Analyse & Classification sur le dataset ‘‘Student Performance’’

Trang Nghiem

I. Description de dataset

Dans ce projet, je vais analyser les les données de performances des étudiants. Ce dataset s’appelle *Kalboard 360*, collecté en 2016 par le système de management d’étude.

Ce dataset comprend 480 lignes et 16 caractères (features). Les features sont classé en trois grandes catégories :

(1) Caractères démographiques telles que le sexe et la nationalité.

(2) Caractères de formation académique telles que le niveau éducatif, le niveau scolaire et la section

(3) Caractères comportementales, telles que nombre de fois de main levée, visite & discussion des élèves, satisfactions de l’école.

Le dataset comprend 305 hommes et 175 femmes. Ces élèves viennent de différente origines, comme 179 de Kuwait, 172 de Jordan, 28 de Palestine.

Les données sont collectées au cours de 2 semestres : 245 dossiers pendant le premier semestre, 235 dossiers pendant le deuxième semestre.

Les jours d’absence sont également comptés et divisés en deux classes : 191 élèves dépassent 7 jours d’absence et 289 élèves avec moins de 7 jours.

Les tendances de relation avec les parents, et les satisfactions des parents sont également incluses dans ce dataset.

Les étudients sont classifié en 3 classes, selon leurs notes :

+ L (Low-level) (Niveau bas) : notes entre 0-69

+ M (Middle-Leval) (Niveau moyen) : note entre 70-89

+ H (High-Level) (Niveau élevé) : note entre 90-100

II. Analyse et classification

Ces données seront étudiées en 2 étapes : visualisation et model de classification.

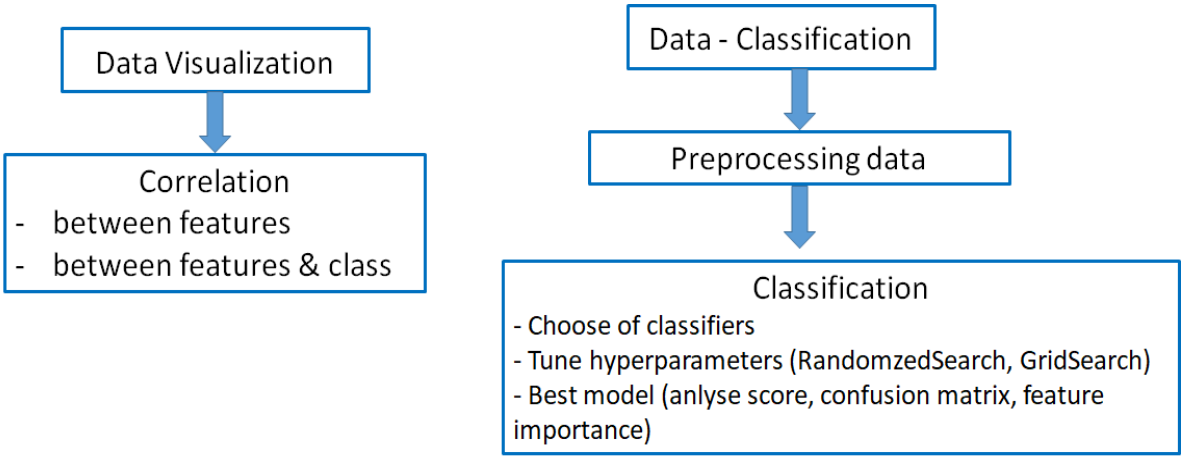


Figure 1 : Schéma des étapes de visualisation et classification.

La partie visualisation nous permet d'avoir une idée de corrélation entre les caractères (features), entre les caractères et les résultats des élève (ici, la ‘‘class’’ des élèves).

En deuxième partie, nous allons traiter ces données, c-a-d ‘‘preprocessing data’’, afin d'avoir une bonne entrée pour les modèles de machine learning. Les modèles choisis sont évidemment des ‘‘Classifier’’, comme *DecisionTreeClassifier*, *RandomForestClassifier*, *LogisticRegression* et *GradientBoostingClassifier*.

