Projet apprentissage statistique (phase 1)

Fabrice Rossi

L'objectif du projet est de mettre en œuvre des méthodes d'apprentissage statistique dans un cadre essentiellement prédictif.

1 Consignes générales

1.1 Aspects logiciels

Le projet sera réalisé de préférence en R ou à défaut en python, en utilisant toutes les bibliothèques jugées utiles pour l'application, notamment celles qui intègrent des méthodes d'apprentissage (comme *caret* en R et *scikit-learn* en python).

1.2 Données

Le projet porte sur l'analyse de deux fichiers de données (format csv, projet-app-13-learn.csv et projet-app-13-test.csv) concernant une campagne *marketing* conduite auprès d'un ensemble de clients. Chaque ligne d'un fichier décrit un client. Le fichier projet-app-13-learn.csv contient en outre les résultats de la campagne pour les clients concernées. Les variables sont les suivantes :

- `age` : âge ;
- `sex` : genre ;
- `f_name` : prénom ;
- `last name` : nom ;
- `commune` : nom de la commune de résidence ;
- `insee code` : code insee de la commune de résidence ;
- `city type` : type de la commune de résidence ;
- `department` : numéro du département de résidence ;
- `reg` : code de la région de résidence ;
- `catégorie` : code de catégorie socio-professionnelle, selon la table 1 ;
- `revenue` : salaire mensuel en équivalent temps plein (attention, cette information n'est pas disponible pour tous les clients) ;
- `cible` : résultat de la campagne codé par *success* pour un résultat considéré comme positif et *failure* dans le cas contraire, seulement dans le fichier projet-app-13-learn.csv.

Il est important de noter que Lyon, Paris et Marseille sont découpées en arrondissements et que le nom de commune est alors celui de la ville suivi de l'indication d'arrondissement.

Certaines variables seront naturellement lues par R ou Python comme des variables numériques (par exemple `reg` et `catégorie`) alors qu'elles ne contiennent pas des nombres mais des codes. Il est vivement conseillé de convertir les variables concernées au format nominal (`factor` en R, par exemple) pour faciliter la suite des traitements.

Il est aussi vivement conseillé d'enrichir les données fournies par des données annexes, notamment liées à la géographie de la France. De telles données sont fournies sur l'espace Mycourse du cours.

code	CSP
1	Agriculteurs
2	Artisans, commerçants, chefs d'entreprise
3	Cadres
4	Professions intermédiaires
5	Employés qualifiés
6	Employés non qualifiés
7	Ouvriers qualifiés
8	Ouvriers non qualifiés
9	Non déterminé
10	Étudiants
11	Chômeurs
12	Inactifs
13	Retraités

TABLE 1 – Code des catégories socio-professionnelles

2 Travail à effectuer

L'objectif final du projet est d'une part de construire un modèle prédictif, de fournir ses prévisions sur les données de test et une évaluation des performances attendues sur de nouvelles données. D'autre part, on tentera de sélectionner les variables les plus pertinentes et d'interpréter le modèle. Une analyse exploratoire minimale des données est un préalable à l'analyse prédictive.

2.1 Analyse prédictive

La tâche de prédiction consiste à déterminer le succès de la campagne marketing en fonction des caractéristiques des clients. Le critère de qualité principal retenu est le taux d'erreur, mais il est intéressant dans ce type d'application de trier les clients en fonction de leur appétence supposée à la campagne. Une évaluation basée sur la courbe ROC pourra donc être envisagée.

Le projet devra mettre en œuvre au moins deux méthodes prédictives différentes comme par exemple la régression logistique et les *random forests*.

2.2 Sélection de variables

Un autre objectif de l'analyse est de déterminer quelles sont les variables importantes pour le modèle retenu. Une discussion sur les variables est donc attendue dans le rapport à rendre, cf ci-dessous. Si les données le justifient, il est opportun de construire un modèle n'utilisant qu'une partie des variables d'origine.

2.3 Sélection des paramètres et évaluation des modèles

Les paramètres des modèles étudiés seront sélectionnés par une procédure de ré-échantillonnage adaptée. De même, on choisira le modèle le plus adapté par une méthode adaptée. On évaluera aussi les futures performances du modèle retenu d'une façon robuste.

3 Résultats attendus

3.1 Contenu du rapport

Le rapport doit contenir les éléments suivants :

1. analyse exploratoire minimale des données (statistiques univariées, dépendances, etc.) ;
2. justification du modèles prédictif choisi ;
3. description précise de la chaîne de traitement : prétraitements éventuels, ajustement des modèles, choix du modèle, évaluation de ses performances attendues (le rapport doit impérativement contenir un tableau indiquant la qualité numérique attendue pour les prévisions sur le fichier *test*) ;
4. analyse de l'importance des variables : cela peut être fait avant l'ajustement des modèles, pendant celui-ci ou après le choix du modèle final. Dans tous les cas, le rapport doit discuter de l'opportunité de construire des modèles sur une partie seulement des variables. Si c'est le cas, les prévisions finales et les performances attendues doivent concerner les modèles n'utilisant que les variables pertinentes ;
5. interprétation du modèle retenu : si cela est possible, une interprétation de la façon dont les décisions du modèle retenu sont prises fournira un complément très important au reste de l'analyse.

3.2 Remise du travail

Les étudiants doivent rendre leur travail sous forme de trois fichier regroupés dans une archive au format zip (exclusivement, pas de 7z ou autre format exotique). Les fichiers sont :

1. un rapport de quelques pages, le fichier `rapport.pdf` (format pdf uniquement), détaillant le choix des modèles, les procédures et méthodes employées, et les résultats obtenus. Il est très vivement conseillé d'utiliser le système *knitr* pour écrire ce rapport (par exemple avec markdown dans Rstudio) ou une solution équivalente en Python. Le rapport devra notamment faire apparaître explicitement les performances attendues pour les deux modèles retenus. Aucun code ne devra apparaître dans le rapport ;
2. un code R (ou à défaut python), le fichier `traitements.R`, réalisant l'intégralité des traitements demandés, avec chargement des données dans le dossier d'exécution (chemins absolus interdits) et sauvegarde des résultats dans ce dossier. Aucune intervention humaine ne doit être demandée dans ce code ;
3. un fichier `prevision.Rds` contenant une `data.frame` avec exclusivement une colonne `cible` contenant les prévisions du modèle sur les données test. Les prévisions seront données dans l'ordre du fichier *test*.