



HAL
open science

Aide à la détection de l'échange d'information entre étudiants dans les contrôles à distance

Gilles Burel, Khawla Ben Gaied, Ramla Ben Abdelkader, Roland Gautier

► **To cite this version:**

Gilles Burel, Khawla Ben Gaied, Ramla Ben Abdelkader, Roland Gautier. Aide à la détection de l'échange d'information entre étudiants dans les contrôles à distance. Journal sur l'enseignement des sciences et technologies de l'information et des systèmes, 2023, 22 (2023), pp.0001. 10.1051/j3ea/20230001 . hal-03367723v2

HAL Id: hal-03367723

<https://hal.univ-brest.fr/hal-03367723v2>

Submitted on 28 Jan 2023

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.



Distributed under a Creative Commons Attribution 4.0 International License

Aide à la détection de l'échange d'information entre étudiants dans les contrôles à distance

Gilles Burel, Khawla Ben Gaied, Ramla Ben Abdelkader, Roland Gautier
Université de Bretagne Occidentale
Lab-STICC, CNRS, UMR 6285, F-29200 Brest

Contact : Gilles.Burel@univ-brest.fr

J3eA, Journal sur l'enseignement des sciences et technologies de l'information et des systèmes.

Résumé

La situation sanitaire subie depuis mars 2020 a conduit à une utilisation massive des contrôles à distance, sous la forme des tests en ligne, de type QCM ou apparentés (questions à réponses numériques, graphiques, calculées, ...). Ces contrôles étant réalisés sans surveillance, il est tentant pour certains étudiants d'échanger des informations en cours de test. Cet échange d'information, lorsqu'il a lieu, laisse une empreinte sous la forme d'anomalies statistiques dans les réponses aux questions ainsi que dans l'horodatage. L'objectif de notre travail était de proposer et de tester des méthodes visant à détecter automatiquement ces anomalies, en vue d'attirer l'attention de l'enseignant sur les cas suspects. Dans notre domaine (EEA/STIC) le nombre limité d'étudiants (quelques dizaines) et de questions rend d'emblée inadaptés des modèles paramétriques complexes, tels que l'on peut en trouver dans la littérature, ces derniers étant conçus pour des tests à très grande échelle (avec des milliers de candidats) fournissant suffisamment de données pour ajuster le modèle. Nous décrivons la méthode développée, qui est bien adaptée à notre domaine (faible nombre d'étudiants et de questions), et nous l'illustrons à travers les résultats réels obtenus sur trois contrôles en ligne.

Mots clés : *Contrôles à distance, Détection de la fraude, Analyse statistique, Docimologie*

1 Introduction

L'utilisation de contrôles à distance dans l'enseignement supérieur a explosé depuis mars 2020, du fait d'une situation sanitaire réduisant les possibilités de contrôles en présentiel. C'est particulièrement vrai en ce qui concerne les contrôles continus. Dans ces circonstances il ne fait guère de doute que le travail de l'étudiant n'est pas toujours individuel et la presse s'en est largement faite l'écho [1]. Nous nous sommes intéressés à l'analyse a posteriori de résultats de contrôles à distance qui se sont déroulés au sein de notre université afin de mettre en

évidence quelques indicateurs qui pourraient être utiles à l’enseignant pour détecter l’échange d’information entre étudiants.

Des méthodes d’analyse statistique a posteriori des résultats du test, reposant sur des modèles statistiques élaborés, parfois complétées par des techniques de data mining, ont déjà été proposées. Un rapport de 87 pages [2] établi en 2018 par le régulateur Britannique (Office of Qualifications and Examinations Regulation) donne une excellente vue d’ensemble de ces techniques. Par nature, ce type de méthode nécessite de très grandes quantités de données pour pouvoir fournir des résultats significatifs et est donc adapté aux tests longs (en termes de nombres de questions) et à grande échelle (c’est à dire concernant des milliers de candidats). Ainsi, par exemple, Man et al. [3] utilisent un test comprenant 170 questions réalisé par deux groupes d’environ 1640 candidats (soit plus de 3200 candidats au total) et 50000 candidats en simulation. Le contexte considéré est généralement celui de tests papier, réalisés en présentiel sous surveillance. En conséquence, l’horodatage des questions n’existe pas, et le type de fraude que l’on cherche à détecter est davantage celui d’une fuite préalable au test (candidats ayant frauduleusement pu avoir connaissance a priori d’une partie ou de la totalité des questions).

Le contexte auquel nous nous sommes intéressés est assez différent et correspond à la réalité des tests que nous avons réalisés durant l’épisode Covid. Nous nous plaçons donc dans un contexte qui réunit simultanément les caractéristiques suivantes :

- Tests réalisés en distanciel, donc avec un horodatage disponible.
- Tests réalisés sur de petits groupes d’étudiants (une trentaine).
- Tests comprenant au maximum quelques dizaines de questions.
- Triche se caractérisant par un échange d’information entre certains étudiants en cours de test et non par une connaissance préalable des questions.

Le nombre limité de questions est très fréquent dans notre domaine EEA/STIC (et se retrouve très probablement dans d’autres domaines voisins, telles que les mathématiques). En effet, les questions nécessitant une part de réflexion (il ne s’agit pas de « par cœur »), l’étudiant a besoin de quelques minutes pour élaborer la réponse : on ne peut donc pas poser un très grand nombre de questions sur la durée du test. En conséquence, nous essayons d’être concrets, afin de fournir des indices que l’enseignant peut facilement comprendre et interpréter. En cela, on se place davantage dans une optique d’aide (attirer l’attention de l’enseignant sur des cas douteux et lui fournir des éléments d’appréciation) que dans la recherche de méthodes entièrement automatiques. De plus, le nombre limité d’étudiants et de questions rend d’emblée inadaptés des modèles paramétriques complexes.

Un argument supplémentaire en faveur d’une telle approche est que l’on n’a pas en général la « réalité terrain » (c’est à dire la connaissance exacte des tricheurs et des questions sur lesquelles ils ont triché) qui permettrait d’évaluer, de comparer, des méthodes. Il nous a donc semblé plus pertinent de fournir à l’enseignant des avertissements et indicateurs simples, plutôt que des sorties d’algorithmes complexes qu’il aura du mal à interpréter sans une connaissance approfondie du modèle paramétrique sous-jacent.

Les résultats anonymisés de trois contrôles en ligne ont été utilisés pour tester les méthodes proposées. Ces contrôles avaient été réalisés sur la plateforme Moodle [4], très utilisée au sein

de notre université et, à notre connaissance, dans l'enseignement supérieur en général. Nous avons créé une base de données, dans laquelle nous avons indiqué, pour chaque couple (étudiant, question), la réponse donnée, la bonne réponse, l'ordre de la question et l'horodatage de la réponse.

L'article est organisé comme suit : après un aperçu du contexte bibliographique (section 2), nous décrivons la manière dont nous avons collecté et mis en forme les données (section 3). Dans la section 4 nous décrivons les méthodes développées pour analyser les données, en commençant par l'analyse de l'horodatage, qui est de nature très différente selon que le test est à navigation libre (sous-section 4.1) ou séquentielle (sous-section 4.2), et en la complétant par l'analyse des réponses aux questions (sous-section 4.3).

Les programmes que nous avons développés sont librement accessibles sous licence GNU GPLv3 pour ceux qui souhaitent les utiliser, les adapter ou les améliorer [17], étant entendu qu'il s'agit de programmes de test sans interface dédiée et non d'un logiciel à vocation commerciale.

2 Etude bibliographique

Dans le cadre de la prévention/détection de la triche dans les examens à distance, il est important de bien distinguer les solutions basées sur la télésurveillance des solutions basées sur une analyse a posteriori des résultats du test.

2.1 La télésurveillance

Selon la presse [5][6], plusieurs universités et établissements d'enseignements supérieurs se sont montrés intéressés par des solutions de télésurveillance : de telles solutions (payantes) sont proposées par des sociétés privées avec pour objectif d'empêcher ou tout au moins de détecter en temps réel la fraude dans les tests à distance. Parmi les sociétés qui bénéficient d'une bonne couverture médiatique, on trouve TestWe [7], Proctorexam [8] Managemex [9] et Smowl [10]. Ces sociétés proposent des solutions qui utilisent des techniques telles que :

- Repérage automatique et en temps réel des anomalies pouvant survenir pendant la session d'évaluation grâce notamment à une analyse visuelle et/ou auditive de l'environnement des candidats : il peut s'agir de double caméra (PC et smartphone) permettant d'avoir une surveillance globale de la pièce dans laquelle le candidat compose, de détection automatique de bruits suspects, etc.
- Mise en œuvre d'algorithmes de reconnaissance automatique des visages pour vérifier l'identité de l'utilisateur en ligne.
- Suivi automatique du regard de l'étudiant (eye-tracking) pour la détection de comportements suspects.

Bien que potentiellement efficaces, ces solutions intrusives ne peuvent être utilisées dans n'importe quel contexte et soulèvent des inquiétudes [11] en termes de vie privée (télésurveillance de la résidence privée de l'étudiant pendant l'épreuve). Du point de vue légal l'utilisation de ces solutions nécessite de grandes précautions en vertu de l'article 9 du règlement général sur la

protection de données, notamment parce qu'elles font appel à du traitement de données biométriques (identifiant d'une manière unique une personne physique) : reconnaissance du visage ou de la voix.

Des solutions de surveillance moins intrusives existent néanmoins, comme par exemple le système Evalbox [12]. Il s'agit d'un système de QCM en ligne qui propose des fonctionnalités telles que l'envoi à l'enseignant, en temps réel, d'un message d'alerte quand l'étudiant dépasse un délai en fonction de chaque question, ou quand il ouvre une autre fenêtre de navigation.

2.2 L'analyse a posteriori des résultats du test

Une gamme de techniques statistiques a été développée pour étudier les anomalies dans les résultats de tests et d'évaluations à enjeux élevés [2]. Celles-ci impliquent généralement un test statistique basé sur la différence entre les scores de test des candidats et ceux attendus des modèles théoriques.

Beaucoup de méthodes utilisent de manière plus ou moins explicite le modèle de Rasch [13] : il modélise la relation fondamentale postulée entre le trait latent (compétence, capacité, aptitude, etc.) de l'individu (étudiant) et sa probabilité de réussir correctement à un item (question). Il analyse les réponses des candidats pour détecter d'éventuelles situations problématiques. Ce modèle est paramétrique : il permet d'exprimer sur une échelle la difficulté relative des questions et la compétence relative des candidats (basé sur la probabilité de la réussite d'un candidat avec un niveau de performance donnée à une question de difficulté donnée). Il a été utilisé pour les épreuves à choix multiple du test d'évaluation de français [14].

Une caractéristique très fréquente des méthodes proposées est le fait qu'elles se placent d'emblée dans un cadre bayésien (voir [15], par exemple). Le modèle bayésien peut être réduit (basé sur les réponses aux items) ou complet (basé sur les réponses aux items et les temps de réponse). Ce modèle implique des connaissances préalables concernant les candidats et les items. L'un des défauts du cadre bayésien est que la petite taille des données disponibles peut conduire à une mauvaise estimation des probabilités, ce qui entraîne par voie de conséquence des classifications inexactes. On note aussi que l'essentiel des travaux dans ce domaine se concentre sur la modélisation d'un type spécifique de triche, à savoir la pré-connaissance des questions par certains candidats, due à une fuite d'information avant le contrôle.

Concernant la modélisation des temps de réponse, on trouve beaucoup moins d'études dans la littérature. Un modèle lognormal intéressant mais assez complexe a été proposé par van der Linden [16] et fait souvent référence. Dans notre étude nous avons préféré proposer un critère plus simple afin d'éviter d'avoir besoin d'une grande quantité de données nécessaires à l'estimation des paramètres du modèle.

L'utilisation de techniques d'apprentissage a également été proposée dans ce contexte [3]. L'apprentissage permet d'agréger les sorties de différentes méthodes d'analyse statistique, ce qui laisse espérer un gain de performance, mais nécessite d'immenses quantités de données. L'apprentissage peut être supervisé ou non supervisé.

Dans le cas non-supervisé, il s'agit d'algorithmes de clustering qui regroupent des observations similaires : K-moyennes, modèles de mélanges normaux multivariés, modèles d'auto-organisation. Dans ce dernier cas, les clusters peuvent être représentés sur une grille bidimensionnelle qui montre quels clusters sont similaires les uns aux autres. Il s'agit d'un outil utile pour la visualisation des données et la validation des clusters détectés.

Dans le cas supervisé, il est nécessaire que la machine apprenne à partir d'une grande base de données dans laquelle une investigation a permis d'identifier clairement les tricheurs. Des méthodes telles que SVM (support vector machine) ou RF (Random Forest) ont été utilisées dans ce contexte [3].