II.1. Visualisation des données

Quelques corrélations visuelles entre les caractères et la classe (résultat) finale des étudiants sont représentés dans cette partie.

D'abord, j'analyse les résultats selon le sexe : homme (M-Male) et femme (F-Female). La plupart des femmes ont de bons résultats, classés en M (Middle-level) et H (High-level). Au contraire, la majorité des hommes sont en classes M et L (Low-level). En corrélation visuelle, on peut dire de : le caractère d'être une femme a un effet positif sur la classe finale, et d'être un homme – effet négatif.

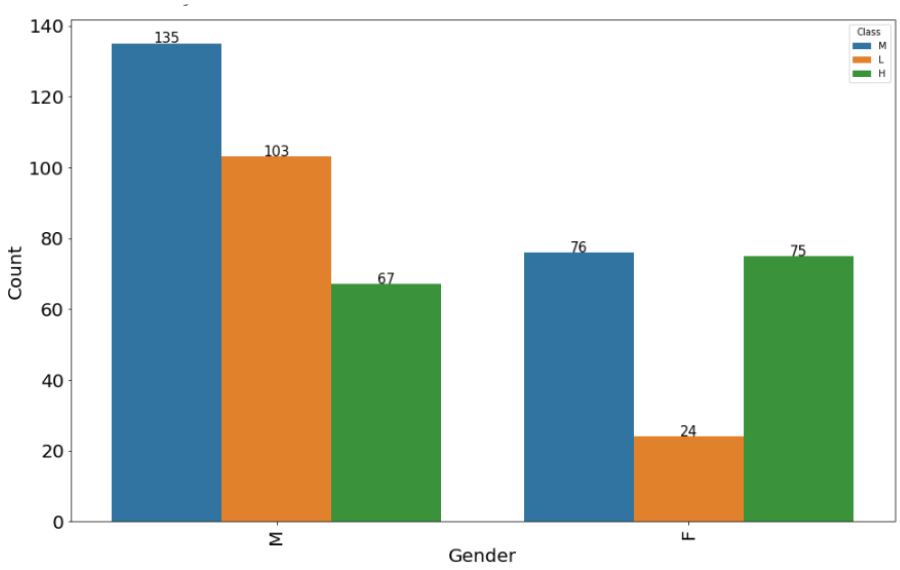


Figure 2 : Les classes de résultats (M, L, H) en fonction du sexe. M : bleu, L : jaune, H : vert

Ensuite, la tendance de relation avec les parents est illustrée dans la figure 3. La plupart des élèves, qui ont de bonne relation avec leur mère, obtiennent des note H e M. Ce caractère a un impact positif sur le résultat d'étude.

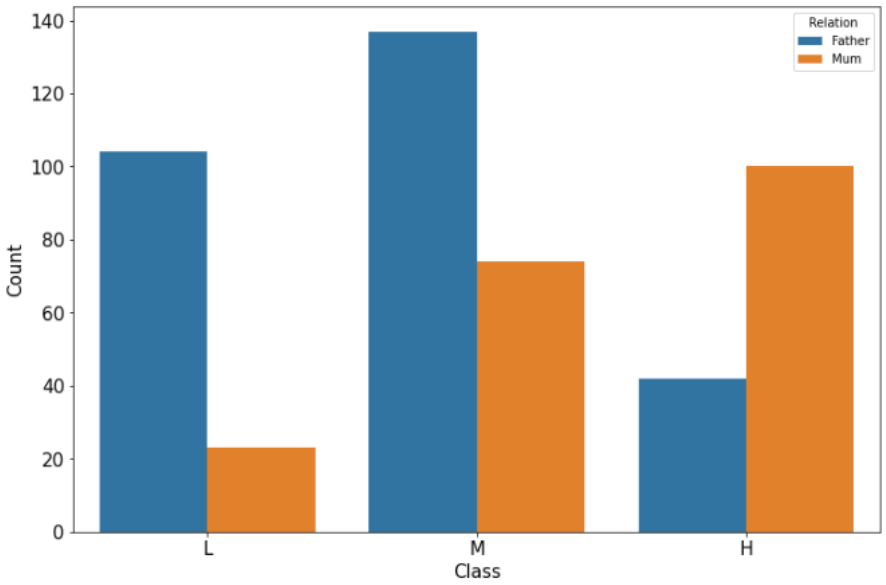


Figure 3 : Impact de relation entre les élèves et leurs parents sur le résultat d'étude

