Projet Python Data Analysis

Online Shopper Intention predictions

ESILV LÉONARD DE VINCI | ÉCOLE D'INGÉNIEURS PARIS-LA DÉFENSE

Ali YOUSSEF
Antoine THIBAULT
A5

Le dataset

- Notre dataset concerne les intentions d'achats sur des personnes qui fréquentent un site e-commerce.
- Il comporte les informations des utilisateurs de ce site sur environ 1 an.

Les points importants de notre dataset :

- 12 330 sessions dont 85% sans achat
- 18 colonnes :
  - 10 numériques
  - 8 catégoriques
- Aucune information manquante dans le dataset

La dataset

Chaque session possède un attribut “Revenue” qui est booléen et qui permet de savoir si l’utilisateur a effectué un achat ou non durant sa session sur le site.

Le but est donc de réussir à prédire à l’aide d’un modèle si un utilisateur va convertir sur le site ou non à l’aide des autres informations.

Nous allons donc utiliser la data-visualisation pour essayer de ressortir des variables significative et comprendre les attitudes des utilisateurs.

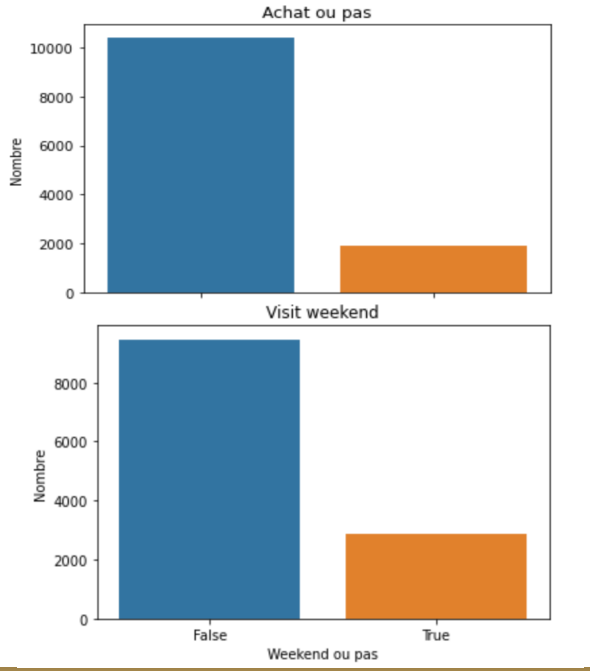
Les informations du dataset

- Month : Le mois ou la visite sur le site a été effectuée
- SpécialDay : Permet de savoir si l'achat a été effectué proche d'un date comme la st valentin ou noel
- OperatingSystem : Le système d'exploitation que possède l'utilisateur
- VisitorType : Le type de visiteur (nouveau ou ancien)
- Browser : Le navigateur de l'utilisateur
- Weekend : Afin de savoir si la visite a été effectué le week-end
- Region : Le numéro de région de l'utilisateur
- TrafficType : Le type de traffic
- Revenue : Qui permet de savoir si la personne a effectué un achat sur le site ou non

Les informations du dataset

- BounceRates / ExitRates : Pourcentage sur le taux de rebond et le taux de sortie (permet d'obtenir des informations sur l'interet de l'utilisateur)
- PageValue : C'est le nombre moyen de page visité par l'utilisateur avant d'acheter
- Administrative / Informational / ProductRelated : Permet de savoir le nombre de page visité par l'utilisateur dans chaque type
- Administrative_duration / Informational _duration/ ProductRelated_duration : Permet d'obtenir le temps passé dans chaque type de page du site