Il est important de mentionner que si des méthodes très élaborées de ce type, combinant modèles statistiques et apprentissage, sont utiles pour le traitement de tests à très grande échelle (énormément de candidats, beaucoup de questions), leur utilisation dans le cadre des contrôles à distance tels que nous les avons mis en œuvre dans notre domaine EEA/STIC n'est pas réaliste car la quantité de données disponibles est bien plus faible. Un modèle complexe ne conduit pas nécessairement à une détection plus précise des valeurs aberrantes et peut au contraire voir ses performances fortement dégradées par le manque de données. Il s'agit typiquement d'un problème de surdimensionnement du modèle par rapport aux données disponibles.

Notre étude s'est donc plutôt orientée sur la proposition d'une solution simple qui ne demande ni connaissances a priori sur les candidats (nécessaires pour estimer les probabilités a priori dans le cadre bayésien) ni grandes quantités de données. L'objectif est d'alerter l'enseignant sur les fraudes potentielles en lui présentant des indicateurs simples qu'il peut facilement comprendre et interpréter. Dans la suite nous essayons d'être aussi concrets que possible, en illustrant la démarche par des données réelles et en évitant de passer sous silence la problématique de la récupération des données nécessaires à l'analyse. En effet, comme nous allons le voir dans la section suivante, la récupération des données d'horodatage et la remise en ordre des informations est déjà, en soi, une difficulté.

3 Collection et mise en forme des données

La plateforme Moodle [4] permet d'exporter les réponses de l'ensemble des étudiants dans des fichiers au format csv. Ce format très simple, constitué de champs séparés par un délimiteur, présente l'avantage d'être compatible avec les tableurs (LibreOffice, Excel, ...) et avec les logiciels

Numéro d'étudiant	Réponse 1	Réponse correcte 1	Réponse 2	Réponse correcte 2	Réponse 3	Réponse correcte 3
1	Faux	Faux	4 Mo	4 Mo	6 Mo	2 Mo
2	Faux	Faux	-	4 Mo	Cela dépend.	2 Mo
3	Faux	Faux	-	4 Mo	Cela dépend.	2 Mo
4	Faux	Faux	4 Mo	4 Mo	6 Mo	2 Mo
5	Faux	Faux	-	4 Mo	6 Mo	2 Mo
6	Faux	Faux	12 Mo	4 Mo	2 Mo	2 Mo
7	Faux	Faux	12 Mo	4 Mo	6 Mo	2 Mo
8	Faux	Faux	4 Mo	4 Mo	2 Mo	2 Mo
9	Faux	Faux	4 Mo	4 Mo	2 Mo	2 Mo
10	Faux	Faux	-	4 Mo	2 Mo	2 Mo

FIGURE 1 – Extrait du tableau des réponses

de calcul (Octave, Matlab, ...). La figure 1 montre un extrait d'un fichier de réponses. Chaque ligne correspond à un étudiant et chaque colonne à une question. Les questions sont rangées dans l'ordre original et non dans l'ordre dans lequel elles ont été présentées à chaque étudiant. On a des groupes de deux colonnes contenant la réponse apportée par l'étudiant et la réponse correcte. Les réponses peuvent être des valeurs numériques, des booléens, des mots, des phrases, ... Dans le cas général on considère donc que le contenu de chaque case est une chaîne de caractères.

Moodle ne permet pas, par contre, d'exporter une information qui nous a semblée très pertinente, à savoir l'horodatage. L'information est cependant présente : en effet, la fonctionnalité «Relecture du test» permet à l'enseignant d'afficher, pour un étudiant donné, une page très détaillée donnant les réponses de l'étudiant et les instants de réponse. Nous avons donc dû trouver une alternative : enregistrer séparément les réponses des étudiants et écrire un programme en langage PERL qui traite l'ensemble des fichiers de réponses et génère le tableau d'horodatage. L'enregistrement des réponses des étudiants doit être fait par l'enseignant, ce qui prend environ 3 minutes pour une trentaine d'étudiants. Nous avons choisi de les enregistrer au format texte. Un extrait d'un tel fichier est montré sur la figure 2. Ensuite le traitement des fichiers par le programme PERL se fait en quelques secondes.

Le langage PERL est particulièrement adapté à ce type de traitement de fichiers texte. Par exemple (en simplifiant), si les questions elles-mêmes ne contiennent pas le mot «Enregistré», la détection et l'extraction de l'horodatage peut se faire en quelques lignes par :

```
if ($ligne =~ /Enregistré/)
{ $ligne =~ /(\d)+:(\d)+/;
  my $heure = $$;
  print $out $heure."";
}
```

où \$ligne désigne une ligne du fichier texte et \$out est le pointeur sur le fichier de sortie. Bien entendu, pour une utilisation à grande échelle, il serait nécessaire de complexifier un peu le programme pour le rendre plus robuste ou bien de traiter des enregistrements au format html

```

    Texte de la question
Une image couleur RGB standard, de 800 pixels sur 1000, comprimée d'un
facteur 12, nécessite environ:
Veuillez choisir une réponse :
a.
67 Ko
b.
2 Mo
c.
3 Mo
d.
200 Ko
/
/
    Feedback
La réponse correcte est : 200 Ko
    Commentaires
Faire un commentaire ou modifier une note
<https://moodlescienze.univ-brest.fr/moodle/mod/quiz/comment.php?attempt=*****&slot=25>
    Historique des réponses
Étape  Heure  Action  État  Notes
1
<https://moodlescienze.univ-brest.fr/moodle/mod/quiz/reviewquestion.php?attempt=*****&slot=25&step=0>
9 nov. 20, 15:50  Commencé  Pas encore répondu
2
<https://moodlescienze.univ-brest.fr/moodle/mod/quiz/reviewquestion.php?attempt=*****&slot=25&step=1>
9 nov. 20, 16:03  Enregistré : 200 Ko  Réponse enregistrée
3 9 nov. 20, 16:42  Tentative terminée  Correct  2,00
    Question 6
Correct
Note de 1,00 sur 1,00
Non marquéeMarquer la question

```

FIGURE 2 – Extrait du fichier de réponses d'un étudiant

et non au format texte.

À l'issue de ce traitement, on obtient un fichier d'horodatage. Un extrait d'un tel fichier est montré sur la figure 3. Chaque ligne correspond à un étudiant et chaque colonne à une question (à l'exception de la première colonne qui est l'heure de démarrage du test par l'étudiant). Ici, contrairement au fichier de réponses, les questions sont rangées dans l'ordre dans lequel elles ont été présentées à chaque étudiant et non dans l'ordre original. Lorsque l'étudiant n'a pas répondu à une question, on n'a pas d'horodatage et donc nous avons choisi d'indiquer 0 (cela permet d'identifier une non-réponse car en principe un contrôle ne se fait pas à minuit).

L'horodatage indiqué sur la figure 3 correspond à un test séquentiel, c'est à dire que les étudiants ne pouvaient pas revenir en arrière. La figure 4 montre un exemple d'horodatage dans

	A	B	C	D	E	F	G
1	15:50	00:00	15:59	16:00	16:02	16:03	16:04
2	15:52	15:53	15:54	00:00	16:13	16:16	16:17
3	15:52	00:00	16:13	16:17	16:18	16:18	16:19
4	15:53	15:56	15:57	16:00	16:01	16:03	16:04
5	15:51	15:52	15:54	15:57	16:02	16:04	16:06
6	15:50	15:55	15:59	16:03	16:04	16:06	16:06
7	15:50	15:58	15:59	00:00	16:01	16:05	16:06
8	15:53	15:55	15:55	15:55	16:00	16:06	16:07
9	15:50	15:53	15:54	15:56	16:03	16:04	16:07
10	15:50	15:57	15:59	16:03	16:06	16:16	16:19

FIGURE 3 – Extrait de l'horodatage du CC3

	A	B	C	D	E	F	G
1	10:00	10:21	10:44	10:45	10:38	10:24	10:41
2	10:00	10:32	10:57	10:45	10:38	10:58	10:59
3	10:00	10:47	10:09	10:10	10:53	10:55	10:45
4	10:00	10:06	10:10	10:17	10:19	10:24	10:26
5	10:00	10:03	10:09	00:00	10:22	10:22	10:25
6	10:01	10:07	10:13	10:14	10:15	00:00	10:26
7	10:00	10:03	10:10	00:00	10:52	10:31	10:44
8	10:00	10:04	11:00	10:05	10:10	10:15	10:17
9	10:00	10:05	10:52	10:13	10:13	10:17	10:47
10	10:00	10:03	10:05	10:09	10:13	10:18	10:24

FIGURE 4 – Extrait de l’horodatage du CC1

le cas d’un test à navigation libre. Ici on peut remarquer que certains étudiants répondent aux questions dans l’ordre dans lequel elles leur sont présentées, alors que d’autres effectuent de nombreux retours en arrière. Un comportement extrêmement erratique de l’horodatage peut correspondre à un étudiant qui répond aux questions dans l’ordre dans lequel il arrive à obtenir des réponses fournies par ses collègues.

Nous avons vu que le tableau des réponses est rangé dans l’ordre original des questions, alors qu’en ce qui concerne le tableau d’horodatage, sur chaque ligne l’ordre est celui dans lequel les questions ont été présentées à l’étudiant. Pour certains traitements, il va être nécessaire de remettre les données dans l’ordre original. Par exemple, imaginons que l’on calcule le temps de réponse de chaque étudiant à chaque question. Pour pouvoir comparer ces temps de réponse, il est évidemment nécessaire de les remettre dans l’ordre original des questions, faute de quoi on comparerait des temps de réponses correspondant à des questions différentes. Pour cela, nous avons généré un autre tableau, qui indique, pour chaque étudiant, l’ordre dans lequel les questions lui ont été présentées. Là aussi, le problème est que Moodle ne permet pas d’exporter cette information. C’est donc le programme PERL que nous avons écrit qui permet (en plus de l’extraction de l’horodatage) de retrouver l’ordre original des questions. Cela se fait par comparaison de chaînes de caractères. Pour chaque fichier correspondant aux réponses d’un étudiant, le programme isole la chaîne de caractères correspondant à chaque question, et par comparaison avec un fichier texte contenant l’énoncé des questions dans l’ordre original, il retrouve le numéro de chaque question. Un extrait d’un tableau fourni par ce programme est donné sur la figure 5. Ici par exemple l’étudiant numéro 1 s’est vu présenter d’abord la question 12, puis la question 5, etc.

Pour résumer, notre analyse va donc se baser sur trois tableaux au format csv :

- reponses.csv
- horodatage.csv
- ordre.csv

Nous disposons des résultats de 3 contrôles continus qui avaient été réalisés en ligne :

- CC1 : Contrôle à navigation libre comprenant 19 questions. Les questions sont de plusieurs types (QCM, Numériques, ...), à l’exception des questions calculées.

	A	B	C	D	E	F	G
1	12	5	3	7	25	19	1
2	18	22	13	14	19	1	12
3	23	14	21	8	6	18	24
4	17	25	14	9	3	20	4
5	9	22	3	12	19	20	15
6	11	3	12	8	21	7	10
7	23	6	17	1	7	10	24
8	25	16	10	22	15	19	6
9	13	19	7	24	22	15	10
10	3	1	22	25	5	9	26

FIGURE 5 – Ordre des questions du CC3

- CC2 : Contrôle séquentiel comprenant 25 questions. Les questions sont de plusieurs types (QCM, Numériques, Calculées, ...).
- CC3 : Contrôle séquentiel comprenant 26 questions. Les questions sont uniquement des QCM.

Les contrôles étaient indépendants, portaient sur des sujets différents (mais néanmoins tous dans le domaine EEA/STIC) et ont été réalisés à plusieurs semaines d'intervalle. Cependant, c'est le même groupe de 35 étudiants de Master qui a passé les contrôles CC2 et CC3, ce qui, nous le verrons, permet de faire des recoupements intéressants entre les résultats d'analyse. Le CC1 a été réalisé avec un groupe de Licence comprenant 30 étudiants.

On rappelle que dans un contrôle donné les étudiants se voient présenter les mêmes questions, mais dans un ordre différent d'un étudiant à l'autre. De plus, pour les questions calculées (présentes dans le CC2) les valeurs numériques données dans la question sont aléatoires et en conséquence la bonne réponse attendue varie d'un étudiant à l'autre.

4 Analyse des données

Dans la suite, nous noterons Q le nombre de questions et nous utiliserons les indices q et p pour numéroter les questions : q désigne le numéro original de la question et p le numéro de la question vu de l'étudiant. Enfin, on notera r le rang de réponse. Pour un contrôle séquentiel, on a $p(r) = r$ puisque l'étudiant doit répondre aux questions dans l'ordre dans lequel on les lui présente. Par contre, pour un contrôle à navigation libre, l'étudiant peut répondre dans l'ordre qui lui convient : dans ce cas, la fonction $p(r)$ apporte, on le verra, des informations intéressantes sur le comportement de l'étudiant.