La plupart des étudiants de classes M et H ont pris moins de 7 jours d'absence. Contrairement, les étudiants de classe L plus de 7 jours. Donc, ce caractère (moins de 7 jours d'absence) a potentiellement un effet positif sur les classes finales.

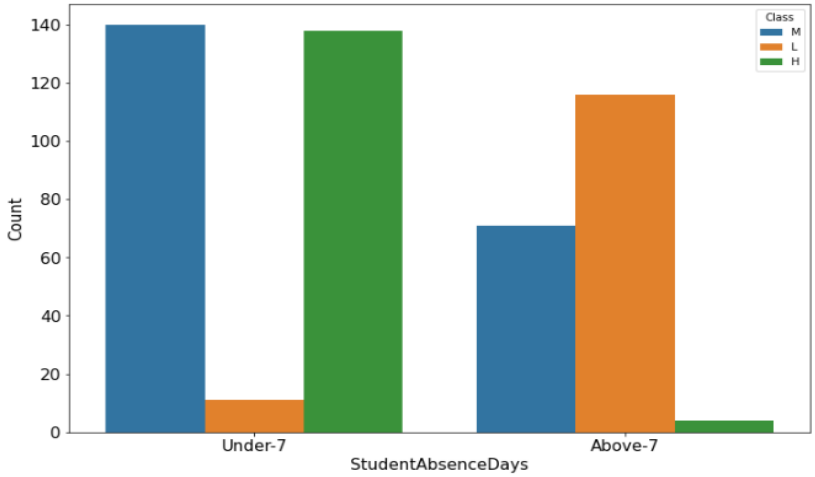


Figure 4 : Impact de nombre de jours d'absence sur le résultat d'étude

Les caractères peuvent potentiellement impacter sur les résultats d'études sont : les nombre de discussion, main levée, visite et vue d'annonce. Il est évident que les bons élèves participent activement pendant les cours et les activités de l'écoles ; et les moins bons sont moins manifestent. Par conséquence, ces caractères sont potentiellement des critères, pour classer les classes de résultats.

