne à un item, un carré blanc une bonne réponse et un carré noir une mauvaise réponse, le modèle regroupe simultanément (tableau après réorganisation) les étudiants ayant des réponses similaires (par exemple, A, C et H) et les items qui ont donné lieu à des réponses similaires (par exemple, les items 2 et 6, ratés par beaucoup plus d'étudiants que les autres items). A partir de cette réorganisation, des classes d'étudiants et d'items peuvent être identifiés (tableau avec les classes).

Tableau avant réorganisation

	1	2	3	4	5	6	7
A	■	■	■	■	■	■	■
B	■	■	■	■	■	■	■
C	■	■	■	■	■	■	■
D	■	■	■	■	■	■	■
E	■	■	■	■	■	■	■
F	■	■	■	■	■	■	■
G	■	■	■	■	■	■	■
H	■	■	■	■	■	■	■
I	■	■	■	■	■	■	■
J	■	■	■	■	■	■	■

Tableau après réorganisation

	2	6	1	4	3	5	7
E	■	■	■	■	■	■	■
D	■	■	■	■	■	■	■
G	■	■	■	■	■	■	■
I	■	■	■	■	■	■	■
A	■	■	■	■	■	■	■
C	■	■	■	■	■	■	■
H	■	■	■	■	■	■	■
B	■	■	■	■	■	■	■
F	■	■	■	■	■	■	■
J	■	■	■	■	■	■	■

Tableau avec les classes

	2	6	1	4	3	5	7
E	■	■	■	■	■	■	■
D	■	■	■	■	■	■	■
G	■	■	■	■	■	■	■
I	■	■	■	■	■	■	■
A	■	■	■	■	■	■	■
C	■	■	■	■	■	■	■
H	■	■	■	■	■	■	■
B	■	■	■	■	■	■	■
F	■	■	■	■	■	■	■
J	■	■	■	■	■	■	■

Fig. 3 Utilisation du modèle des blocs latents pour regrouper les étudiants et les items en fonction de leurs réponses.

#### 4. Un outil d'aide à la décision pour la création de groupes d'étudiants

Nous proposons ici un outil d'aide à la décision qui permet à partir des résultats SELF de proposer automatiquement des groupes homogènes d'étudiants (c'est-à-dire ayant un profil similaire en ce qui concerne leurs réponses aux items du test), et simultanément, des groupes homogènes d'items (c'est-à-dire des items ayant reçu les mêmes réponses (mauvaises ou bonnes) par groupe d'étudiants).

##### 4.1. Prétraitement des données

La figure 4 présente les résultats bruts enregistrés par SELF sur un jeu de données réel (session de positionnement en anglais à l'IUT de Grenoble en 2019). Dans cette représentation matricielle,

- chaque ligne représente un étudiant (ordre chronologique de passation) ;
- chaque colonne correspond à un item (ordre chronologique du test) ;
- un carré blanc représente une bonne réponse ;
- un carré noir représente une mauvaise réponse ;
- un carré jaune indique que l'étudiant n'a pas rencontré cet item (i.e. l'item appartient à une étape qu'il n'a pas passée).

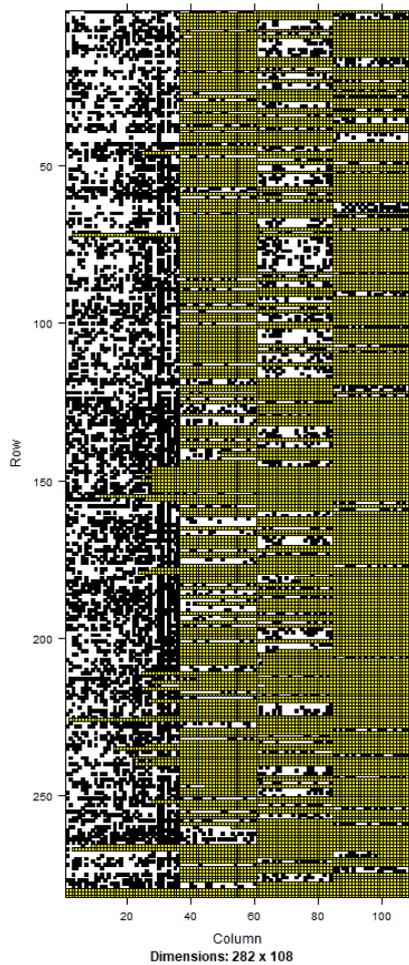


Fig. 4 Représentation matricielle de résultats bruts enregistrés sur SELF.