4.1 Analyse de l'horodatage dans le cas d'un contrôle à navigation libre

Dans le cas d'un contrôle à navigation libre, on distingue deux types de comportements :

- Certains étudiants parcourent les questions de manière parfaitement linéaire ($p(r) = r$), sans retour en arrière ou quasi-linéaire, avec peu de retours en arrière (Fig. 6)

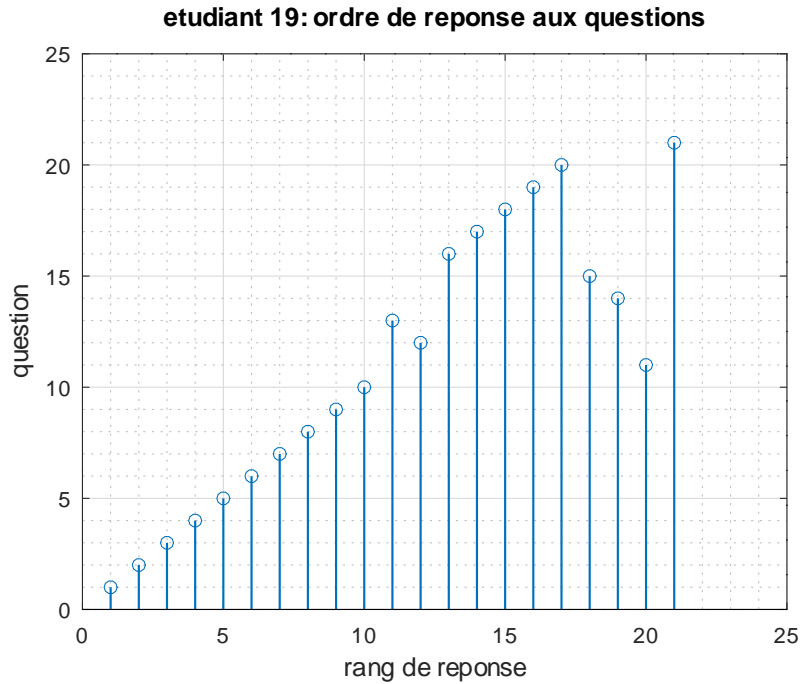


FIGURE 6 – CC1, étudiant 19 : fonction $p(r)$

— D’autres ont un parcours très erratique avec d’incessants retours en arrière (Fig. 7). La fonction $p(r)$ a dans ce cas un aspect aléatoire.

Par exemple, après avoir répondu à la question 5, l’étudiant 11 (Fig. 7) a passé plusieurs questions pour aller répondre à la question 16, puis est revenu en arrière pour répondre à la question 11, etc.

Autant quelques retours en arrière peuvent se comprendre dans le cas d’un étudiant qui préfère passer quelques questions difficiles pour y revenir vers la fin du test, autant un parcours totalement erratique peut être le signe d’un étudiant qui répond aux questions dans l’ordre dans lequel il arrive à obtenir des réponses fournies par des collègues.

Dans le cas extrême d’un étudiant qui ne cherche aucune réponse par lui-même, on peut considérer que son parcours est totalement aléatoire puisque le ou les autres étudiants qui lui fournissent des réponses se sont vus présenter les questions dans un ordre différent du sien (et donc aléatoire, de son point de vue). Définissons le saut comme :

$$s(r) = |p(r + 1) - p(r)| \tag{1}$$

Dans le cas d’un étudiant qui répond dans un ordre aléatoire, $p(r)$ suit, en première approximation, une loi uniforme. En utilisant un résultat donné en annexe, on peut dans ce cas calculer le saut moyen (c’est-à-dire la moyenne de $s(r)$) :

$$\bar{s} = \frac{Q + 1}{3} \tag{2}$$

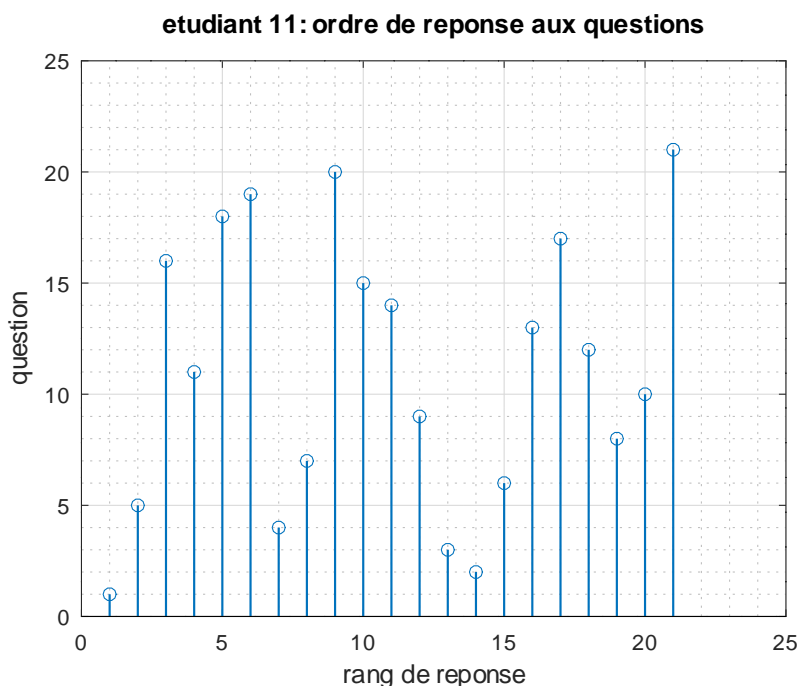


FIGURE 7 – CC1, étudiant 11 : fonction $p(r)$

Nous avons donc calculé, pour chaque étudiant, l’amplitude moyenne des sauts. Pour un étudiant qui a un parcours totalement linéaire, cette valeur vaut 1, puisque le numéro de question est incrémenté d’une unité à chaque étape alors que pour un étudiant qui a un parcours erratique cette valeur est élevée. Nous normalisons ces valeurs en les divisant par ce que l’on obtiendrait avec un parcours totalement aléatoire, c’est à dire par \bar{s} . Les valeurs normalisées obtenues sont représentées sur la figure 8 : on peut remarquer le cas de l’étudiant 2, dont la valeur normalisée proche de 1 laisse penser qu’il ne cherche aucune réponse par lui-même. Nous avons placé un seuil à 0.6 (soit 60% du cas totalement aléatoire) pour détecter les étudiants dont le parcours est suspect. On a donc ici un groupe de 6 étudiants (numéros 1, 2, 7, 11, 12, 14) dont le comportement peut être considéré comme suspect.

Le choix de seuil provient du fait que si l’on renumérote les étudiants dans l’ordre des sauts moyens (Fig 9) on voit apparaître une nette rupture entre le 24^e et le 25^e rang.

Nous recherchons également les similitudes entre horodatages. Pour cela, nous recherchons, pour chaque couple d’étudiants (i,j), la valeur médiane de l’écart entre les instants de réponse aux différentes questions :

$$m_{ij} = med_q (|h_i(q) - h_j(q)|) \quad (3)$$

où q désigne le numéro de question (considéré dans l’ordre original) et $h_i(q)$ l’instant auquel l’étudiant i a répondu à la question q . La valeur médiane est calculée en utilisant toutes les questions auxquelles les deux étudiants i et j ont répondu (lorsque l’un des deux n’a pas répondu, il manque un horodatage).

Les questions étant présentées aux étudiants dans un ordre aléatoire pendant la durée T

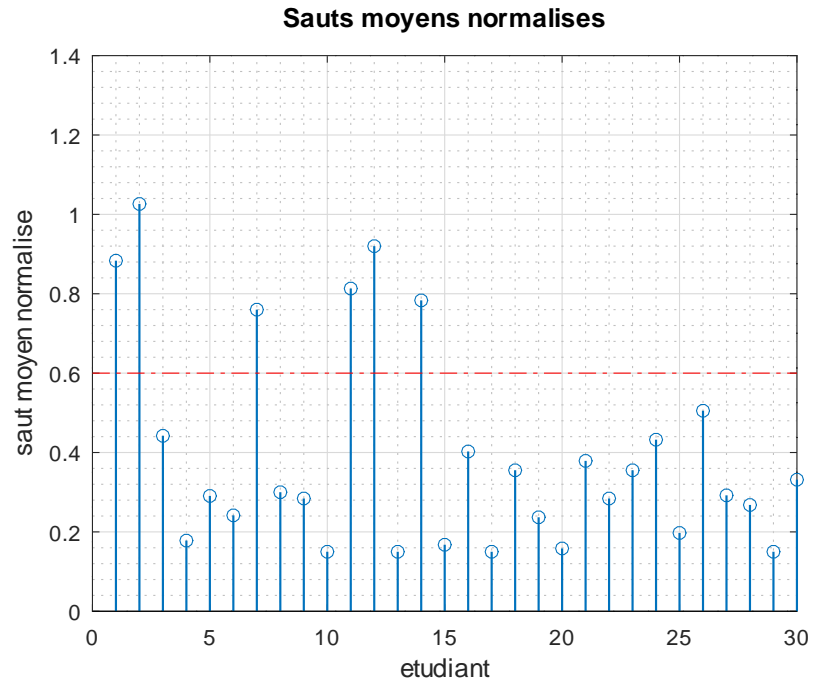


FIGURE 8 – CC1 : amplitude moyenne des sauts

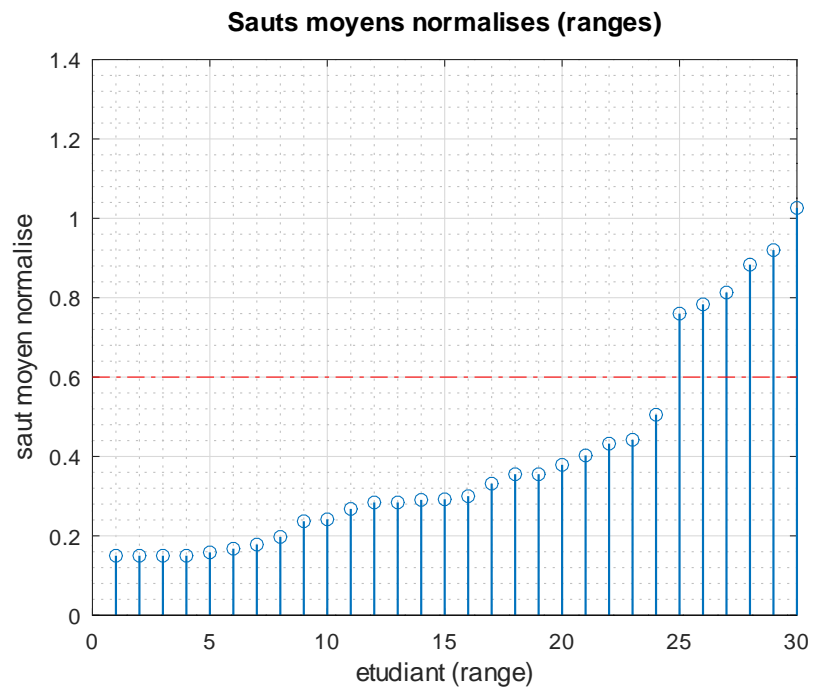


FIGURE 9 – CC1 : amplitude moyenne des sauts après renumérotation

du test, les instants de réponse de deux étudiants à une question donnée devraient donc suivre des lois uniformes indépendantes. Un calcul élémentaire basé sur la théorie des probabilités (cf. annexe) montre que la valeur médiane de l'écart entre les instants de réponse devrait être :

$$\check{m} = \left(1 - \frac{1}{\sqrt{2}}\right) T \quad (4)$$

Ainsi, pour un test d'une durée de 60 mn cette valeur théorique est d'environ 18 mn. Nous normalisons donc en divisant par cette valeur :

$$n_{ij} = \frac{m_{ij}}{\check{m}} \quad (5)$$

Nous obtenons une matrice symétrique de taille 30x30 dans laquelle l'élément situé ligne i colonne j représente n_{ij} (Fig. 10). Si la valeur normalisée est significativement inférieure à 1 on peut considérer qu'il y a eu échange d'information entre les étudiants. Nous avons mis un seuil à 0.25 (soit un m_{ij} inférieur à 25% de \check{m}) pour attirer l'attention de l'enseignant sur les cas les plus suspects. La figure 11 montre, en noir, les valeurs de la figure 10 qui sont en dessous de ce seuil.