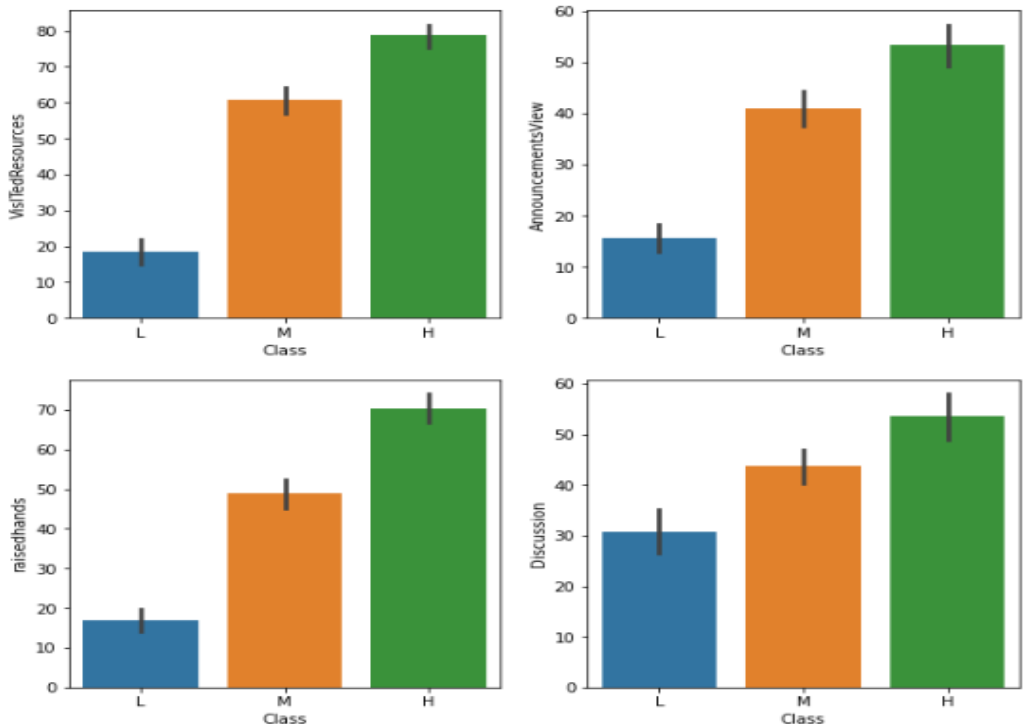


Figure 5 : Impact des caractère sur le résultat d'étude : visite (en haut à gauche), Vue d'annonce (haut-droite), main levée (bas-gauche), discussion (bas-droite)

Le tableau 1 résume les corrélations ci-dessus.

	Positive	Negative
Being male		–
Being female	+	
Relation with mum	+	
Absence Days (above 7)		–
Disussion, RaisedHand, Visitesd Ressource, Announcement	+	

Tableau 1 : Analyse de corrélation entre les caractères et les résultats des étudiants.

II.2. Modèle de classification

Les étapes de traitement de données (data preprocessing), choix des modèles de classification et raffinement des hyperparamètres, discusion des résultats obtenus, sont illustrées dans le schéma de figure 6.

