8IAR403 : Apprentissage automatique pour la science des données

Devoir #1
Compréhension et préparation des données d'apprentissage

Le travail est en individuel
Date de remise le 10 février 2025

1. But

- Valider la méthodologie de conduite de projets en Machine Learning vue en cours sur une étude de cas de vente en ligne;
- Se familiariser davantage avec la problématique de la compréhension et la préparation des données d'apprentissage;
- Recours aux pipelines pour la préparation des données.

2. Étude de cas

Le contexte métier utilisé est celui d'un site d'e-commerce pour lequel on souhaite prédire les revenus que vont générer de nouveaux clients. Par conséquent, chercher le profil du bon client ! L'ensemble des données utilisées sur les clients, comme le montre la figure suivante, possède 10 000 lignes et disponible sur le site du cours dans un fichier (Customer.csv) en format CSV.

	age	pages	first_item_prize	gender	ReBuy	News_click	country	revenue
0	41	6	28.00000	Fem	False	4	China	113
1	34	4	15.50000	Fem	True	2	China	36
2	38	5	40.43173	Fem	False	7	China	111
3	20	1	44.00000	Fem	False	2	China	71
4	39	10	10.00000	Fem	True	4	China	80

Les variables (caractéristiques) décrivant les clients sont :

1. **age** : âge du client;
2. **pages** : le nombre de pages du site visité;
3. **first_item_prize** : le prix du premier article acheté;
4. **gender** : masculin ou féminin;
5. **ReBuy** : le client a-t-il acheté le premier article plus d'une fois ?
6. **New_click** : nombre de fois où le client a cliqué sur une compagne de publicité du site;
7. **country** : le pays dont provient l'adresse IP;
8. **revenue** : revenu généré par le client sur le site.

Deux datasets additionnels CountryGDP et CountryPopulation indiquent respectivement le PIB (produit intérieur brut) et la population du pays correspondant à l'adresse IP de la requête du client. Ces deux datasets sont aussi disponibles en format CSV sur le site du cours.

3. Travail à faire

Il s'agit de reproduire **les étapes (2-4)** de la démarche de conduite de projet ML vue en cours[1] dans le but de préparer les données nécessaires à l'entrainement du modèle prédictif visé dans le prochain **devoir#2**. Plus précisément, ce travail servira à préparer vos données d'apprentissage pour la suite des travaux à faire.

Pour cela, vous devez conduire deux opérations principales de nettoyage et d'enrichissement des données à travers l'écriture d'un **pipeline pour automatiser**, le plus possible, ces opérations en question :

- Nettoyage : repérer les données manquantes, aberrantes puis remédier selon les techniques appropriées. Le plus commode est d'écrire des fonctions de transformation en python.

- Enrichissement : transformer le dataset de base (Customer.csv) en rajoutant les informations liées à la population (CountryPopulation.csv) et au PIB (CountryGDP.csv) du pays de provenance de la requête. Il est fort utile d'écrire des fonctions en python et les introduire dans un pipeline.

3.1 Reproduire les étapes 2-4 de la méthodologie de conduite de projet ML (2 points)

3.2 Nettoyage des données du dataset de base Customer.csv

3.2.1 Remplacement des données manquantes (6 points)

a) Utilisez les fonctions info(), describe() ou describe(include='all') et la visualisation pour les repérer. Notez que certaines caractéristiques ne sont pas dans le type approprié. Par exemple, la variable « revenue » qui doit être numérique, elle est de type Object (texte) à cause de certaines données dont la valeur est « **unknown** ». Avant de modifier son type, vous devez

[1] A consulter le tutoriel du chapitre 2 du cours.

éliminer ou remplacer ces valeurs inconnues ou manquantes. Une fois transformée, vous pouvez utiliser, par exemple, la fonction **to_numeric()** de Pandas.

Écrivez des fonctions en python pour localiser ces données. Le but est de pouvoir les transformer en numérique afin d'utiliser par la suite l'imputation «SimpleImputer » qui manipule uniquement des variables de type numérique. Toute autre solution est la bienvenue, la création n'a pas de limite !

b) Certaines données dont les valeurs sont manquantes sont représentées par des symboles spéciaux comme par exemple (?), qu'il faudrait localiser et remplacer.

Vous pouvez transformer la fonction de changement de type des données en un **transformateur sur mesure** en utilisant la classe FunctionTransformer afin de les intégrer dans le pipeline, à l'image de la classe SimpleImputer. Le but est que cette fonction sera automatique y compris pour le jeu de test qui doit avoir la même configuration que les données d'entrainement.

3.2.2 Remplacement des données aberrantes (extrêmes, outliers) (6 points)

Utilisez la méthode « **boite à moustaches** » permettant de visualiser graphiquement le bruit, s'il y en a, dans vos données relativement à chaque variable. Pour y remédier, écrivez une fonction (un transformateur sur mesure) permettant de remplacer ou éliminer les données en question. Implanter la technique de « amplitude interquartile » basée sur les quartiles permettant de remplacer le bruit par une valeur estimée (interpolation) moyennant le premier quartile (25%), la médiane (deuxième quartile) et le troisième quartile (75%). Voir la fonction quantile() de Pandas

3.3 Enrichissement des données (6 points)

Cette étape ressemble à l'étape 3.3 d'expérimentation du tutoriel avec la combinaison de variables sans faire de calcul. Au lieu de créer de nouvelles variables dérivées, on vous demande de rajouter les deux datasets CountryPopulation et CountryGDP au dataset de base Customer.

Écrivez une fonction qui jouera le rôle de Transfomateur afin de former un dataset fusionné via une **jointure** et qui contiendra les données sur la Population et éventuellement le PIB. On peut supposer que le PIB sera l'hyperparamètre de ce transformateur. Pour cela, ce transformateur doit:

a) Nettoyer les deux dataset (CountryGDP et CountryPopulation) qu'on souhaite ajouter au dataset de base (Customer.csv).

b) Utiliser la fonction **merge()** fournit par Pandas pour faire la première jointure entre le dataframe de base (Customer) et le dataframe (CountryPopulation). Pour faire une jointure, il faut une clé commune entre ces deux dataframes. La clé commune est la variable **country** (de

type Object) qui doit être écrite de la même manière dans les deux datasets en **respectant les majuscules et les minuscules**. Renommer s'il y a lieu pour égaliser.

c) Si l'option de l'hyperparamètre est à True, cela signifie qu'on doit ajouter le PIB. Par conséquent, faire une deuxième jointure entre le **résultat de la première jointure** et le dataframe CountryGDP. Avant de faire cette jointure s'assurer aussi que la clé « country » de cette dernière est conforme, comme le premier cas.

d) Retourner le résultat de la jointure comme nouveau dataset fusionné.

4. Livrable

4.1 Fournir un seul fichier **ipynb** comprenant les étapes de **2-4 de la méthodologie de conduite de projet ML** Je vous recommande de travailler avec jupyter notebook.

4.2 Pour me faciliter la correction, vous devez afficher le résultat d'exécution de chaque cellule du notebook. Cela m'évitera de re-exécuter votre code, à chaque fois.

4.3 Commenter chacune des étapes en utilisant Markdown et en faisant référence aux questions posées.

4.4 Indiquer dans l'introduction du fichier **ipynb,** si vous avez ajouté de nouvelles fonctionnalités non-demandées et il faut les commenter

Bonne continuation !