



M2-Économétrie Appliquée et Statistiques

IAE Nantes – Université de Nantes

# Predict Futures Sales

Compétition kaggle

SVM et réseau de neurones

**Auteurs** :

André ANGWE

Sory BARRY

Année universitaire 2021-2022

Résumé

Ce rapport est un résumé des étapes effectuées pour notre projet du cours SVM et réseau de Neurones du Master 2 EKAP. Le but de ce projet est de prédire les ventes futures des magasins. Afin de mieux comprendre ce rapport, nous vous recommandons de le lire avec le notebook qui contient tous les codes effectués pour l'analyse exploratoire des données et la modélisation. Nous avons utilisé divers modèles tout au long de ce rapport.

## Description des variables :

Notre base d'entrainement est composée de 6 variables

Date: date au format jj/mm/aaaa

Date_block_num: un numéro de mois consécutifs, utilisé pour plus de commodité. Ex : Janvier 2013=0, février 2013=1 ….

Shop_id: identifiant unique d'une boutique

Item_id: identifiant unique d'un produit

Item_price: prix actuel d'un article

Item_cnt_day: Nombre de produits vendus (variable d'intérêt)

## Partie Exploration des données (EDA)

Après avoir importé nos différentes bases de données, nous avons procédé à

l'exploration de notre base de données d'entrainement. Nous avons commencé par observer que notre base de données d'entrainement (sales_train) est composée de 2935849 observations et 6 variables. La base de données ne contient aucune donnée manquante. Nous avons ensuite combiné la base des ventes et la bases items contenant les articles des magasins par l'identifiant en commun (item_id).

Nous avons par la suite converti la date au format aaaa/mm/JJ afin d'utiliser la bibliothèque Datetime. Nous avons observé que la base de données est constituée des données allant du 01/01/2013 au 31/10/2015.

Nous avons ensuite rajouté dans la base ''sales_train'' les variables month, year et day qu'on a pu extraire à partir de la variable date.

À l'aide des bibliothèques matplotlib et seaborn nous avons afficher les graphiques du nombre d'article vendu par mois pour l'ensemble des magasins, les ventes réalisées chaque année pour l'ensemble des magasins, les prix des articles des magasins. Les graphiques nous ont permis de constater que les ventes des magasins augmentent beaucoup à partir du mois de novembre, ce qui peut être dû au fait de l'approche des fêtes de noël. Nous constatons également une augmentation des prix des articles à partir du mois d'août, cela peut être dû au fait que les familles préparent la rentrée.

Nous avons également représenté en diagramme les 10 catégories d'articles les plus vendus par mois dans l'ensemble des magasins. Nous avons ensuite déterminé 60 boutiques, et nous avons ensuite affiché les histogrammes du nombre d'articles vendus dans chaque boutique.

A l'aide de l'affichage du graphique de corrélation, nous avons pu observer les différentes corrélations entre nos différentes variables.

Nous avons ensuite identifié des prix et des ventes négatifs dans notre base de données. Ce qui nous a emmené à les supprimer de la base. Ces données sont probablement dû à des retours d'articles par des clients.

Nous avons affiché par la suite les boites à moustaches des variables item_price et item_cnt_day, nous remarquons que ces deux variables présentent des points atypiques que nous supprimons de la base de données.

Nous avons finalement calculé la somme des ventes totales par mois de tous les magasins et nous avons représenté les ventes sur un graphique interactif permettant de voir le nombre d'articles vendus chaque mois.

## **Modélisation**

Présentation des différents modèles utilisés

Les différents Modèles utilisés

Light GBM

Light Gradient Boosting est un cadre de boosting de gradient qui utilise un algorithme d'apprentissage basé sur les arbres. Le light gbm fait croître les arbres verticalement alors

que d'autres algorithmes font croitre les arbres horizontalement, ce qui signifie que le Light GBM fait croître l'arbre par feuille alors que les autres algorithmes font croître l'arbre par niveau. Light GBM peut traiter des données de grandes tailles et nécessite moins de mémoire pour fonctionner.