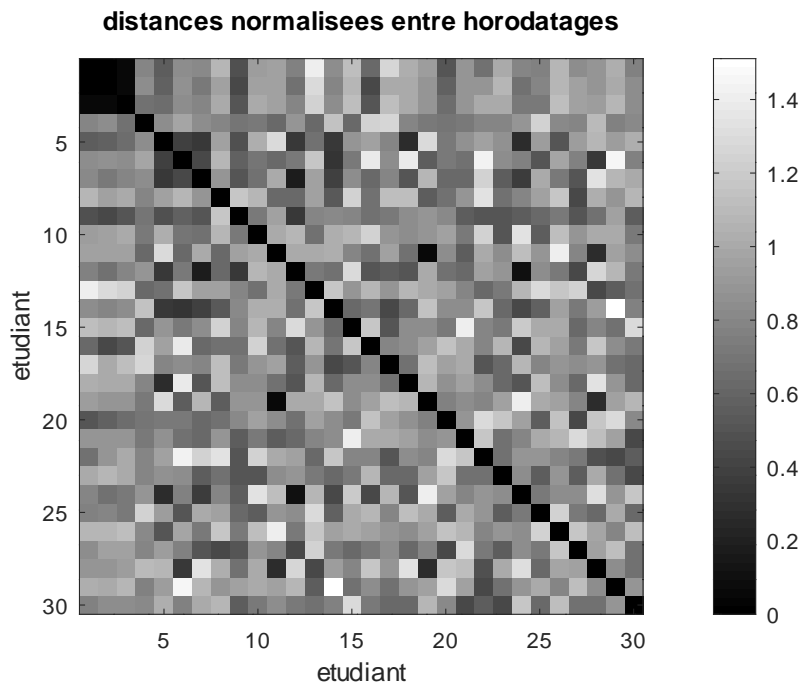


FIGURE 10 – CC1 : distances entre horodatages

Les éléments non-diagonaux qui apparaissent en noir indiquent des suspicions d'échange d'information entre étudiants (par exemple, ici, entre les étudiants 1, 2 et 3, 7 et 12, 11 et 19, 12 et 24). Le programme d'analyse que nous avons développé recherche automatiquement les groupes ainsi formés et présente les résultats à l'enseignant sous forme graphique (Fig. 12).

Cette figure fait apparaître 3 groupes d'étudiants au sein desquels on peut suspecter de

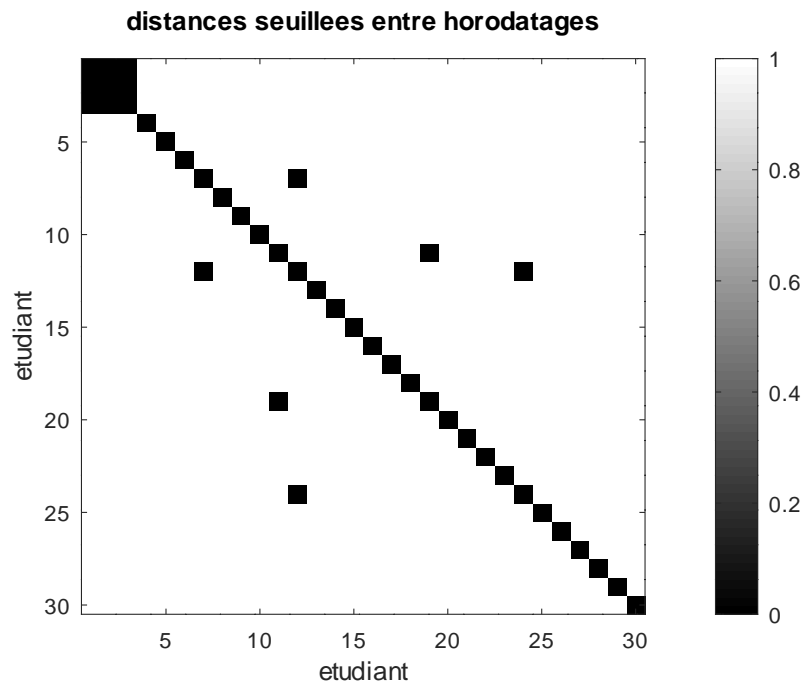


FIGURE 11 – CC1 : distances seuillées entre horodatages

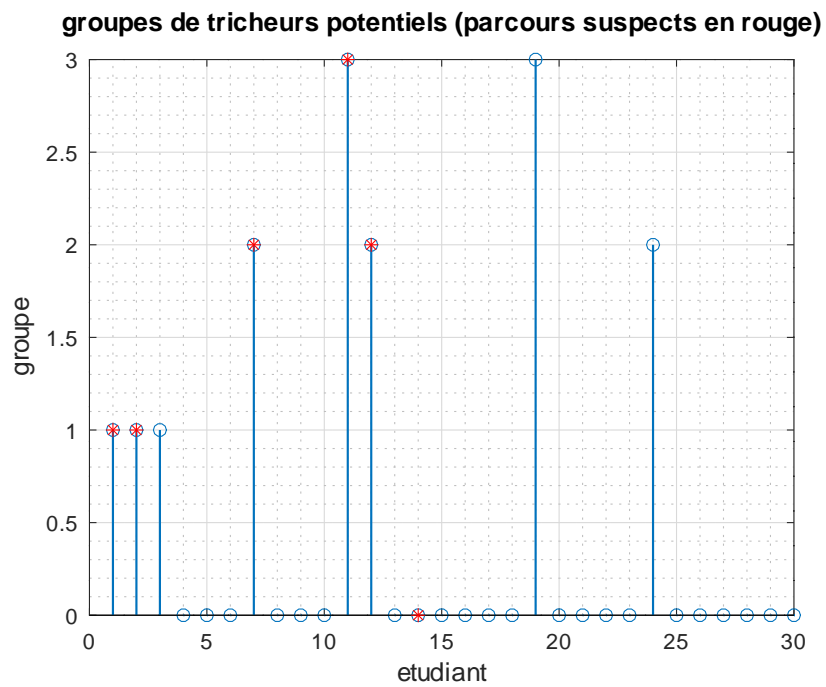


FIGURE 12 – CC1 : groupes détectés par l'analyse de l'horodatage

l'échange d'information : (1,2,3) (7,12,24) et (11,19). De plus, au sein de chaque groupe, lorsqu'un étudiant a eu un parcours nettement plus linéaire que les autres on peut faire l'hypothèse qu'il a fourni l'essentiel de l'information que les autres ont recopiée. Ainsi, dans le groupe (1,2,3) les sauts moyens normalisés sont respectivement (0.9, 1.0 et 0.4) : on peut donc suspecter que c'est l'étudiant 3 qui a fourni l'essentiel du travail, les étudiants 1 et 2 se contentant de recopier les réponses qu'il leur a fourni. Ceci se voit sur la figure car les étudiants qui avaient été détectés sur la base des sauts moyens sont indiqués en rouge. De même, dans le groupe (7,12,24), la figure nous suggère que 24 est la source d'information et 7 et 12 les copieurs. Enfin, dans le groupe (11,19), c'est probablement 11 qui copie sur 19.

Ceci est confirmé par l'affichage de l'ordre des réponses aux questions de l'étudiant 19 (Fig. 6) qui est plus cohérent que celui des l'étudiant 11 (Fig. 7). Le même type d'analyse peut être fait pour les autres groupes.

On peut noter que l'étudiant 14 avait été détecté sur la base des sauts moyens (c'est pourquoi il apparaît en rouge sur la figure 12) mais n'a pas franchi le seuil de similitudes. On peut faire l'hypothèse qu'il a obtenu l'information de manière plus diffuse, à travers une multitude de sources.

Afin de recouper ces détections avec d'autres informations, nous avons recherché la formation d'origine des étudiants concernés (c'est-à-dire l'établissement et le cursus qui les accueillait l'année précédente). Les étudiants (1,2,3) provenaient de la même formation et ils étaient les trois seuls dans ce cas, ce qui constitue un élément prouvant qu'ils se connaissaient très bien et qui tend donc à confirmer la validité de la détection. Nous avons également regardé les binômes de TP. Le groupe (11,19) travaillait en binôme en TP, ce qui là aussi confirme une forte proximité renforçant la vraisemblance d'un échange d'information. Enfin, dans le groupe (7,12,24) les étudiants 7 et 24 travaillaient en binôme en TP.

4.2 Analyse basée sur l'horodatage dans le cas d'un contrôle séquentiel

Dans le cas d'un contrôle séquentiel, pour faciliter le traitement, nous commençons par interpoler l'horodatage lorsqu'une donnée est manquante. En effet, nous avons trois cas de figure :

- Si l'étudiant a répondu à la question et à la précédente, on peut calculer exactement la durée de réponse par différence d'horodatage.
- Si l'étudiant n'a pas répondu à la question, on estime le temps passé à la lecture de la question et à une brève réflexion conduisant à la non-réponse à 20 secondes (soit 1/3 puisque l'unité de temps pour Moodle est la minute).
- Si l'étudiant a répondu à la question mais pas à la question précédente on estime le temps de réponse par différence entre l'horodatage de la question courante et l'horodatage estimé de la question précédente.

Dans le cas d'un contrôle séquentiel, les étudiants n'étant pas autorisés à revenir en arrière, le comportement naturel devrait être de répondre aux questions sans perdre de temps. Or, on constate que certains étudiants ont un comportement très suspect. Sur la figure 13 on voit que sur un contrôle d'une durée d'une heure, comprenant 26 questions, il passe une dizaine de

minutes sur la première question qui lui est présentée, puis encore une dizaine de minutes sur la 2^e question, accumulant de plus en plus de retard. Une douzaine de minutes avant la fin du contrôle il se met à répondre très rapidement à un grand nombre de questions. On peut faire l'hypothèse que le retard accumulé est dû au fait que l'étudiant attend de recevoir des réponses de la part de ses collègues (qui, on le rappelle, ne se voient pas proposer les questions dans le même ordre), puis, à mesure que le temps avance, de plus en plus de réponses deviennent disponibles, ce qui explique l'accélération en fin de test.

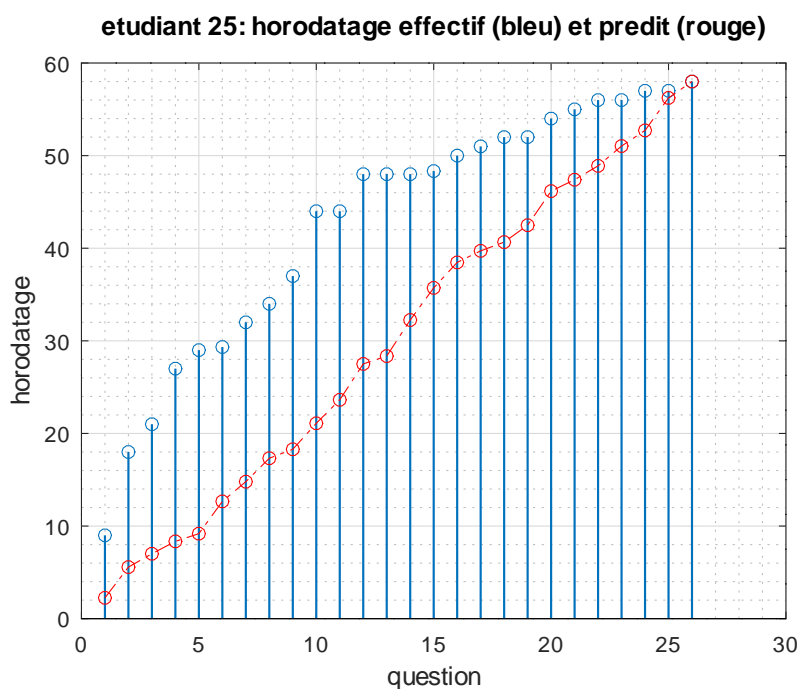


FIGURE 13 – CC3 : horodatage de l'étudiant 25

Nous expliquerons plus loin comment nous estimons la durée moyenne de réponse à chaque question. En utilisant ces durées moyennes estimées et l'ordre dans lequel les questions ont été présentées à l'étudiant, nous pouvons prédire ce qu'aurait été son horodatage s'il avait consacré à chaque question une durée correspondant à la durée moyenne que l'ensemble des étudiants y a consacré : c'est l'horodatage indiqué en rouge sur la figure. On peut ensuite calculer, pour chaque question, le retard pris par l'étudiant par rapport à l'horodatage prédit (différence entre les points bleus et rouges). Pour l'étudiant 25, sur le CC3 (Fig. 13), le retard maximal atteint est de 23 minutes (soit plus du tiers de la durée totale du contrôle).

La figure 14 montre le retard maximal atteint par chaque étudiant sur le CC2. Nous avons mis un seuil à 20% de la durée totale du contrôle, soit 12 minutes, au dessus duquel le comportement de l'étudiant est considéré comme suspect (la figure 15 montre que ce seuil correspond à une rupture visible, bien qu'elle ne soit pas très franche).