Dans cette représentation, on distingue quelques lignes presque entièrement jaunes, qui correspondent à des étudiants n'ayant pas terminé le test. Ces étudiants n'ont pas vocation à intervenir dans la composition des groupes d'étudiants. L'application commence donc par supprimer ces lignes pour obtenir la figure 5.

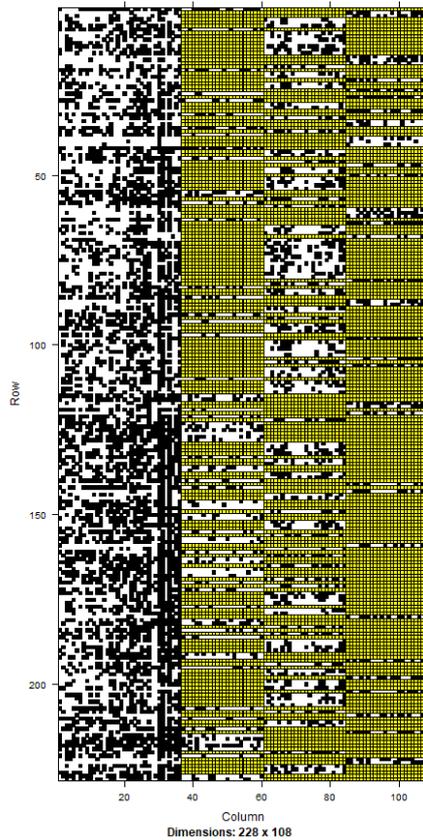


Fig. 5 Représentation matricielle des résultats de SELF après suppression des passations non terminées.

Une fois ces lignes supprimées, on remarque qu'il n'y a plus aucune donnée manquante dans les premières colonnes : il s'agit des items constituant le minitest, passé par tous les étudiants. Dans les colonnes suivantes, les données manquantes apparaissent selon 3 schémas, qui correspondent chacun à l'une des étapes qui suivent le minitest. Pour améliorer la lisibilité des résultats, on peut regrouper les étudiants ayant le même schéma de données manquantes, c'est-à-dire en fonction de l'étape qu'ils ont passée. On reconstitue ainsi les groupes créés par SELF pour obtenir la représentation matricielle de la figure 6. Sur cette Figure, on retrouve maintenant très nettement sur les premières lignes un groupe de 70 étudiants "débutants" ayant répondu aux items de l'étape 1, puis un autre groupe de 117 étudiants "moyens" ayant répondu aux items de l'étape 2, et enfin, un troisième groupe de 41 étudiants "avancés (bons)" ayant répondu aux items de l'étape 3.

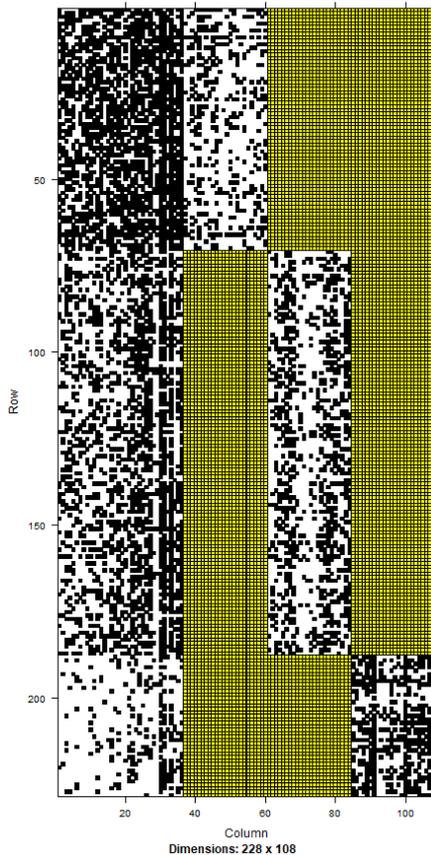


Fig. 6 Représentation matricielle des résultats SELF après prétraitement et ordonnés en fonction de l'étape passée par l'étudiant.