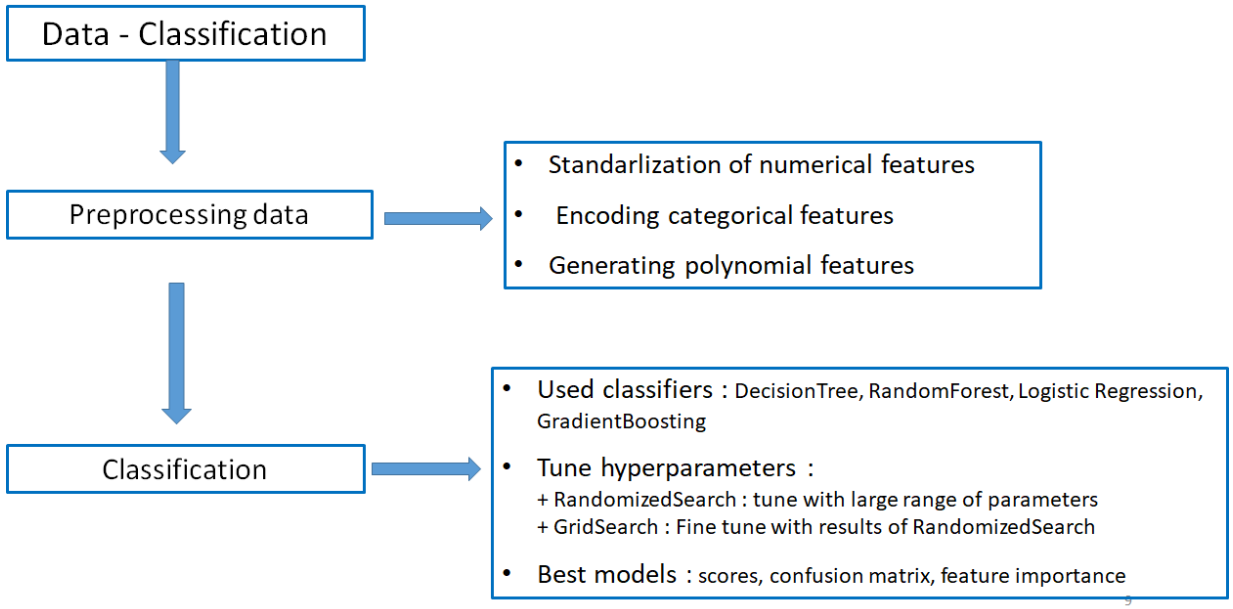


Figure 6 : Schéma des étapes de classification

A. Traitement des données (Data preprocessing) :

- Les caractères numériques sont standardisés, afin limiter les larges écarts entre eux.
- Les caractères catégoriels sont encodés, en utilisant OneHotEncoder
- Test s'il existe des relations entre les caractères, ou l'ordre des caractères en utilisant PolynomialFeatures.

  Dans ce dataset, il n'en a pas, donc la degrée des caractères reste à 1.

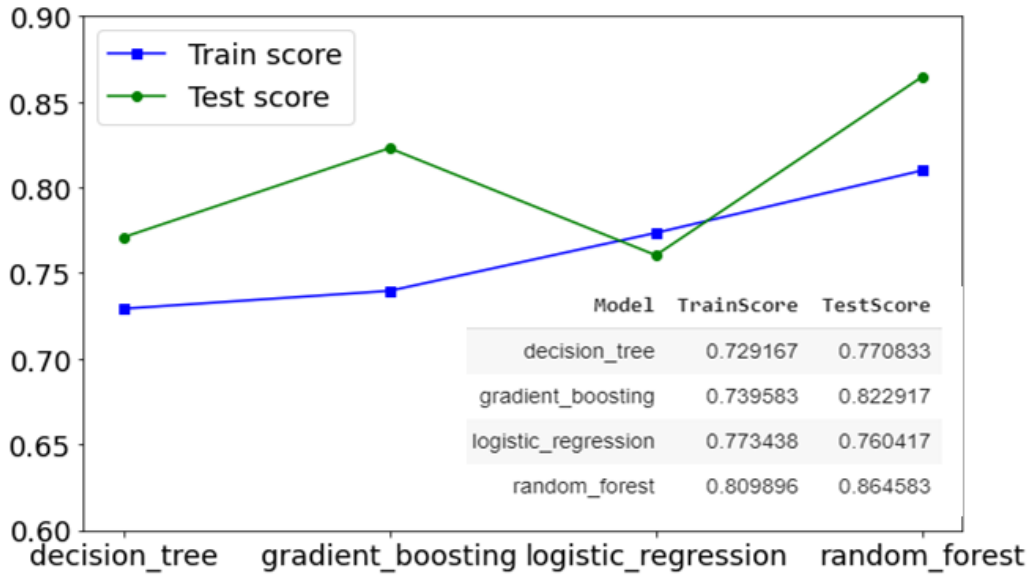
B. Modèle de classification :

Les résultats des étudiants sont divisé en 3 classes, donc, les classifiers de sklearn (Python) sont à choisir.

- Choix de modèles pour classifier : DecisionTreeClassifier, RandomForestClassifier, LogisticRegression, GradientBoostingClassifier.
- Rafinement des hyperparametres sont effectués en 2 étapes :

  + RandomizedSearch : scanner avec larges ranges afin trouver rapidement les bons ensembles de paramètres

  + GridSearch : grace aux paramètres trouvés par RandomizedSearch, les paramères sont encore plus raffinés, afin de trouver les finaux paramètres.
- Analyse des résultats obtenus : scores, matrice de confusion, importances des caractères (feature importances).



Model	TrainScore	TestScore
decision_tree	0.729167	0.770833
gradient_boosting	0.739583	0.822917
logistic_regression	0.773438	0.760417
random_forest	0.809896	0.864583

Figure 7 : Scores obtenus par les 4 modèles choisis. Train score est en courbe bleu, test score est en vert

Les train scores et test scores sont tracés en fonction des modèles de classification dans la figure 7. Les score sont assez élevés avec RandomeForestClassifier : train score de 81%, test score de 86.5%. Par contre, le test score est légèrement plus grand que le train score, donc ce modèle est un peu underfitting. Cet effet est également présent dans le cas de DecisionTree et GradientBoosting. Pour le cas de LogisticRegression, les score sont assez proches (train score 77.3%, test score 76% - pas d'effet underfitting). En machine learning, les datase contient normalement de milliers à plusieurs millions donnés. Cela permet les modèles de travailler sur les features et de donner de bon résultats. Mais le dataset Student Performance contient seulement 480 lignes, qui sont très peu pour les modèles de Machine Learning. Une augmentation de données pourra fortement accroître la précision des modèles.