Les figures 16 et 17 montrent les résultats pour le CC3. Ici on peut noter qu'il y a davantage d'étudiants qui dépassent le seuil. Une explication vraisemblable est que le CC3 étant un QCM, il est plus tentant et plus facile pour les étudiants d'échanger de l'information (alors que dans

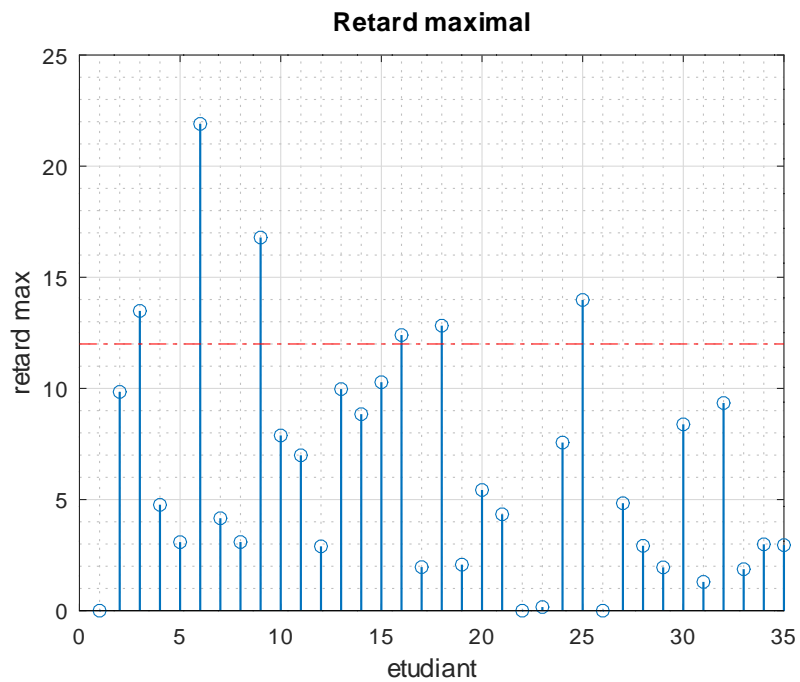


FIGURE 14 – CC2 : retard maximal

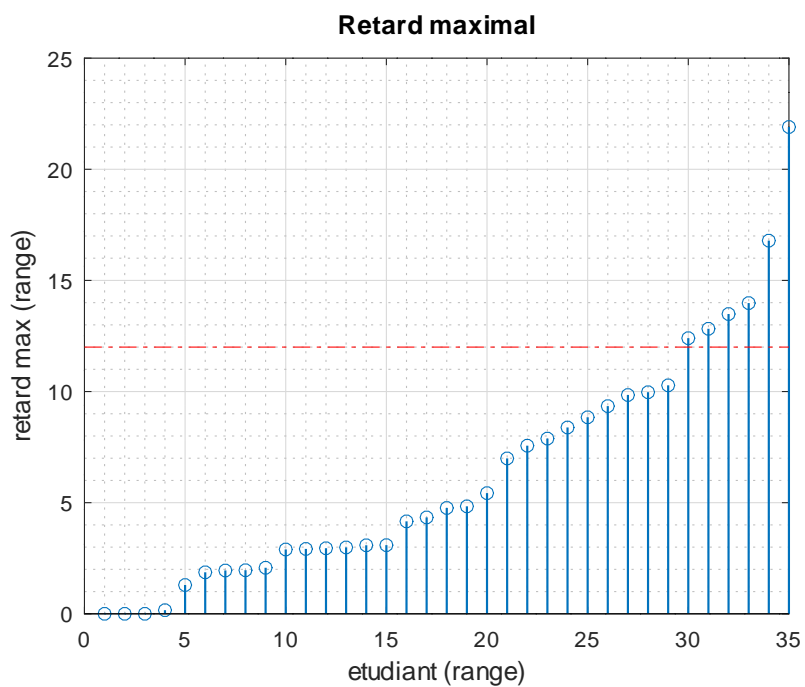


FIGURE 15 – CC2 : retard maximal rangé

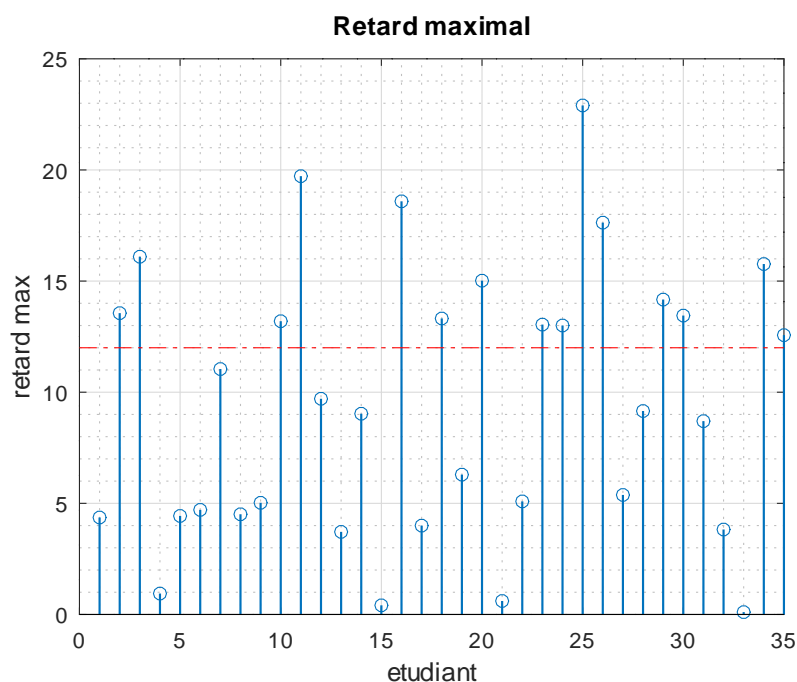


FIGURE 16 – CC3 : retard maximal

le CC2, certaines questions étant calculées, l'échange est plus compliqué).

Nous utilisons un second indicateur de comportement suspect, qui est basé sur les durées de réponse aux questions. Il s'agit de détecter les étudiants qui répondent très rapidement à des questions difficiles et/ou très lentement à des questions faciles.

La durée de réponse est calculé par différence d'horodatage. Lorsque la durée de réponse calculée est 0, on remplace cette valeur par $1/3$ (de minute). En effet, Moodle donne l'horodatage avec un pas d'une minute. En conséquence, lorsque le calcul par différence donne 0, cela signifie que t (horodatage de la question considérée) et t_p (horodatage de la question précédente) tombent dans le même intervalle de durée 1 minute. En supposant que t et t_p suivent des lois uniformes, un calcul élémentaire (voir annexe) montre que l'espérance mathématique de $t - t_p$ (sous la contrainte $t > t_p$) est $1/3$.

On range ensuite les durées dans l'ordre original des questions, puis on calcule la durée que les étudiants ont consacré, en moyenne, à chaque question. Pour ce calcul nous n'utilisons que les durées sûres (les durées calculées à partir d'horodatages estimés ne sont pas prises en compte). Le résultat obtenu pour le CC3 est indiqué sur la figure 18). Cela donne une bonne indication de la difficulté de chaque question : on voit par exemple que les étudiants ont répondu en une minute à la question 9 (question facile), mais on dû consacrer en moyenne 4 minutes à la question 2 (question difficile).

On peut constater que certains étudiants ont un profil de réponse qui semble totalement décalé de la difficulté des questions. Sur la figure 19 nous avons représenté en bleu la durée que l'étudiant 16 a consacré à chaque question du CC3 et en rouge la durée que les étudiants y ont consacré en moyenne.

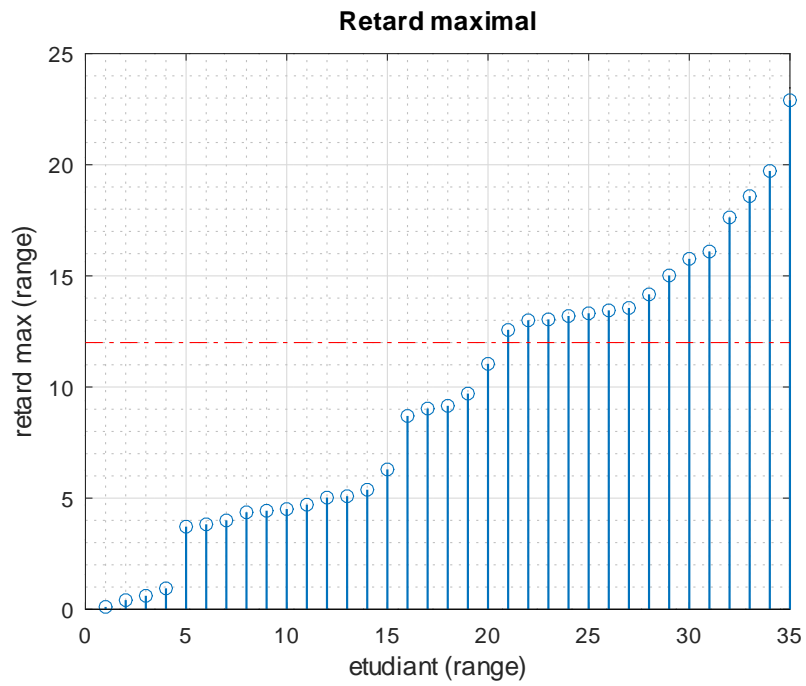


FIGURE 17 – CC3 : retard maximal rangé

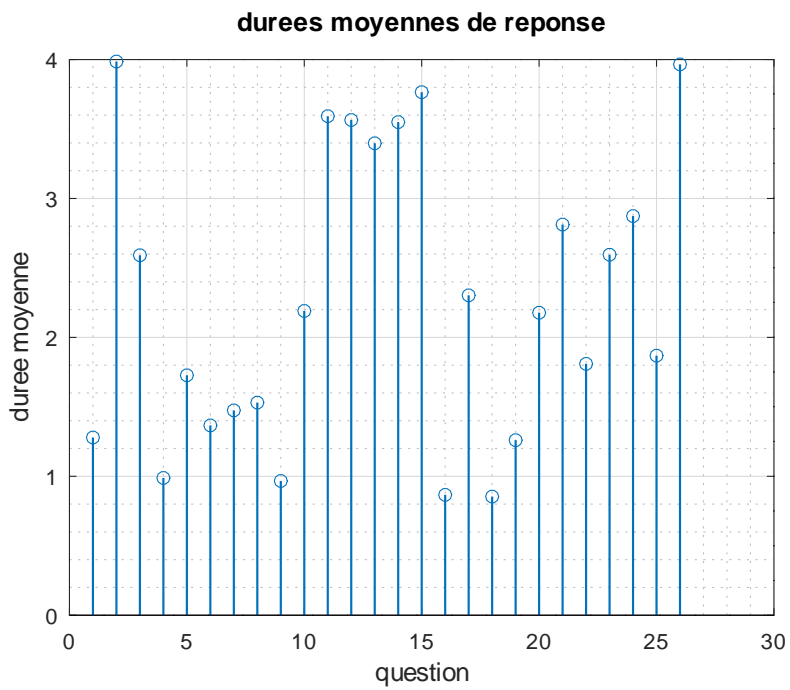


FIGURE 18 – CC3 : durées moyennes de réponse aux questions

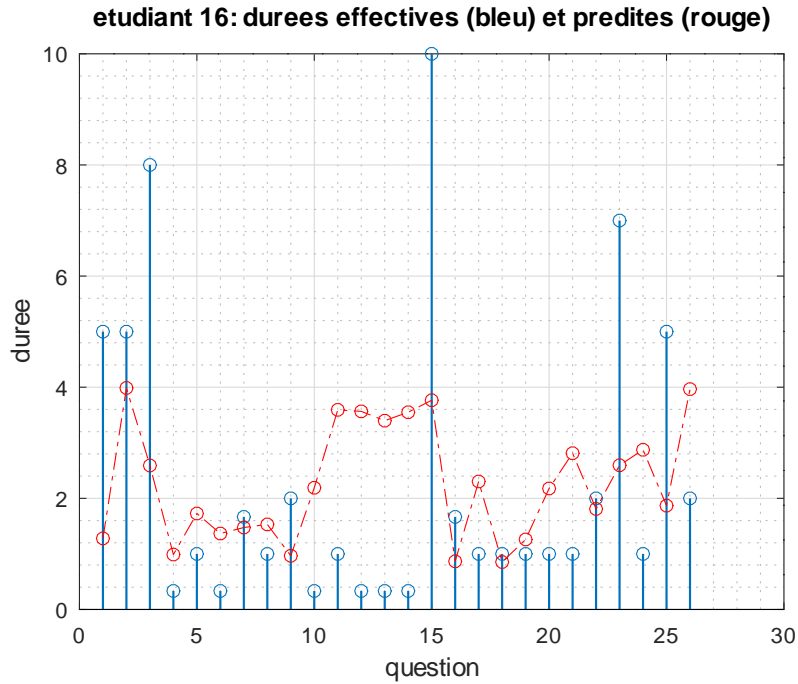


FIGURE 19 – CC3 : durées de réponse de l’étudiant 16

Le critère que nous utilisons pour détecter les étudiants au comportement suspect est basé sur l’écart logarithmique entre les durées de réponse d’un étudiant et les durées moyennes. Ainsi, si on note $d_{e,q}$ la durée de réponse de l’étudiant e à la question dont le numéro original est q , et \bar{d}_q la durée moyenne de réponse à cette question, on calcule :

$$L_e = \frac{1}{\#_eq} \sum_q \left| \ln \left(\frac{d_{e,q}}{\bar{d}_q} \right) \right|^2 \quad (6)$$

Le choix du carré a simplement pour but de donner plus de poids aux fortes déviations ce qui fait mieux ressortir les cas suspects. La somme n’est réalisée que sur les valeurs $d_{e,q}$ fiables et $\#_eq$ désigne le nombre de valeurs sommées. On peut remarquer que ce critère n’est rien d’autre que le moment d’ordre 2, c’est à dire la variance non centrée de l’écart logarithmique. Pour faire court, nous l’appelons donc “variance” dans les légendes des figures.