Une fois les résultats ré-ordonnés de cette manière, il est alors aisé de récupérer les items du minitest initial et, pour chaque étudiant, l'étape vers laquelle il a été dirigé par SELF. Dans notre exemple, on travaillera par la suite avec 228 étudiants et les 36 items du minitest. Nous cherchons à répondre aux questions suivantes :

- Quel est le découpage optimal en 3 groupes ?
- Quels sont les items qui participent à ce classement ?
- Quels sont les nombres de groupes d'étudiants et de groupes d'items optimaux ?

Précisons que SELF ne considère pas les résultats dans leur ensemble (tous les étudiants en même temps, comme ici) puisque le résultat est calculé en temps réel pendant la passation. Les résultats de SELF ne dépendent donc que des réponses fournies par l'étudiant, et non des tendances de réponses de la population qui passe le test. De plus, nous ne traitons ici que les résultats obtenus

au minitest, partie commune à tous les étudiants, mais gardons à l'esprit que le test continue ensuite pour affiner le niveau de l'étudiant et déterminer le niveau CERCL en sortie.

## **4.2. Principes de l'Algorithme**

Nous cherchons maintenant, à partir des résultats du minitest, à obtenir des groupes d'étudiants et des groupes d'items les plus homogènes possibles. Nous disposons pour cela d'un algorithme inspiré des modèles à blocs latents, qui permutent les lignes et les colonnes de la matrice afin d'obtenir des blocs homogènes, c'est-à-dire ayant la même densité de carrés noirs ou blancs (mauvaises ou bonnes réponses). L'algorithme effectue simultanément une classification des étudiants (c'est-à-dire une répartition des étudiants en groupes) et une classification des items (c'est-à-dire une répartition des items en groupes). On parle alors de co-classification.

En calculant une distance entre deux profils d'étudiants et une distance entre deux profils d'items, l'algorithme est capable de choisir la co-classification qui minimise la distance entre deux individus d'un même groupe et qui maximise la distance entre deux individus de deux groupes différents. On parle alors de co-classification optimale.

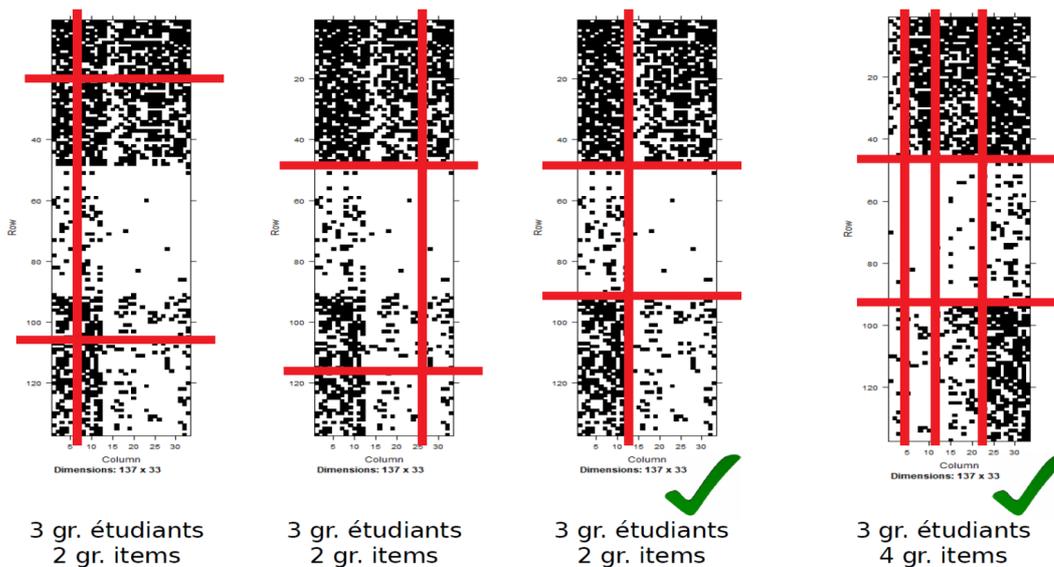


Fig. 7 Trois classifications d'étudiants à 3 groupes et d'items à 2 groupes (à gauche) et une classification d'étudiants à 3 groupes et d'items à 4 groupes (à droite). La troisième classification est optimale parmi celles à 3 groupes d'étudiants et 2 groupes d'items. La quatrième classification est optimale quels que soient les nombres de groupes d'étudiants et de groupes d'items.

