Machine learning: Business case

Présentation Business:

“ Rossmann Store Sales ”

Présenté par :

Ahmed BEJAOUI

Aymen DABGHI

Aymen MEJRI

Med Rostom GHARBI

Salma JERIDI

Plan

1. Contexte et objectifs
2. Présentation des données & insight métier
3. Méthodologie & Résultats du projet Data Science
4. Recommandations & Next steps

1.Contexte et objectifs

Contexte

- Rossmann est une entreprise et une chaîne allemande de distribution de drogueries créée en 1972 par Dirk Roßmann.
- Rossmann a plus de 3000 magasins en 7 pays européens différents : Allemagne, Albanie, Pologne, République tchèque, Turquie et Hongrie.
- Effectif : 51 000 personnes.
- Chiffre d'affaires : 9 000 000 000 euros en 2017.
- Concurrents principaux : DM-Drogerie Markt et Schlecker.

ROSSMANN

Objectifs

- Prévoir la vente quotidienne sur les 1115 magasins Rossman situés dans toute l'Allemagne, 6 semaines à l'avance.

Impact de cette solution :

- Meilleure gestion des horaires du personnel.
- Prévoir suffisamment de temps pour que les directeurs des magasins se concentrent sur les clients et leurs équipes.
- Augmenter l'efficacité des employés.

2. Présentation des données & insight métier

Présentation des données

Dans ce problème, on dispose de 3 datasets:

- **Train_set** : Représente l'historique des données de ventes quotidiennes de 1115 magasins à partir du 01/01/2013 au 31/07/2015. Cette partie des données compte environ 1 million d'entrées et comprend de multiples variables explicatives qui pourraient avoir un impact sur la vente.
- **Store_set** : Représente des informations supplémentaires sur les magasins.
- **Test_set** : Représente des données similaires à la Train_set (à l'exception de "customers" et "sales") pour les 6 semaines suivantes.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1017209 entries, 0 to 1017208
Data columns (total 9 columns):
Store            1017209 non-null int64
DayOfWeek        1017209 non-null int64
Date             1017209 non-null datetime64[ns]
Sales            1017209 non-null int64
Customers        1017209 non-null int64
Open             1017209 non-null int64
Promo            1017209 non-null int64
StateHoliday     1017209 non-null object
SchoolHoliday    1017209 non-null int64
```

Tain_set

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 41088 entries, 0 to 41087
Data columns (total 8 columns):
Id               41088 non-null int64
Store            41088 non-null int64
DayOfWeek        41088 non-null int64
Date             41088 non-null datetime64[ns]
Open             41077 non-null float64
Promo            41088 non-null int64
StateHoliday     41088 non-null object
SchoolHoliday    41088 non-null int64
```