La figure 20 montre les résultats pour le contrôle CC2. Si l’on range les étudiants dans l’ordre des valeurs croissantes du critère (Fig. 21) on peut voir une rupture nette qui nous incite à mettre un seuil à 0.7 sur la figure 20 et conduit à considérer que les comportements des étudiants 3, 6, 16, 25, 31 sont potentiellement suspects.

La figure 22 montre les valeurs de L_e pour le CC3. On remarque que quelques étudiants (étudiants 25, 16, voire 35), se distinguent. Notons au passage que les étudiants qui étaient présents à ce contrôle sont les mêmes que ceux qui ont passé le contrôle CC2 : on retrouve ici les étudiants 16 et 25 que nous avons déjà repérés lors de l’analyse du CC2. La figure 23 qui contient les mêmes valeurs mais rangées, ne permet cependant pas de voir de rupture nette,

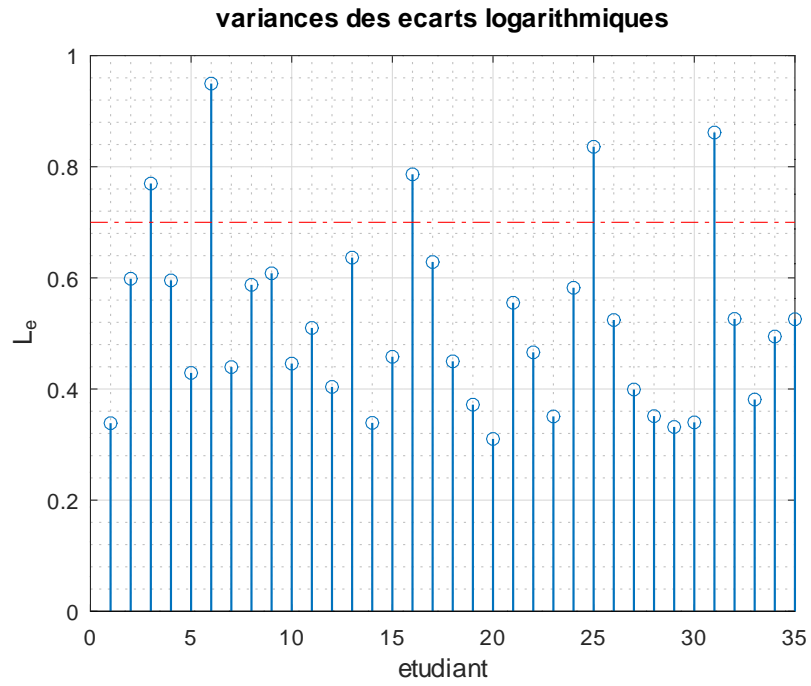


FIGURE 20 – CC2 : variances des écarts logarithmiques

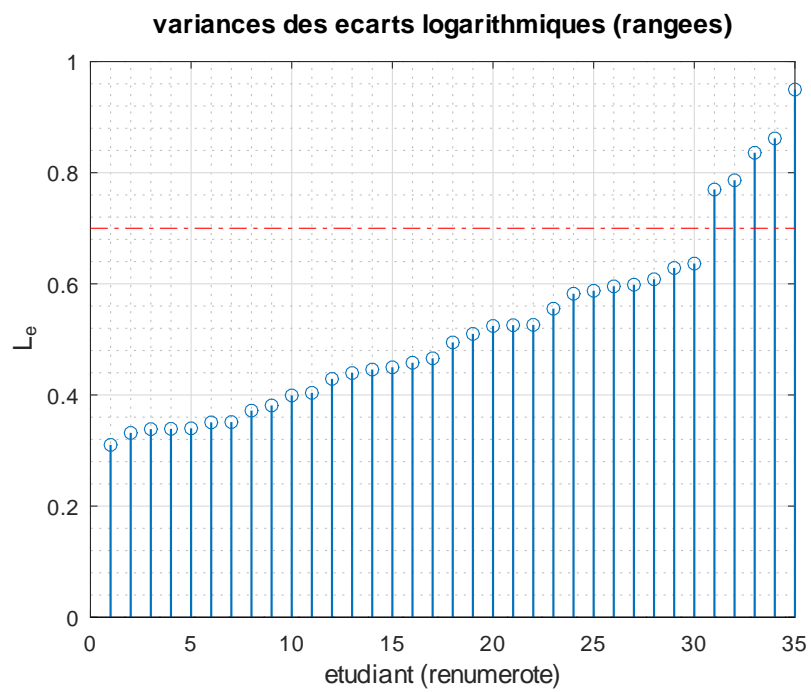


FIGURE 21 – CC2 : Variances rangees des écarts logarithmiques

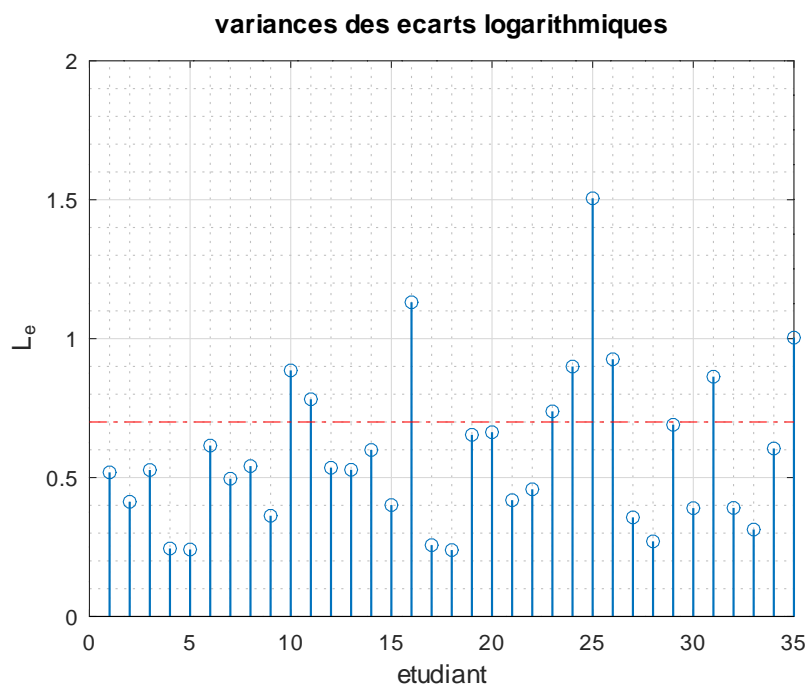


FIGURE 22 – CC3 : Variances des écarts logarithmiques

hormis pour un étudiant (étudiant 25 dans la numérotation originale). Le nombre d'étudiants qui dépassent le seuil est plus important que pour le CC2, probablement pour la raison qui a déjà été évoquée lorsque nous avons examiné les retards.

L'examen des histogrammes (Fig. 24 et 25) montre une dispersion légèrement plus importante pour le CC3, notamment du côté des valeurs négatives (étudiants répondant très rapidement au regard de la difficulté de la question) ce qui tend à confirmer cette hypothèse.

4.3 Analyse basée sur les réponses aux questions

4.3.1 Critère de similarité des réponses

Dans cette partie nous avons travaillé sur le tableau des réponses décrit dans la section 3. Ce tableau contient les réponses correctes et les réponses des étudiants pour chaque question. Nous avons également utilisé le tableau d'ordre (qui contient l'ordre dans lequel les questions ont été proposées à chaque étudiant).

Afin d'analyser les réponses, nous avons extrait les réponses des étudiants à chaque question et construit un nouveau tableau représenté comme suit (Fig. 26) :

- Le code « 0 » correspond aux non-réponses
- Le code « 1 » correspond aux bonnes réponses
- Les codes « n » ($n \geq 2$) correspondent aux différentes fausses réponses rencontrées

Au niveau de la figure 26, prenons l'exemple de l'étudiant 5 (cinquième ligne) : on voit qu'il a bien répondu à la première question, n'a pas répondu à la 2^e question, a mal répondu à la 3^e question, etc. Pour cette 3^e question (colonne 3), on voit que les étudiants 5 et 7 ont fait la

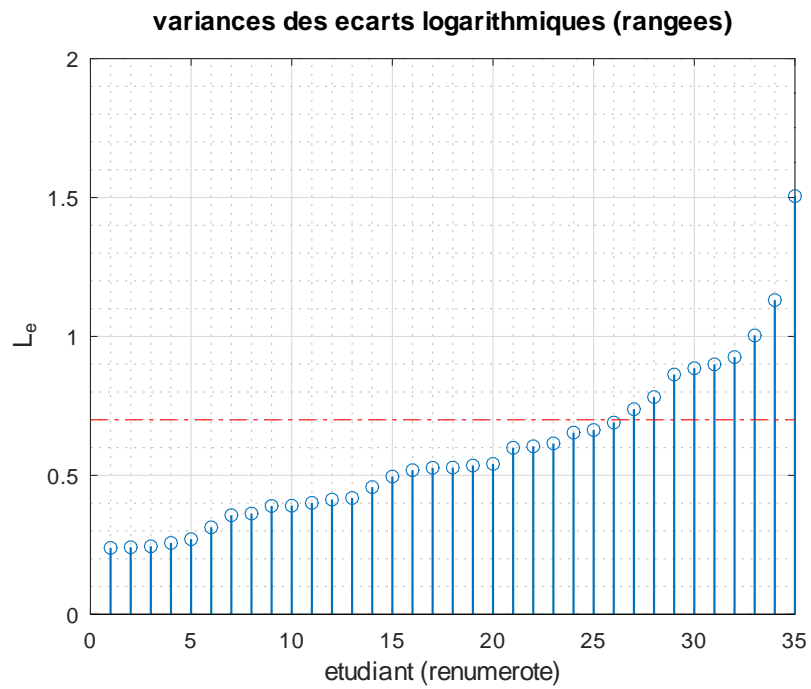


FIGURE 23 – CC3 : variances rangees des écarts logarithmiques

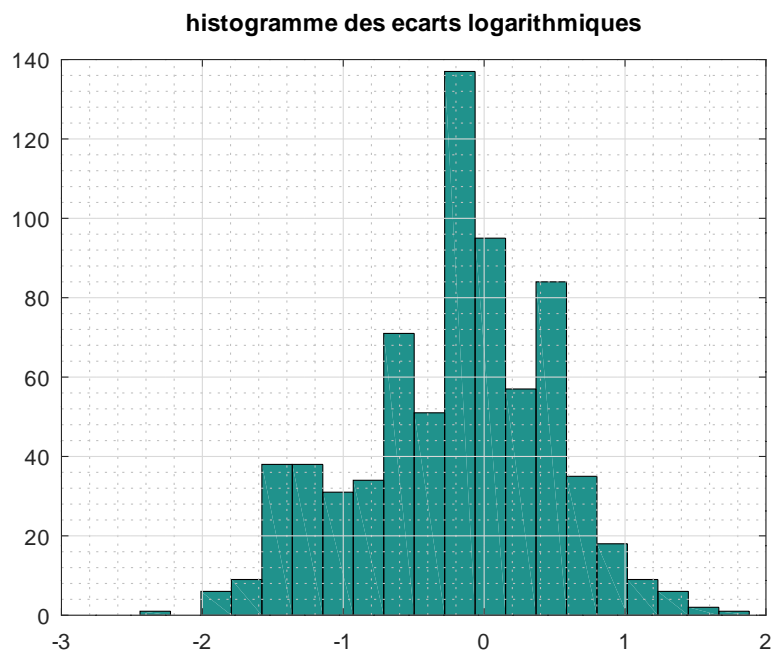


FIGURE 24 – CC2 : histogramme des écarts logarithmiques

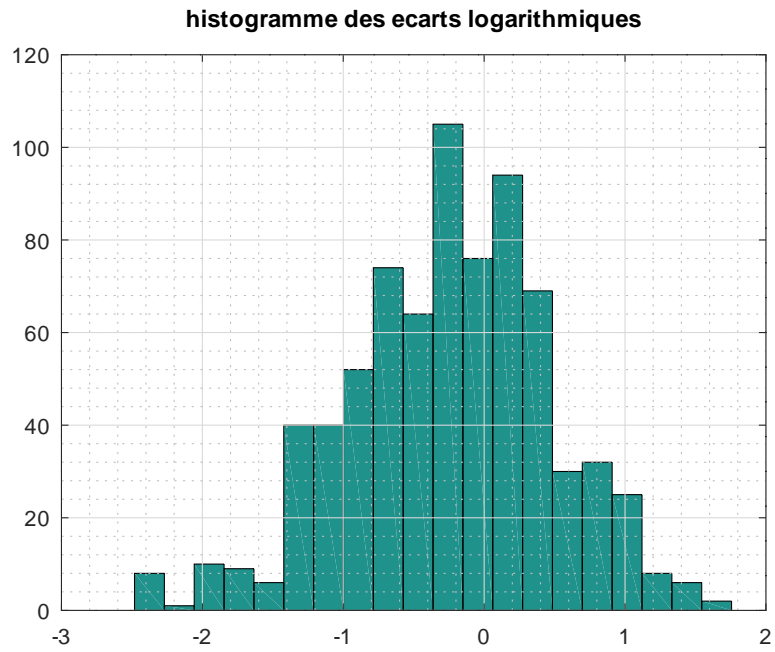


FIGURE 25 – CC3 : histogramme des écarts logarithmiques

	A	B	C	D	E	F	G
1	1	1	2	1	1	1	1
2	1	0	3	1	1	1	1
3	1	0	3	1	1	1	1
4	1	1	2	1	1	1	2
5	1	0	2	1	1	1	2
6	1	2	1	1	1	1	2
7	1	2	2	1	2	1	1
8	1	1	1	1	1	1	1
9	1	1	1	1	1	1	1
10	1	0	1	0	1	0	0

FIGURE 26 – CC3 : extrait de la distribution des réponses aux questions (1 ligne par étudiant)

même erreur (code 2) alors que l'étudiant 3 a également une erreur, mais différente (code 3). Au sein des erreurs, lorsque l'on compare les réponses de 2 étudiants, ce codage nous permet donc de distinguer les erreurs identiques et les erreurs différentes.