Les trois premières matrices de la figure 7 montrent trois co-classifications à trois groupes d'étudiants et 2 groupes d'items. On voit clairement que la troisième est celle qui présente les blocs les plus homogènes (densité homogène de carrés blancs et noirs à l'intérieur d'un bloc). C'est la co-classification optimale parmi toutes celles à 3 groupes d'étudiants et 2 groupes d'items, au sens où les blocs sont les plus homogènes possibles et les plus différents entre eux.

Mais l'algorithme est également capable de comparer des co-classifications à nombre de groupes différents. On peut alors lui faire parcourir une grille de nombres de groupes (par exemple, entre 2 et 5 groupes d'étudiants, et entre 2 et 6 groupes d'items). Pour un nombre donné de groupes d'étudiants et un nombre donné de groupes d'items, il calcule la co-classification optimale, et il compare ensuite ces co-classifications entre elles pour ne garder que la plus homogène. Cette dernière définit alors le nombre optimal de groupes d'étudiants et le nombre optimal de groupes d'items. Dans notre exemple, la co-classification optimale est celle à 3 groupes d'étudiants et 4 groupes d'items, qui semble meilleure que celle à 3 groupes d'étudiants et 2 groupes d'items, comme on peut le voir en comparant les 2 dernières matrices de la figure 7.

### 4.3. Présentation de l'application

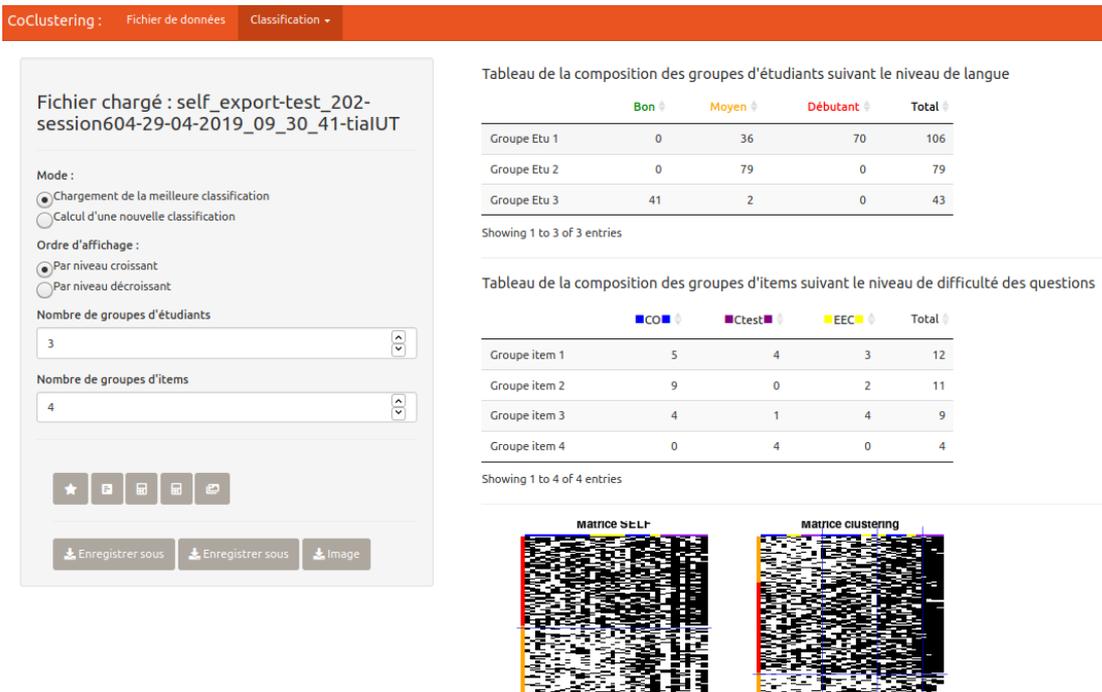


Fig. 8 Aperçu de l'application (CO : Compréhension de l'oral, Ctest : cloze test, EEC : expression écrite courte).