Dans la figure 8, la matrice de confusion obtenue par RandomForestClassifier est présentée. La plupart des classes sont bien classé.

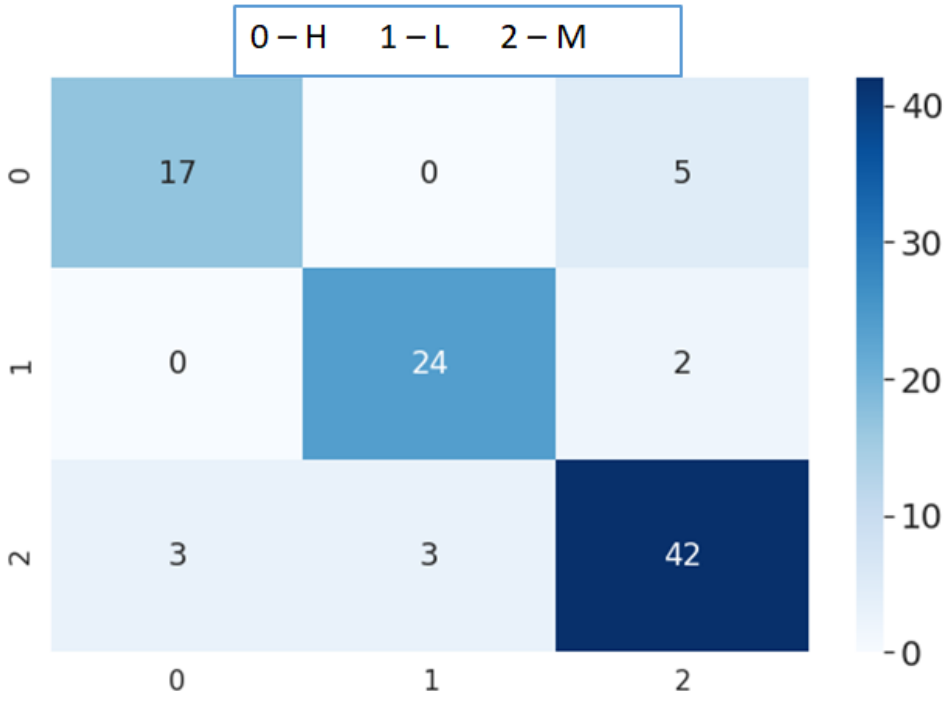


Figure 8 : Matrice de confusion obtenue par RandomForestClassifier sur le test set. Les classes sont encodées : 0 représente classe H, 1 pour L, 2 pour M

C. Feature Imporance

Dans cette partie, nous allons comparer les 'feature importance' des modèles, en particulier RandomForestClassifier et GradientBoostingClassifier. Les coefficients d'importance des 15 features, qui sont les plus imporants dans ces deux modèles, sont reportés dans la Figure 9.

Il est très intéressant à noter que ces features sont mêmes et les mesures d'importance sont très proches entre ces 2 modèles. Ils sont divisés en 3 classes : les 6 premiers (dans cage rouge), suivi par deux groupes de 4 (cage jaune) et 5 (cage vert) features. Les features correspondant dans ces trois groupes sont mêmes. Il y a une légère différence sur les coefficients d'importance, mais ils sont à même ordre grandeur. Cela me permet de valider les deux modèles sur le dataset Student Performance.

