

TP Gamification adaptative

Élise Lavoué, Stuart Hallifax et Audrey Serna

Ce TP est basé sur les travaux menés dans le cadre du projet e-FRAN [LudiMoodle](#) et de la thèse de Stuart Hallifax (Université de Lyon). Il s'agit d'un environnement numérique (sur moodle) pour réaliser des exercices de mathématiques en classe de collège. Le système affecte à chaque apprenant un élément de jeu pour le motiver à faire les exercices. Vous pouvez vous référer au cours sur la gamification adaptative (Élise Lavoué) et celui sur la plasticité et l'adaptation des IHM (Audrey Serna) pour plus de détails.

Dans ce TP, nous nous intéresserons à la mise en place de cet algorithme d'adaptation statique des éléments de jeu à partir du modèle de l'utilisateur (élève).

Les éléments de jeu ayant été mis à disposition sur l'environnement numérique de l'élève sont :

- Avatar
- Badges
- Scores
- Timing
- Progression individuelle dans la tâche
- Classement par rapport à d'autres élèves

Les données utilisateurs (sur les élèves) à disposition pour ce TP sont :

- Leurs motivations initiales à faire des mathématiques (identifiées à partir du questionnaire AMS)
- Leur profil de joueur HEXAD (identifié à partir d'un questionnaire)

NOTATION : Ce TP est noté, un barème indicatif est indiqué pour chaque question.

RENDU : Travail à rendre le 18/01/2021

Vous rendrez une archive avec vos noms et prénoms contenant les fichiers sources de vos différents algorithmes + un compte-rendu (CR) pour les questions qui demandent des justifications.

Première partie : Recommandations à partir de profils (1h30) //

8 points

Objectif : Mettre en évidence les liens entre éléments de jeu et profils utilisateurs à partir d'analyses statistiques pour émettre des recommandations. (Utiliser la base de code R pour effectuer des analyses [PLS Path modelling](#) (PLS))

Étape 1. Préparation des données

- Télécharger R et R studio (R studio pas obligatoire, mais FORTEMENT conseillé)
 - <https://rstudio.com/products/rstudio/>
- Installer la librairie "plsmp"
 - `install.packages("plsmp")`
 - OU voir: <https://github.com/gastonstat/plsmp#installation>
 - Si vous rencontrez des soucis pour l'installation de devtools - relancez l'installation en cliquant sur "non" pour l'installation des codes sources quand le popup vous le demande
- Télécharger les données: [R Code.zip](#)
- Charger les données .csv en R. Celles-ci contiennent pour chaque élève :
 - l'élément ludique qui lui a été affecté (colonne B)
 - une valeur pour chaque type de motivation initiale (avant d'utiliser l'environnement numérique, colonnes Q-W)
 - une valeur pour chaque dimension du profil de joueur (colonnes K-P)
 - une valeur pour chaque type de motivation finale (après avoir utilisé l'environnement numérique, colonnes X-AD)
 - quelques données sur l'activité des élèves (temps passé sur la plateforme, nombre de bonnes réponses etc. colonnes C-I)

Étape 2. Analyses PLS (PLS Path modelling)

- Comprendre l'analyse PLS (Chapitre 2) : [PLS Path Modeling with R](#)
- Lancez le script R fourni pour faire les analyses PLS mettant en évidence pour chaque élément de jeu les influences (vous devrez avoir un total de :
 - **du** profil Hexad de l'élève **sur** la variation de motivation (entre le début et la fin de l'usage de l'environnement numérique gamifié)
 - **de** la motivation initiale **sur** la variation de motivation (entre le début et la fin de l'usage de l'environnement numérique gamifié)
- Vous devez avoir un total de 24 matrices
 - 12 pour le profil Hexad
 - 6 pour les influences

- 6 pour les p.values (rappel les influences sont significatives si $p < 0.05$, voir $p < 0.1$ en fonction de la précision souhaitée).
- 12 pour le profil de motivation (selon la même répartition)
 - Décrivez/commentez deux des matrices de résultats de l'analyse PLS. // 2 points ?

Étape 3. Recommandations à partir des matrices PLS // 6 points

En Python, écrire le code qui permet de générer un vecteur d'affinité pour un élève (tableau ordonné des 6 éléments de jeu, du plus adapté au moins adapté, avec un score d'affinité pour chaque élément) pour son profil Hexad et un vecteur d'affinité pour son profil de motivation. Ce vecteur d'affinité permet de recommander un élément de jeu spécifique selon l'un des deux profils de l'élève (motivations initiales ou profil de joueur Hexad).

Justifier votre stratégie de construction de chaque vecteur d'affinité.

Deuxième partie : Algorithme d'adaptation (1h30) // 12 points

Objectif : Définir un algorithme qui permet de combiner les recommandations spécifiques à différents profils (issues de la première partie).

Il s'agit de spécifier un algorithme qui recommande un élément en prenant en compte à la fois les recommandations pour le profil de joueur Hexad d'un élève et les recommandations pour sa motivation initiale.

Étape 1. Réfléchissez à partir d'exemples. Sur papier, listez les différents cas à considérer pour construire l'algorithme. Pour vous aider, vous pouvez instancier les vecteurs d'affinité (pour Hexad et motivation initiale) pour quelques élèves (par exemple 3 ou 4 élèves différents).

Détailler dans votre CR ces éléments. // 3 points

Étape 2. Il s'agit maintenant de choisir une stratégie de compromis qui permette de recommander un élément de jeu approprié pour chaque élève (à partir des deux premiers vecteurs d'affinité). Expliquez votre stratégie en pseudo-code dans votre CR. // 3 points

Étape 3. Écrivez l'algorithme en Python. // 3 points

Étape 4. En utilisant les données de départ (fichier .csv), évaluez la pertinence de votre algorithme d'adaptation, selon différents indicateurs que vous choisirez à partir des données disponibles dans le fichier CSV (par exemple variation de motivation, temps passé sur la plateforme, nombre de bonnes réponses etc.).

Pour cela, faites d'abord tourner vos différents algorithmes sur le fichier CSV pour créer pour chaque algorithme un groupe d'élèves ayant un élément adapté et un groupe d'élèves ayant un élément non adapté. Vous pouvez ensuite réaliser des tests statistiques pour comparer les moyennes des différents indicateurs choisis pour les élèves ayant reçu un élément ludique qui

leur aurait été recommandé ou non (par exemple, vous pouvez utiliser le t-test selon la taille et la normalité de distribution de vos échantillons). N'oubliez pas de donner votre interprétation des résultats des tests statistiques dans votre compte-rendu. // 3 points