L'application développée dans le cadre du projet CoPoLangues est une application RShiny (Chang *et al.* 2019), développée dans le langage R (R Core Team 2020), très utilisé dans la communauté des statisticiens. Elle permet à un utilisateur d'obtenir la co-classification optimale d'étudiants et d'items à partir d'un fichier de résultats bruts de SELF (fig. 8).

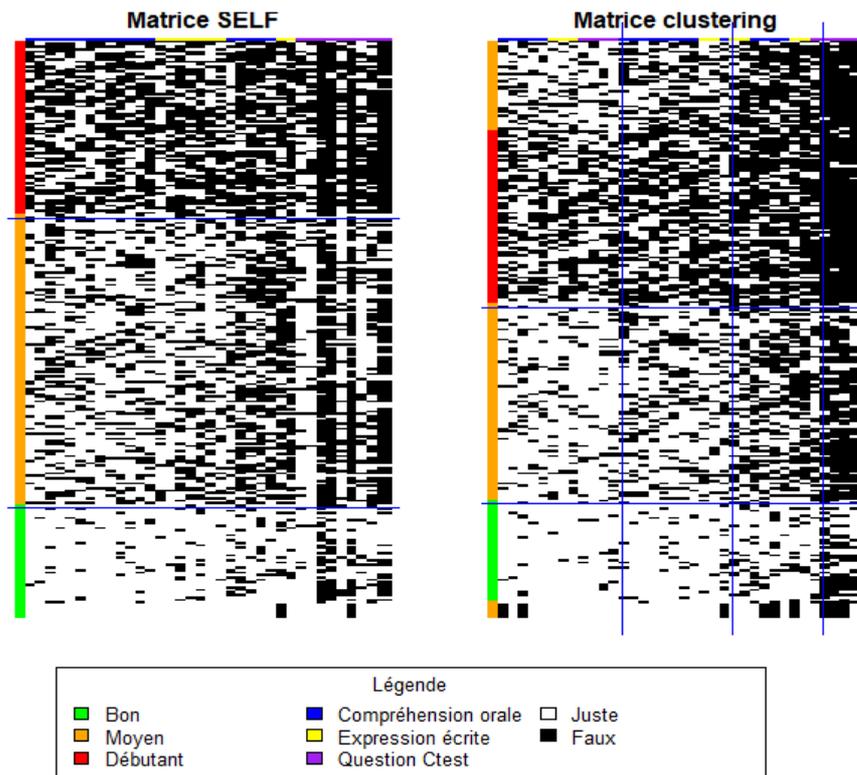


Fig. 9 Exemple de sortie de l'application. A gauche, représentation des résultats bruts de SELF avec les 3 groupes d'étudiants obtenus par SELF (en vert, orange et rouge). A droite, représentation des résultats obtenus par l'algorithme des modèles des blocs latents. Les items ont été colorés selon la compétence visée (en bleu, jaune et violet).

L'application permet également de comparer la répartition des groupes formés par le minitest de SELF. Dans l'exemple mis en table 1, nous observons que le premier groupe formé par l'application est composé de tous les *débutants* et de presque un tiers des *moyens*. Le deuxième groupe est formé du reste des *moyens* sauf deux, et le troisième groupe est composé de tous les *bons* et des deux *moyens* restants. La classification de l'application est donc similaire à celle de SELF pour les *débutants* (qui sont les étudiants ayant été dirigés vers l'étape 1 de SELF suite au minitest) et les *bons* (étudiants dirigés vers l'étape 3). Par contre, elle diffère pour les étudiants *moyens*, dirigés vers l'étape deux, qui sont divisés en trois sous-groupes par l'application : un sous-groupe faible qui rejoint le groupe des *débutants*, un sous-groupe majoritaire, véritablement *moyen*, et un dernier sous-groupe (de deux étudiants seulement) qui rejoint le groupe des *bons*.