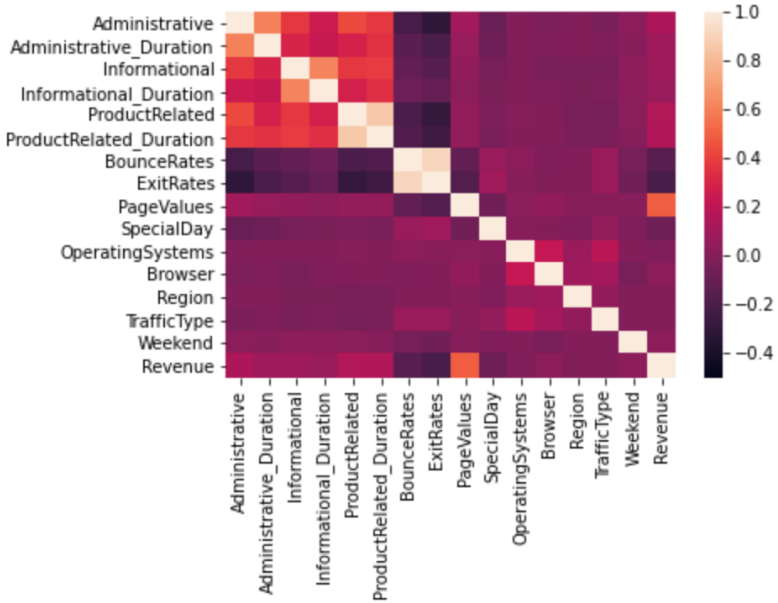
Data-Visualisation



- Cette visualisation permet de voir que la plus part des personnes ayant visité le site (10 000) n'ont pas effectué d'achat

- On voit également que la plus part de visites sur le site se font durant la semaine (un peu moins de 10 000)

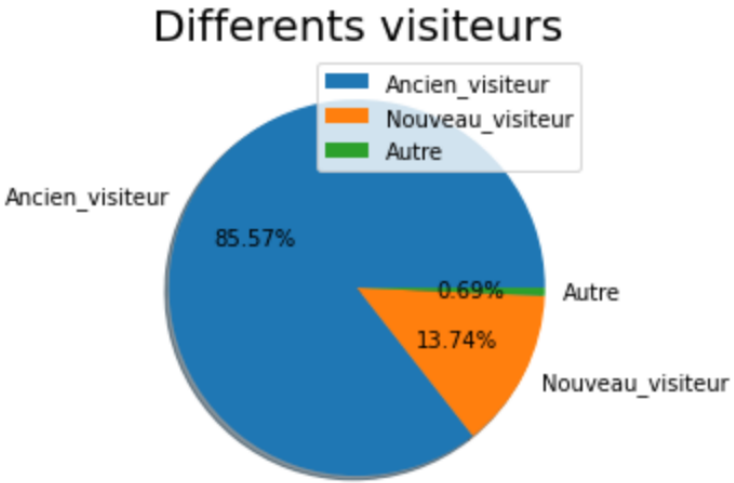
Data-Visualisation



A l'aide de cette matrice on peut voir que déjà plusieurs variables sont corrélées entre elles :

- BounceRates et ExitRates
- ProductRelated et ProductRelated_Duration
- Informational et Informational_Duration
- Administrative et Administrative_Duration

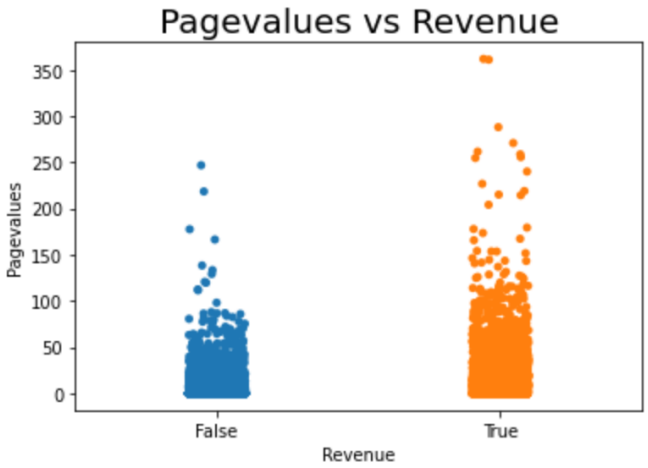
Data-Visualisation



- Les personnes venant sur le site sont principalement des anciens visiteurs, très peu de nouveaux visiteurs.

  Les sessions sont donc principalement (80%) issuent d'anciens clients, de rettargetting ou de personnes revenant et n'ayant pas convertis.

