Guide dans l'application shiny

Bonjour à tous et à toutes ;

Dans ce document vous trouverez quelques consignes de comment utilisez l'application shiny concernant mon essai de l'analyse de la base de données « diamonds » qui est disponible sur R dans la librairie ggplot2.

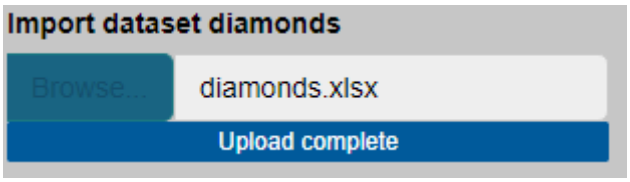
L'application est divisée en deux parties : l'entête (dashboardHeader) et le corps (dashboardBody). Je vous laisse le soin de découvrir par vous-même de quoi il en retourne.

Bonne lecture

Espace Dataset

Dans l'espace Dataset vous trouverez une page comme celle-ci :

Cliquez sur Browse pour importer la base diamonds en format xlsx (je l'ai mis à votre disposition sur mon github).

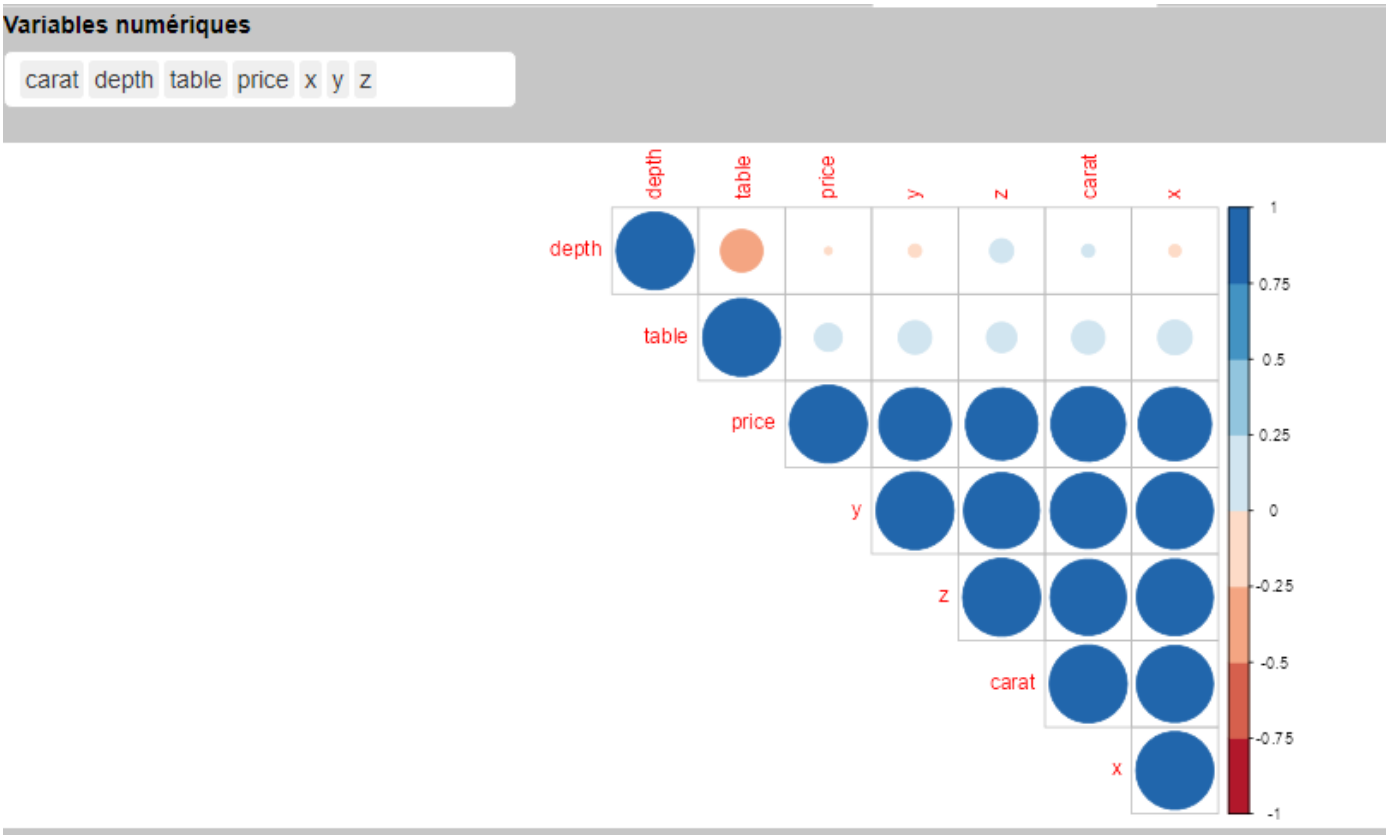


Allez dans le deuxième onglet « Catégorielle vs Prix » pour voir la dispersion des variables catégorielles (cut, color et clarity) en fonction du prix. Pour chaque variable catégorielle le graphique sera mis à jour. Ce que l'on voit par exemple pour la variable cut c'est que : Ideal et Premium ont plus de valeurs élevées que d'autres puisque leurs spectres de points sont plus condensés que celle des autres au niveau de [12000-18000[.