### Tableau de la composition des groupes d'étudiants suivant le niveau de langue

	Bon	Moyen	Débutant	Total
Groupe Etu 1	0	36	70	106
Groupe Etu 2	0	79	0	79
Groupe Etu 3	41	2	0	43

Showing 1 to 3 of 3 entries

Tab. 1 Tableau de répartition des étudiants entre les groupes formés par SELF à l'issue du minitest et les groupes formés par l'application.

De même, un tableau est proposé par l'application pour voir la répartition des types de questions. Sur la table 2, nous pouvons observer que le groupe 4, celui composé des questions avec le plus grand nombre d'erreurs, est composé exclusivement de questions du type *C-test* (les *C-tests*, apparentés aux *cloze tests*, utilisent de courts textes dans lesquels un mot sur deux environ est choisi; la deuxième moitié de chacun de ces mots est effacée et les étudiants doivent alors les compléter : "Manger est important. On ne pe\_\_\_ pas surv\_\_\_ sans nourrr\_\_\_." Ce format de test est souvent utilisé dans des contextes de positionnement (Harsch, Hartig 2016).

### Tableau de la composition des groupes d'items suivant le niveau de difficulté des questions

	CO	Ctest	EEC	Total
Groupe item 1	5	4	3	12
Groupe item 2	9	0	2	11
Groupe item 3	4	1	4	9
Groupe item 4	0	4	0	4

Showing 1 to 4 of 4 entries

Tab. 2 Tableau de répartition des types de questions dans les groupes d'item formés par l'application.

D'autres informations sont disponibles comme par exemple le taux de réussite moyen de chaque groupe d'étudiants par groupe d'items.

De plus, il est possible d'explorer par soi-même les données en choisissant le nombre de groupes d'étudiants et d'items ou encore en demandant la meilleure partition obtenue parmi tous les essais faits, ou de choisir le nombre de groupes

d'étudiants et d'obtenir la meilleure classification quelque soit le nombre de groupes d'items.

Enfin, la plupart des informations sont téléchargeables en *png* pour les figures ou en *csv* pour les tableaux.

## **5. Perspectives de recherches**

Les premiers résultats obtenus et la première version de l'application, issus d'une recherche pluridisciplinaire entre didacticiens des langues et statisticiens, sont très prometteurs. Une fois l'application complétée et optimisée, elle permettra aux didacticiens des langues d'analyser plus finement les résultats obtenus au test SELF, d'une part pour améliorer la construction des groupes d'étudiants, mais aussi pour améliorer les tests eux-mêmes.

Il serait en effet pertinent d'inclure une autre classe de modèles (modèles de blocs latents à classe de bruit, Laclau, Brault 2019) incluant un groupe d'items pour lesquels tous les étudiants auraient la même probabilité de réussir ou d'échouer, indépendamment de leur niveau, et donc de leur groupe. Cette perspective permettrait d'isoler des items qui jouent un rôle très faible dans le classement des étudiants en groupes et militerait pour la suppression de ces items dans les tests. L'application aiderait ainsi à optimiser SELF sur le long terme.

Néanmoins, nous sommes conscients que notre application est encore loin de fournir un outil opérationnel pour la mise en groupes dans les institutions, qui sont soumises à bien d'autres contraintes, comme des tailles de groupes par exemple. Elle ne permet pas non plus de qualifier le profil de chaque groupe. Elle reste donc un outil d'aide à la décision, la décision restant aux mains des humains. Il est à mentionner également qu'une certaine hétérogénéité des compétences peut être bénéfique pour certains groupes classes, et peut rendre plus dynamique les échanges. Mais cette hétérogénéité doit rester modérée ; on observe encore trop souvent des groupes où les apprenants plus faibles n'osent

pas s'exprimer devant les plus forts, ou dont la gestion du rythme de progression est rendue difficile par la trop grande disparité des niveaux.