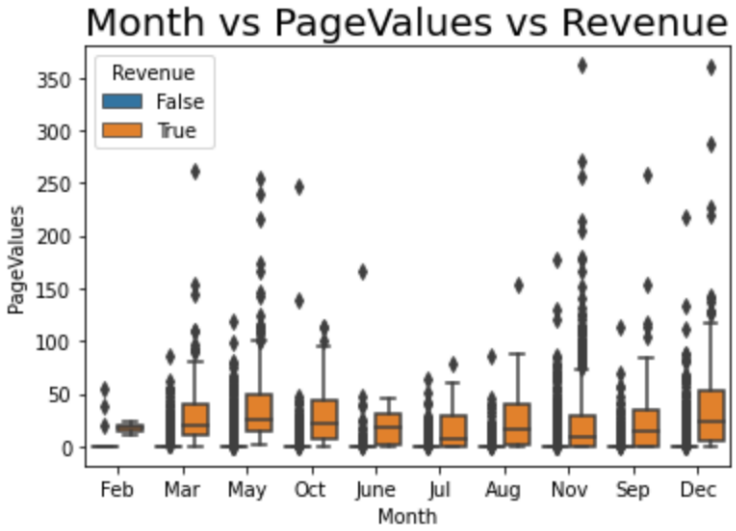
Data-Visualisation



- Ce graphique permet de mettre en relation PageValues et Revenue.

  Il nous permet de voir que concernant les personnes ayant convertis sur le site ont une pageValue pouvant monter bien plus haut que les personnes n'ayant pas acheté.

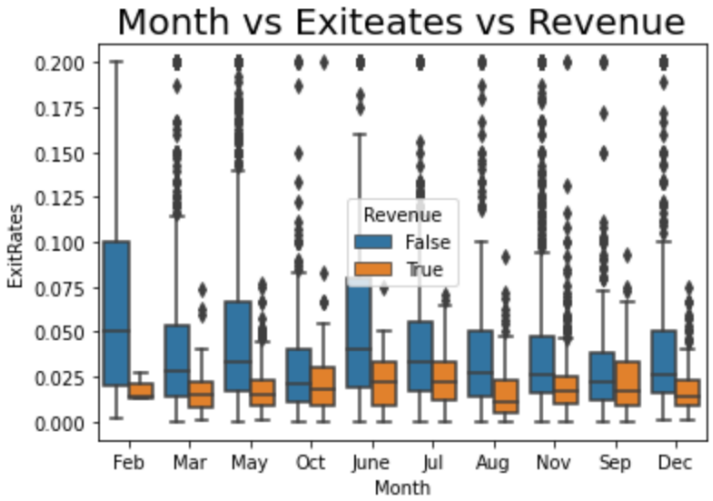
Data-Visualisation



- Celui-ci reprend les mêmes information que le graphique précédent en ajoutant en abscisses les mois nous permettant de voir les mois ou ils y a eu le plus de conversion.

  Ici nous voyons très clairement que le nombre PageValue est significatif, plus celui-ci est haut plus la personnes achètera potentiellement sur le site.

Data-Visualisation



- Ce graphique prends en compte les variables Month, ExitRates ainsi que le Revenu.

  Nous pouvons voir qu'un ExitRates faible peut potentiellement plus mené à une conversion sur le site.

Model et prédiction

Le but étant de savoir si l'utilisateur va acheter ou non : 0 pas d'achat et 1 l'achat est effectué.

Après avoir effectué de la visualisation et trouver des variables significative ou corrélées, nous avons découpé le dataset en train et test.

Nous avons utilisé 3 différents models :

- Random Forest : 0.8934
- Logisitc Regression : 0.8715
- Decision Tress : 0.8572

API

Nous avons décidé de développer une API Django avec un endpoint ‘/predict’.

Pour tester le fonctionnement de l’API, il suffit d’envoyer une requête http sur ‘/predict’ à l’aide d’un outil type Postman ou Insomnia.

Dans le body de la requête, il faut inclure les différents attributs représentant un visiteur du site (vous trouverez deux objets prêts à l’utilisation dans le ReadMe.md du repository GitHub).

L’API retourne alors un objet contenant la prédiction de résultat, ainsi que le score de confiance.