Dans le troisième onglet « Valeurs aberrantes » vous trouverez des boxplot ; utile pour identifier les outliers dans les valeurs numériques (carat, price, table, depth, x, y, z) de notre table. Par exemple, pour la variable depth les valeurs sont toutes concentrées autour [61.6-63[et il y a quelques valeurs qui sont considérées à l'extrême.

Dans le quatrième onglet « Distribution » vous trouverez les distributions de chaque variable numérique. Par exemple, pour celui de price la distribution est asymétrique à droite.

Dans le cinquième onglet, vous pourriez observer la matrice de corrélation (produite par la fonction `corrplot()`) pour les variables numériques.



Espace Prédictions des prix

Un exemple d'utilisation de l'apprentissage supervisé est la prédiction des prix puisqu'on connait déjà les étiquettes des variables d'entrées et de sortie.

Dans le premier onglet on va implémenter une régression linéaire en utilisant la fonction `lm()` de R. Dans notre modèle : les entrées sont toutes les variables présentes sauf le prix et la sortie est le prix. Ensuite le graphique met en relation les valeurs réelles en fonction des valeurs prédites (la couleur selon clarity est subjective). On remarque vite que le score R^2 est assez élevé mais les valeurs prédites ne sont pas assez fiables (la valeur maximale prédite est de 40000 alors la valeur maximale réelles est seulement de 18823).

Dans le second onglet on va utiliser la méthode XGBoost (Extreme Gradient Boosting) avec la fonction `xgboost()`. XGBoost construit un arbre de décision où les caractéristiques mises en entrées seront prises en compte dans chaque nœud. On a choisi 390 itérations arbitrairement pour ajuster au mieux notre modèle de prédiction (vous pourriez le changer selon votre préférence et obtenir des meilleurs résultats). Dans le graphique on a pu noter une certaine cohérence entre les valeurs réelles et prédites (max valeur prédites 18803 environ).

Dans le troisième onglet on retrouve la méthode Random Forest implémentée grâce à la fonction `randomForest()` où le paramètre `ntree` signifie le nombre d'arbre de décision que l'on souhaite. Chaque arbre fournit une prédiction qui au final sera combiné pour avoir la meilleure prédiction possible. Ses prédictions sont assez cohérentes aux valeurs réelles de la base.

Dans le quatrième onglet vous pourriez selon votre télécharger les résultats (prédictions vs réelles) sous forme de tableau excel. Par la suite il serait intéressant de voir les écarts entre les valeurs prédites et les valeurs réelles dans ce tableau. Comme vous le verrez le score R^2 est assez similaire pour les trois modèles mais la prédiction diffère pourquoi ? En fait R^2 mesure à quel point un modèle de régression explique la variance des données observées, c'est-à-dire à quel point vos variables d'entrées arrivent à expliquer la sortie.

Dans notre cas, il serait mieux d'utiliser des arbres de décision plutôt qu'une régression linéaire compte tenu des résultats obtenus.

Espace Clustering

Dans le premier onglet vous trouverez les résultats (graphiques) de l'ACP sur 0.4% (sélectionné aléatoirement) de la base diamonds. Même si vous n'aviez pas importé la base excel ceci reste accessible. La carte des variables permet de conclure : clarity, cut, table expliquent l'axe 2 alors que x, y, z, carat et price expliquent l'axe 1. En se fiant aux critères du coude 3 axes sont à retenir plutôt que 10. Avec 3 axes on retient 78.3% de l'information ce qui est déjà pas mal. Le graphique concernant la contribution des variables aux dimension 1-2 nous permet de dire que : x, y, z, carat, table, price ont une forte contribution.

Dans le second onglet il est question de continuité puisque grâce à l'information de l'ACP on peut créer 3 classes de diamants (en utilisant la méthode KMeans) qui ont des caractéristiques plus ou moins similaires. Vous jugeriez cela dans le troisième résultat qui recense les moyennes de chaque variable pour une classe donnée.

Bon on a fait le tour, j'espère que ceci vous aidera à mieux comprendre la base diamonds et à vous familiariser avec les concepts d'apprentissage supervisé (prédiction des prix) et non-supervisé (clustering).

Merci de votre attention !