Bien des points sont encore à explorer autour de ces algorithmes et de l'application. De nombreuses fonctionnalités pourraient être ajoutées à l'application, comme l'extraction de listes d'étudiants et d'items par groupes. L'inclusion dans l'étude des étapes secondaires de SELF est également incontournable. Elle ne sera cependant possible que dans le cas de très grandes cohortes, pour avoir suffisamment d'étudiants par étape.

À plus long terme, outre des études plus statistiques sur la robustesse de l'algorithme, nous souhaitons adapter l'algorithme afin d'analyser les biais possibles entre les résultats du test et le profil des apprenants (langues maternelles, nationalité, etc.). Ces résultats nous permettraient d'avoir une réflexion sur l'équité du positionnement proposé par SELF.

## **Remerciements**

L'application a été développée avec l'aide de deux groupes d'étudiants en statistiques de l'université Grenoble Alpes, du DUT STID (STatistique et Informatique Décisionnelle) et du Master SSD (Statistique et Sciences des Données) dans le cadre du projet CoPoLangues<sup>4</sup>; nous les remercions pour leur travail sérieux.

## **Références bibliographiques**

Alderson, J. C. (2005) *Diagnosing Foreign Language Proficiency: The Interface between Learning and Assessment*, London : Continuum.

---

<sup>4</sup> Projet IRS financé par IDEX Université Grenoble Alpes.

Bachman, L. F. (2004) *Statistical Analyses for Language Assessment*, Cambridge: Cambridge University Press.

Cervini, C., M. Masperi, M.-P. Jouannaud et F. Scanu (2013) “Defining, modeling and piloting SELF, a new formative assessment test for foreign languages”, in J. Colpaert, M. Simons, A. Aerts and M. Oberhofer (eds.), *Language Testing in Europe : Time for a new framework?*, Antwerp: University of Antwerp Edition, 55-60, <https://hal.archives-ouvertes.fr/hal-01968811>

Chang, W., J. Cheng, J.J. Allaire, Y. Xie et J. McPherson (2020) *shiny : Web Application Framework for R* (1.5.0) [Computer software], <https://CRAN.R-project.org/package=shiny>

Chapelle, C. A., M.K. Enright and J. M. Jamieson (eds.) (2007) *Building a Validity Argument for the Test of English as a Foreign Language™* (1st edition), New York : Routledge.

Conseil de l'Europe (2001) *Cadre européen commun de référence pour les langues : Apprendre, enseigner, évaluer* (01 éd.), Editions Didier.

Govaert, G. et M. Nadif (2003) “Clustering with block mixture models”, *Pattern Recognition* 36(2): 463-473, doi: [https://doi.org/10.1016/S0031-3203\(02\)00074-2](https://doi.org/10.1016/S0031-3203(02)00074-2)

Harsch, C. et J. Hartig (2015) “Comparing C-tests and Yes/No vocabulary size tests as predictors of receptive language skills”, *Language Testing* 33(4): 555-575, doi: <https://doi.org/10.1177/0265532215594642>

Laclau, C. et V. Brault (2019) “Noise-free Latent Block Model for High Dimensional Data”, *Data Mining and Knowledge Discovery* 33(2): 446-473, doi: <https://doi.org/10.1007/s10618-018-0597-3>

Laveault, D. et J. Grégoire (2014) *Introduction aux théories des tests en psychologie et en sciences de l'éducation*, Bruxelles : De Boeck.

Ministère de l'Education Nationale (2005) *Bulletin officiel n°31 du 1er septembre 2005 – Organisation de l'enseignement des langues vivantes étrangères dans l'enseignement scolaire.*

North, B., N. Figueras, S. Takala, P. Van Avermaet et N. Verhelst (2009) *Manuel pour relier les examens de langues au Cadre européen commun de référence pour les langues (CECR)*, Conseil de l'Europe, [http://www.coe.int/t/dg4/linguistic/manuel1\\_fr.asp](http://www.coe.int/t/dg4/linguistic/manuel1_fr.asp)

R Development Core Team (2020) *R: A language and environment for statistical computing*, R Foundation for Statistical Computing, <http://www.R-project.org>