Nous avons donc réalisé un modèle light GBM pour la régression avec des paramètres de bases tels que le boosting, qui définit le type d'algorithme qu'on veut exécuter. Nous avons gardé la valeur gbdt qui sont des arbres de décision traditionnels à boosting de gradient. Nous avons fixé le taux d'apprentissage à 0,03. Le taux d'apprentissage détermine l'impact de chaque arbre sur le résultat final. Nous avons gardé 32 feuilles dans l'arbre complet. Nous avons décidé de booster 200 arbres.

Ce modèle nous a donné des faibles taux d'erreur et une bonne qualité d'ajustement du modèle. Nous avons calculé pour chaque modèle l'erreur absolue moyenne, la moyenne des erreurs au carré et la racine carrée de ces erreurs. (MAE, MSE, RMSE).

Nous avons obtenu avec notre modèle un MAE de 0.367 et un RMSE de 1,78.

Afin d'obtenir des meilleurs résultats, nous avons essayé de faire une grille avec la fonction GridSearchCV de Sklearn. Dans cette grille nous avons mis plusieurs valeurs des paramètres cités plus haut afin que ça nous sélectionne à l'aide d'une validation croisée, les meilleures valeurs pour nos différents hyperparamètres.

La grille a bien été créée mais malheureusement on n'a pas pu l'appliquer à nos données. Le modèle mettait du temps à passer et au bout de quelques heures nous avons dû abandonner. C'est une procédure qui prend énormément de temps, car pour chaque arbre ça va chercher les meilleurs paramètres. On aurait pu avoir des meilleurs scores si on avait réussi à appliquer notre modèle sur nos données.

Long Short Term Memory (LSTM)

Long Short Term Memory ou mémoire à long terme et à court terme sont un type spécial de réseaux de neurones récurrents capables d'apprendre les dépendances à long terme. Il ont été introduits par Hochreiter et Schmidhuber en 197, et ont été par la suite affinés et popularisés à travers plusieurs travaux. Ils fonctionnent extrêmement bien sur une

grande variété de problèmes et sont maintenant largement utilisés. Les LSTM sont une architecture des RNN utilisées dans le domaine de l'apprentissage profond. Les LSTM possèdent des connexions de rétroaction.

Les réseaux LSTM sont bien adaptés la classification,au traitement et aux prédictions basées sur les données de séries temporelles, car il peut y avoir des décalages de durée inconnue entre des évènements importants dans une série temporelle. Les LSTM ont été développés pour résoudre le problème de gradient de fuite qui peut etre rencontré lors de l'entraînement des RNN traditionnels.

**LSTM**

Afin de faire la première modélisation LSTM, nous avons agrégé au niveau mensuel les ventes. Nous préparons donc une base vide composée des 34 colonnes : 33 colonnes constituées par les numéros de mois et 1 colonne par la variable ID. Nous préparons ensuite les données : X_train qui contient les 33 premières colonnes, Y_train la dernière colonne (la colonne à prédire : mois 34). Nous trouvons un MSE = 6,15 soit un RMSE = 2,48.

**Conclusion :**

Afin de conclure, ce projet a été vraiment intéressant à réaliser car il nous a permis de découvrir certaines méthodes de modélisation qu'on ne connaissait pas forcément très bien. Nous avons pu améliorer nos capacités en visualisation des données sous python.

Les véritables difficultés dans la réalisation de ce projet ont été au niveau de la puissance des nos ordinateurs. Nous n'avons pas pu aller au-delà de nos capacités car les modèles tournaient pendant des heures, voir des jours. Par exemple pour le LightGBM nous n'avons pas pu appliquer notre grille sur nos données d'entraînement, ce qui devait normalement améliorer la performance de notre modèle.