La similarité peut être détectée visuellement quand il s'agit d'un très petit groupe (étudiants, questions), mais quand on a 35 étudiants et 26 questions la détection visuelle est impossible. C'est la raison pour laquelle nous avons calculé un critère de similarité, que nous représentons dans une matrice de similarité. Dans une première version, cette dernière estimait les similarités entre les étudiants deux à deux en comptant les non-réponses et les fausses réponses identiques. Nous avons ensuite amélioré ce critère en pondérant les similarités prises en compte. Pour chaque question q , notons $z(q, r)$ la proportion d'étudiants ayant apporté la réponse de code r à la question q . On compte tous les codes, y compris les non-réponses (code 0) et les bonnes réponses (code 1). Notons également $\theta_{i,j}$ l'ensemble des questions pour lesquelles les étudiants i et j ont le même code de réponse et $r_i(q)$ le code de la réponse de l'étudiant i à la question q . Le critère calculé est :

$$S_{i,j} = -\frac{1}{Q} \sum_{q \in \theta_{i,j}} \ln \{z(q, r_i(q))\} \quad (7)$$

Ceci permet de pondérer les similarités. En effet, si pour une question donnée une forte proportion d'étudiant a le code r , le fait que deux d'entre eux aient ce code n'est pas une marque très probante de collusion et il est logique de lui donner une faible pondération (ce qui est bien le cas car alors $-\ln \{z(q, r)\}$ est faible).

La figure 27 montre la matrice de similarité obtenue pour le CC1 (la diagonale a été forcée à 0) et la figure 28 l'histogramme des similarités. L'allure de cet histogramme nous a incités à fixer un seuil s à 0.38 pour détecter les similarités les plus significatives. Nous avons vérifié que ce seuil reste pertinent pour les autres contrôles.

4.3.2 Résultats obtenus sur les trois contrôles

Dans le cas d'un contrôle à navigation libre, la distance entre horodatages nous permettait de repérer des groupes de tricheurs potentiels. Pour le CC1, nous utilisons le critère de similarité des réponses pour valider ou invalider les résultats obtenus par analyse de l'horodatage. Si l'on représente la matrice de similarité en la restreignant aux étudiants que l'horodatage avait considérés suspects, et en les ordonnant par groupe, on voit apparaître des sous matrices (Fig 29 et 30) qui, dans notre cas, valident les 3 groupes qui avaient été détectés. On présente donc à l'enseignant la figure 12 sans modification.

Dans le cas d'un contrôle séquentiel (CC2 et CC3), la situation est différente, puisque l'horodatage nous permet seulement de repérer des étudiants au comportement suspect, mais pas de les regrouper. De plus, lorsqu'au sein d'un groupe un ou plusieurs étudiants sont source d'information (les autres étant les copieurs), ces étudiants source ne sont pas repérés par l'analyse de l'horodatage. Dans ce cas, nous procédons différemment. D'abord, sur la base de la matrice des similarités entre notes, nous repérons des groupes (un étudiant a appartient à un groupe s'il existe au moins un autre étudiant b du groupe tel que $S_{ab} > s$ (où $s = 0.38$ est le seuil défini précédemment)). Ensuite, nous validons chaque groupe uniquement si en son sein l'un au moins

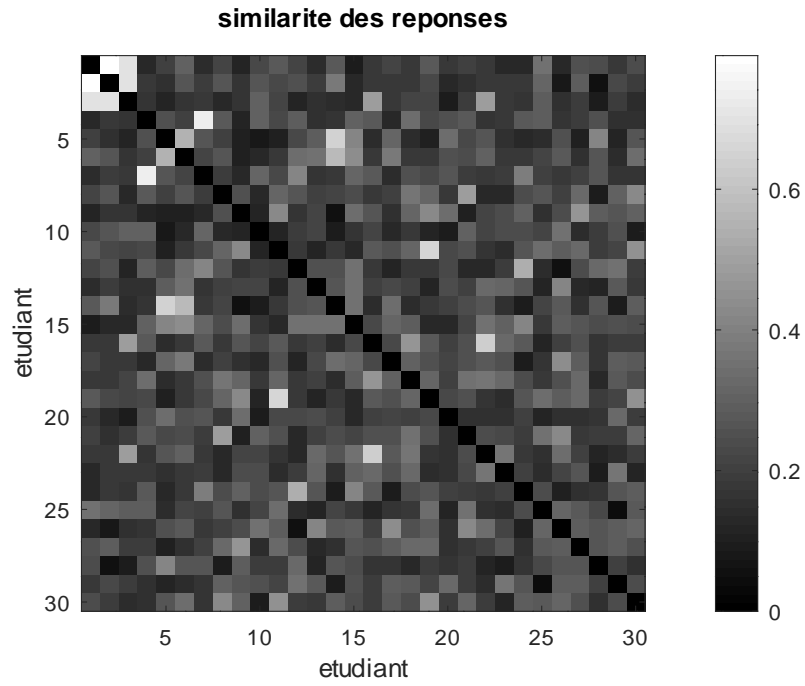


FIGURE 27 – CC1 : matrice des similarités entre réponses

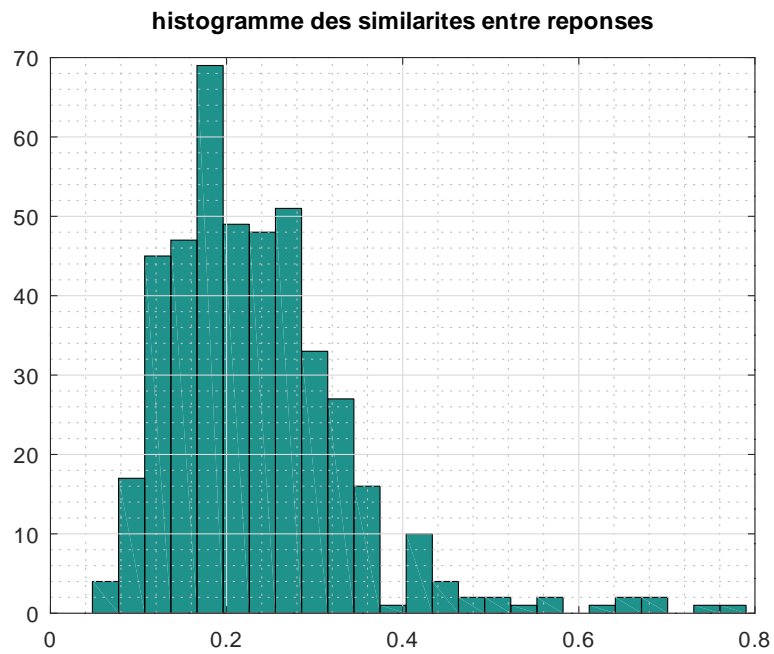


FIGURE 28 – CC1 : histogramme des similarités entre réponses

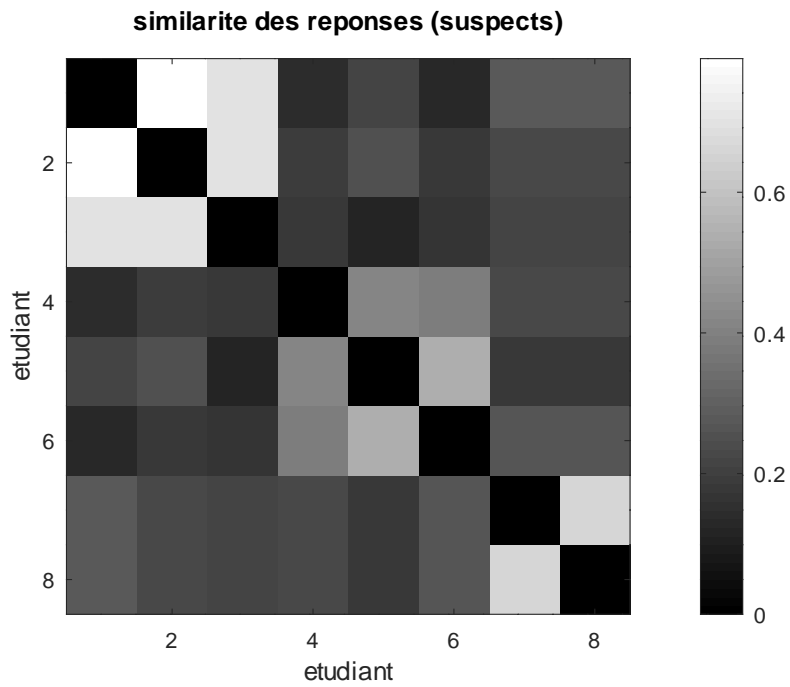


FIGURE 29 – CC1 : matrice des similarités entre réponses, limitée aux cas suspects

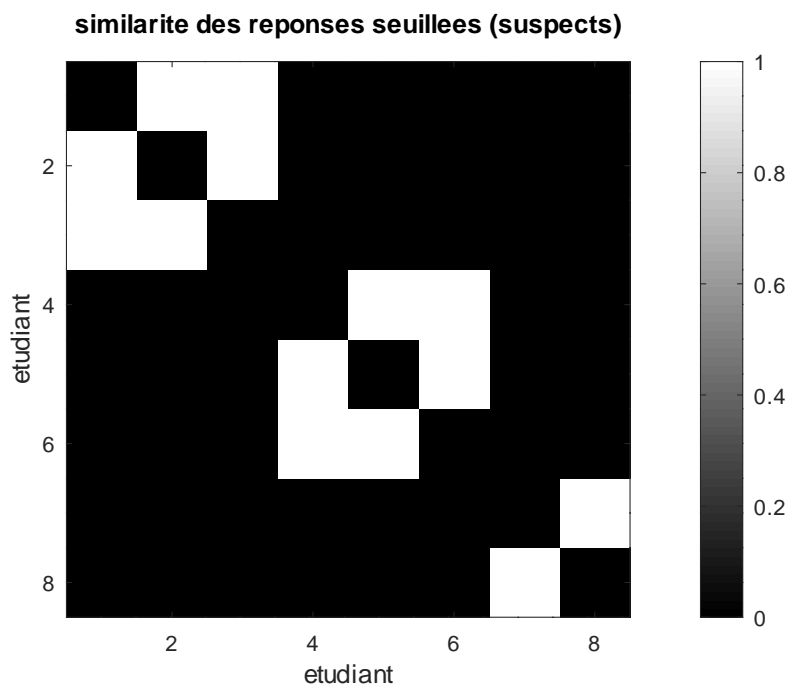


FIGURE 30 – CC1 : matrice des similarités entre réponses, limitée aux cas suspects, seuillée

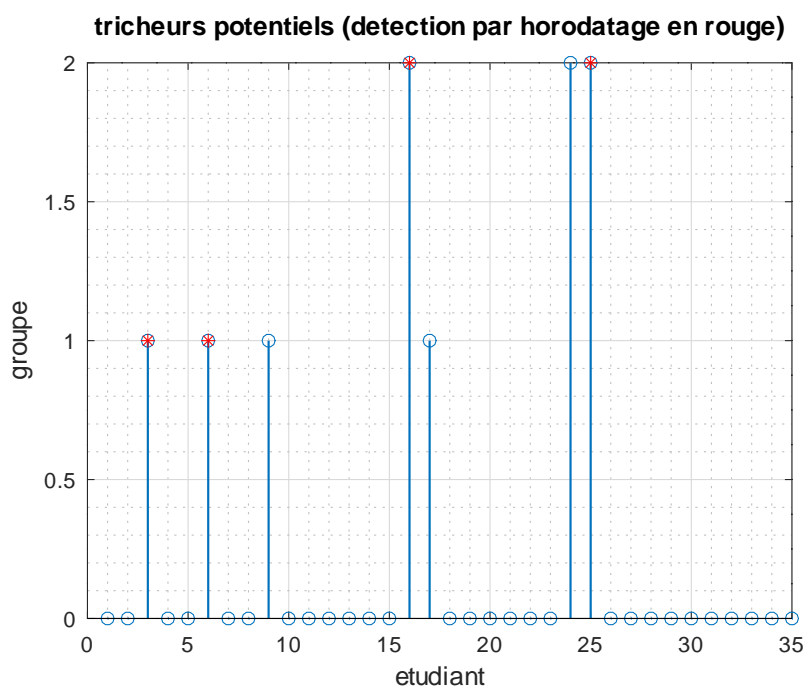


FIGURE 31 – CC2 : groupes de tricheurs potentiels

des étudiants a été détecté par l’horodatage. Les résultats présentés à l’enseignant sont indiqués sur les figures 31 et 32. Il est intéressant de noter que le trinôme (16,24,25) a été détecté sur les deux contrôles, ce qui renforce la conviction qu’il s’agit bien d’un groupe au sein duquel l’information s’échange abondamment. On pourra noter une petite différence cependant : 24 avait été détecté comme source principale d’information dans le CC2, mais ce n’est plus de cas dans le CC3, ce qui peut laisser penser que l’information a circulé de manière moins unidirectionnelle dans ce dernier cas.