Au contraire avec l'analyse visuelle (présenté dans la présentation, slide 7), la nationalité

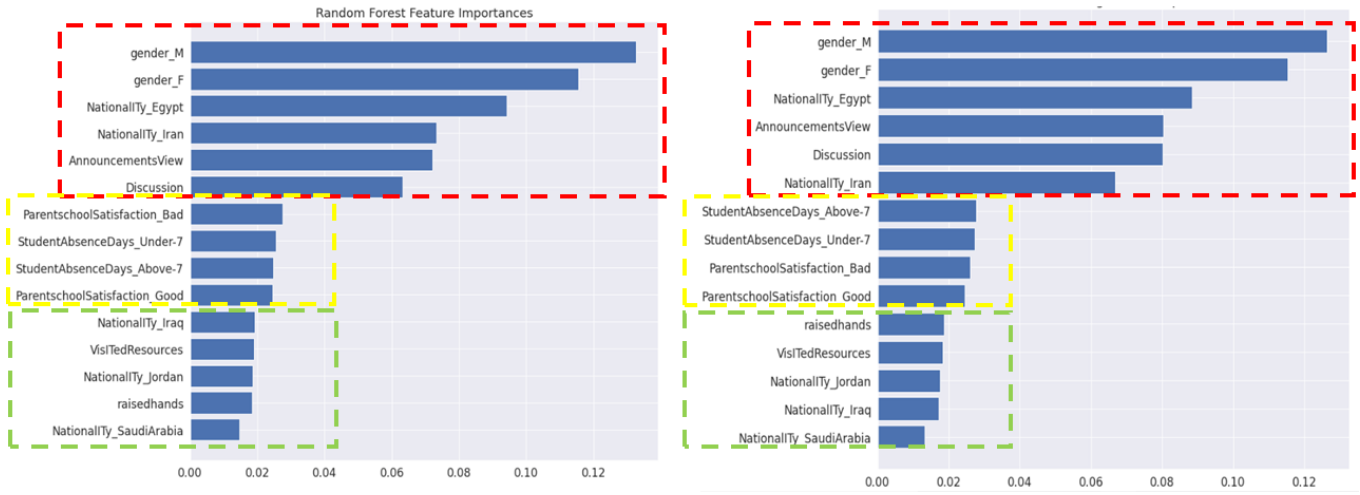


Figure 8 : Matrice de confusion obtenue par RandomForestClassifier sur le test set. Les classes sont encodées : 0 représente classe H, 1 pour L, 2 pour M

III. Conclusion

Dans ce projet, j'ai étudié le dataset Student Performance, qui est collecté en 2016, afin d'évaluer les résultats d'étude des étudiants en tenant en compte divers caractères (gendre, nationalité, relation avec parents, participation). Ces résultats d'études sont divisés en 3 classes.

L'analyse est faite en deux phases : visualisation et classification des classes. Dans la première phase, les relations entre les résultats et différents caractères sont illustré : d'être une femme, la bonne relation avec mère, participation actives au cours et au activité ont tendance d'avoir d'effet positif ; d'être un homme, absence plus de 7 jours ont une tendance négative ; la nationalité ne présente pas de relation explicite.

Dans la partie classification, le traitement de donnée a été réalisé, comme standardisation les caractères numérique, encoding les caractères catégoriques. Les classifiers choisis sont DecisionTree, RandomForest, LogisticRegression et GradientBoosting. Leurs hyper-paramètres sont raffinés via 2 étapes : paramètres résultant de RandomizedSearch sur larges ranges, puis rentré dans GridSearch. Cela permet de choisir les bons paramères pour ces modèles.

Les scores obtenus sont à l'ordre de 75%-85%, mais il y a un léger effet d'underfiting. L'ajout plus de données serait très potentiellement traiter cet effet. Le point le plus intéressant en analysant les résultats obtenus est les feature importances : (i) les 15 features les plus importances sont même dans les 2 modèles RandomForest et GradientBoosting, ce qui montrent que les modèles marchent bien. (ii) le caractère ''nationality'' ne présente pas de relation explicite en analyse visuelle, mais joue un rôle important dans les modèles de Machine Learning choisis.