5 Conclusion

Nous avons proposé différents critères simples permettant de détecter l’échange d’information entre étudiants lors des contrôles à distance et nous les avons illustrés sur des données issues de contrôles ayant réellement eu lieu. Lorsque les mêmes étudiants passent plusieurs contrôles, le fait que certains groupes soient détectés à chaque fois est de nature à renforcer la conviction d’un échange effectif d’information. Les résultats sont présentés graphiquement à l’enseignant et il peut, s’il le souhaite, visualiser le comportement particulier des certains étudiants (ordre des réponses aux questions, comparaison de l’horodatage effectif et de l’horodatage prédit, etc.) afin de renforcer sa conviction.

Nous avons vu que, si sur certains critères le seuil de détection choisi correspond à une rupture assez nette dans les comportements des étudiants, ce n’est pas toujours le cas. Nous ne disposons pas de données en nombre suffisant pour permettre de réellement valider les seuils choisis, même si nous avons pu constater qu’ils donnent de bons résultats sur les trois contrôles

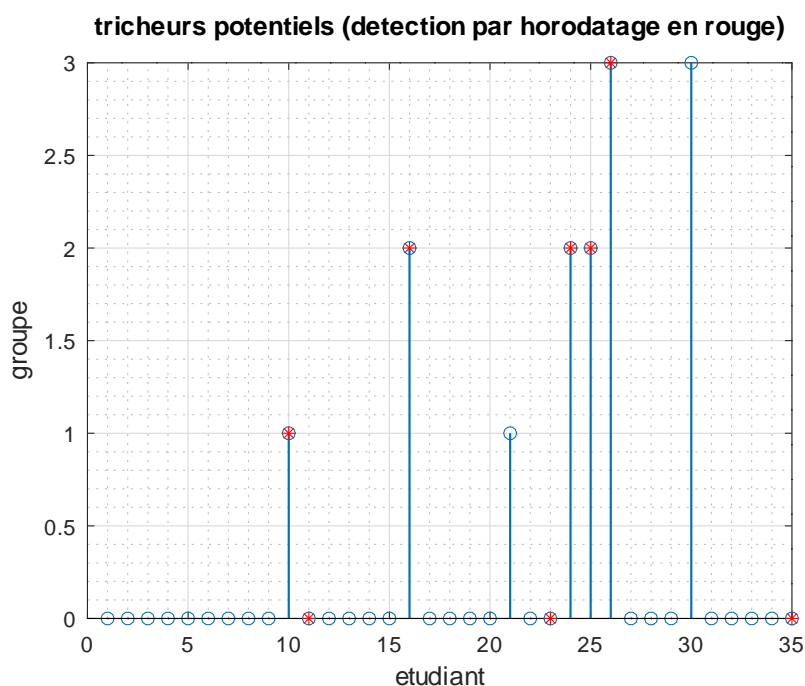


FIGURE 32 – CC3 : groupes de tricheurs potentiels

traités. Une approche alternative pourrait être d’éviter les seuils en demandant à l’enseignant combien de cas suspects il souhaite voir (N) et en lui présentant les N cas les plus suspects, avec différents graphiques lui permettant de se faire son opinion.

Remerciements

Ce travail a été réalisé dans le cadre des actions de formation par la recherche de la chaire “CyberIoT” portée par Roland Gautier.

Les auteurs remercient Philippe Rostaing et Koffi Yao, UBO, pour l’accès à des résultats de contrôles à distance, et Florence Bruneau, UBO, pour ses conseils relatifs à la plateforme Moodle.

Biographie des auteurs

Gilles Burel a obtenu le diplôme d’ingénieur de Supélec en 1988, le doctorat de l’Université de Brest en 1991 et l’habilitation à diriger des recherches en 1996. De 1988 à 1997 il a travaillé à Thomson CSF, puis à Thomson Multimedia, dans le domaine du traitement d’images et des communications numériques. Depuis 1997, il est Professeur à l’Université de Bretagne Occidentale (UBO). De 2008 à 2016, il a été directeur adjoint du Lab-STICC (UMR CNRS 6285). Ses principaux domaines d’intérêt actuels sont les communications quantiques, l’acquisition compressée et la surveillance du spectre.

Khawla Ben Gaied & Ramla Ben Abdelkader ont obtenu le diplôme de M1 de l'Institut Supérieur des Technologies Médicales de Tunis en 2020 et le M2 Signal et Télécommunications de l'Université de Bretagne Occidentale en co-accréditation avec l'ENIB en 2021. Dans le cadre de leur projet de M2, elles ont contribué au travail relaté dans l'article.

Roland Gautier a obtenu un doctorat de l'Université de Nice-Sophia Antipolis en 2000 et l'habilitation à diriger des recherches en 2013. Depuis 2001, il est Maître de Conférences à l'Université de Brest. Il a exercé plusieurs responsabilités d'équipes au sein du Lab-STICC et depuis 2018 il est titulaire de la Chaire d'Excellence «CyberIoT». Ses principaux domaines d'intérêt sont le traitement du signal, les communications numériques et la sécurité de la couche physique des transmissions.

Annexe

Nous donnons ici quelques résultats mathématiques de base utilisés dans l'article.

Considérons une variable x qui suit une loi uniforme dans l'intervalle $[0, 1]$ et notons y la valeur absolue de la différence entre deux réalisations de x .

La moyenne de y est alors :

$$\bar{y} = \int_0^1 \int_0^1 |x_2 - x_1| dx_1 dx_2 \quad (8)$$

$$= 2 \int_0^1 dx_1 \int_{x_1}^1 (x_2 - x_1) dx_2 \quad (9)$$

$$= \int_0^1 dx_1 \left[(x_2)^2 - 2x_1 x_2 \right]_{x_1}^1 \quad (10)$$

$$= \int_0^1 \left(1 - 2x_1 + (x_1)^2 \right) dx_1 \quad (11)$$

$$= \left[x_1 - (x_1)^2 + \frac{(x_1)^3}{3} \right]_0^1 \quad (12)$$

$$= \frac{1}{3} \quad (13)$$

La loi de probabilité de y est

$$p(y) = 2(1 - y) \quad (14)$$

pour $y \in [0, 1]$ et est nulle en dehors de cet intervalle.

La médiane de y , que l'on notera \check{y} est définie par :

$$\int_0^{\check{y}} p(y) dy = \frac{1}{2} \quad (15)$$

ce qui nous donne :

$$\left[y - \frac{y^2}{2} \right]_0^{\check{y}} = \frac{1}{4} \quad (16)$$

On doit donc résoudre

$$2\check{y}^2 - 4\check{y} + 1 = 0 \quad (17)$$

La solution devant être dans l'intervalle $[0, 1]$ on obtient :

$$\check{y} = 1 - \frac{1}{\sqrt{2}} \quad (18)$$

Si la variable x est uniformément répartie dans $[a, b]$ il suffit d'ajuster les formules précédentes :

$$\bar{y} = a + \frac{b - a}{3} \quad (19)$$

et

$$\check{y} = a + \left(1 - \frac{1}{\sqrt{2}}\right)(b - a) \quad (20)$$

Dans le cas particulier où x est un entier appartenant à $[1, Q]$, et où $x_1 \neq x_2$ alors $y \in [1, Q - 1]$ et la probabilité de y est :

$$P(y) = \frac{Q - y}{(Q - 1)Q/2} \quad (21)$$

Le dénominateur étant déterminé par $\sum_{y=1}^Q P(y) = 1$

La moyenne est

$$\bar{y} = \sum_{y=1}^{Q-1} yP(y) \quad (22)$$

$$= \frac{2}{(Q - 1)Q} \left(Q \frac{Q(Q - 1)}{2} - \frac{(Q - 1)Q(2Q - 1)}{6} \right) \quad (23)$$

$$= Q - \frac{2Q - 1}{3} \quad (24)$$

$$= \frac{Q + 1}{3} \quad (25)$$

Références

- [1] Alice Pairo-Vasseur, « Fraude aux examens en ligne : Tout le monde le faisait! », Le Point, 2 juillet 2020 (www.lepoint.fr/education/fraude-aux-examens-en-ligne-tout-le-monde-le-faisait-02-07-2020-2382811_3584.php)
- [2] Qingping He, Michelle Meadows and Beth Black, « Statistical techniques for studying anomaly in test results : a review of literature », Office of Qualifications and Examinations Regulation, United Kingdom, March 2018, Ofqual/18/6355/5 (pdf disponible sur assets.publishing.service.gov.uk)
- [3] Kaiwen Man, Jeffrey R. Harring, Sandip Sinharay, « Use of Data Mining Methods to Detect Test Fraud », JEM : Journal of Educational Measurements, Volume 56, Issue 2, Summer 2019, Pages 251-279

- [4] « Moodle - L'environnement d'apprentissage le plus répandu au monde » (moodle.org)
- [5] Valentin Hamon-Beugin, « Des logiciels de télésurveillance pour les examens inquiètent les étudiants », *Le Figaro*, 10 avril 2020 (www.lefigaro.fr/secteur/high-tech/des-logiciels-de-telesurveillance-pour-les-examens-inquietent-les-etudiants-20200410)
- [6] Romain Ledroit, « Tous les examens seront-ils bientôt surveillés à distance ? », *Usbek&Rica*, 5 juillet 2018 (usbeketrica.com/fr/article/tous-les-examens-seront-ils-bientot-surveilles-a-distance)
- [7] « TestWe - Solution tout en un pour les examens en ligne » (testwe.eu)
- [8] « ProctorExam - Online proctoring services » (www.proctorexam.com)
- [9] « Managexam - Solution d'examens à distance » (managexam.com)
- [10] « Smowl - Online proctoring for online assessments » (smowl.net)
- [11] « Crise sanitaire : la technopolice envahit l'université », *La quadrature du net*, 30 avril 2020 (www.laquadrature.net/2020/04/30/crise-sanitaire-la-technopolice-envahit-luniversite)
- [12] Patrick Finot, « Evalbox : une solution professionnelle pour gérer vos examens en ligne ou sur papier », *Informatique-Enseignant*, 21 octobre 2015 (www.informatique-enseignant.com/evalbox-gerer-examens-en-ligne-ou-sur-papier)
- [13] G. Rasch, « Probabilistic models for some intelligence and attainment tests. » (Copenhagen, Danish Institute for Educational Research), expanded edition (1980) with foreword and afterword by B.D. Wright. Chicago : The University of Chicago Press.
- [14] « TEF : comment détecter les cas de fraudes dans les questionnaires à choix multiple ? », *Le Français des Affaires*, 25 septembre 2018 (www.lefrancaisdesaffaires.fr/tef-comment-detecter-les-cas-de-fraudes-dans-les-questionnaires-a-choix-multiple/)
- [15] Yunxiao Chen, Yan Lu, and Irimi Moustaki, « Detection of Two-Way Outliers in Multivariate Data and Application to Cheating Detection in Educational Tests » arXiv :1911.09408v2, 21 Dec 2020
- [16] Wim J. van der Linden, « A Bivariate Lognormal Response-Time Model for the Detection of Collusion Between Test Takers », *Journal of Educational and Behavioral Statistics*, 2009
- [17] https://github.com/gilles-burel/detection_of_test_cheating