Test_set

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1115 entries, 0 to 1114
Data columns (total 10 columns):
Store                        1115 non-null int64
StoreType                    1115 non-null object
Assortment                   1115 non-null object
CompetitionDistance          1112 non-null float64
CompetitionOpenSinceMonth    761 non-null float64
CompetitionOpenSinceYear     761 non-null float64
Promo2                       1115 non-null int64
Promo2SinceWeek              571 non-null float64
Promo2SinceYear              571 non-null float64
PromoInterval                571 non-null object
```

Store_set

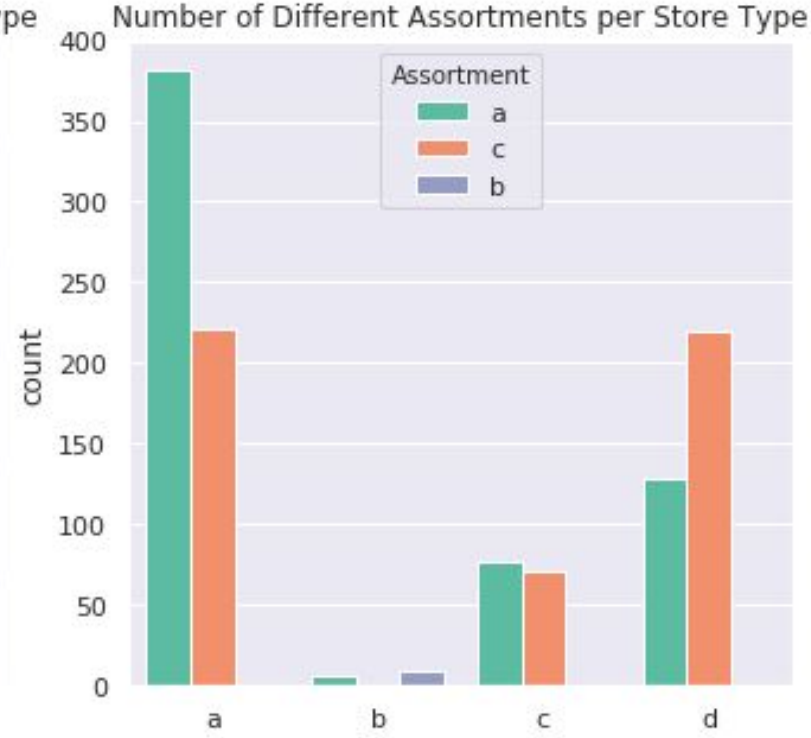
1. Types de magasins

Les magasins de type ‘a’ dominent en terme de présence sur le marché et de ventes réalisées ...

1. Types de magasins

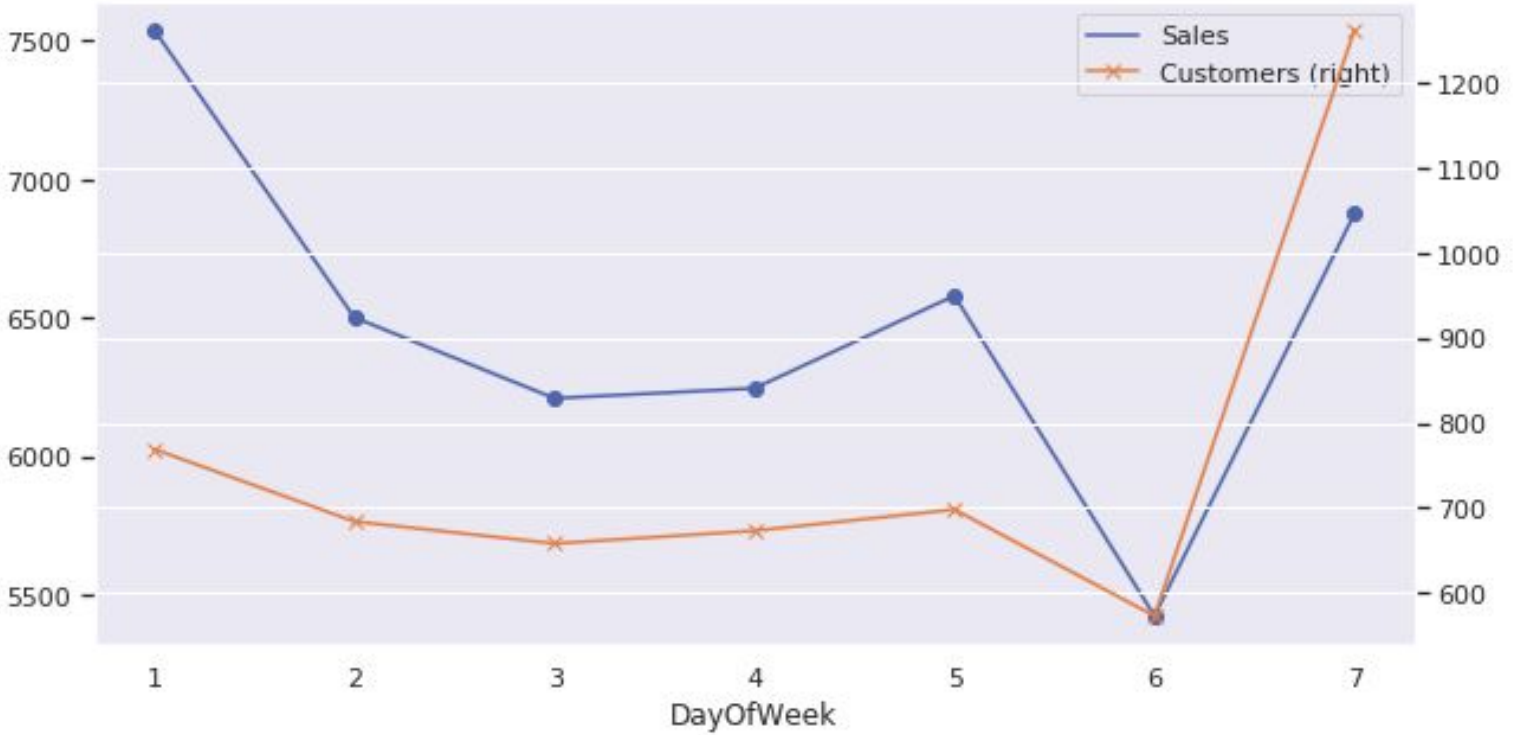
Par contre, ce sont les magasins de type ‘b’ qui possèdent la quantité de ventes moyenne et le nombre de clients moyens par magasin les plus élevés !

1. Types de magasins



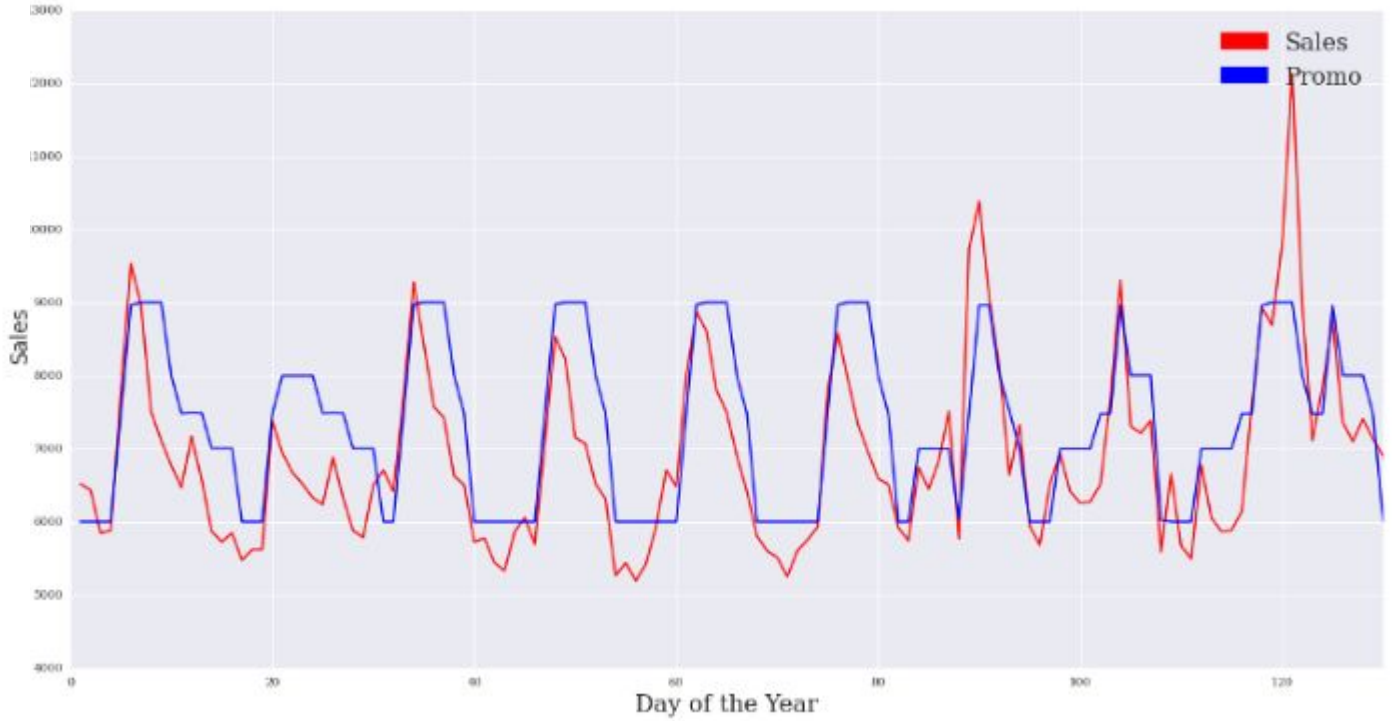
Cependant, la moyenne des dépenses des clients révèlent autre chose ...

2. Jours de la semaine



Comportement inverse pendant les dimanches

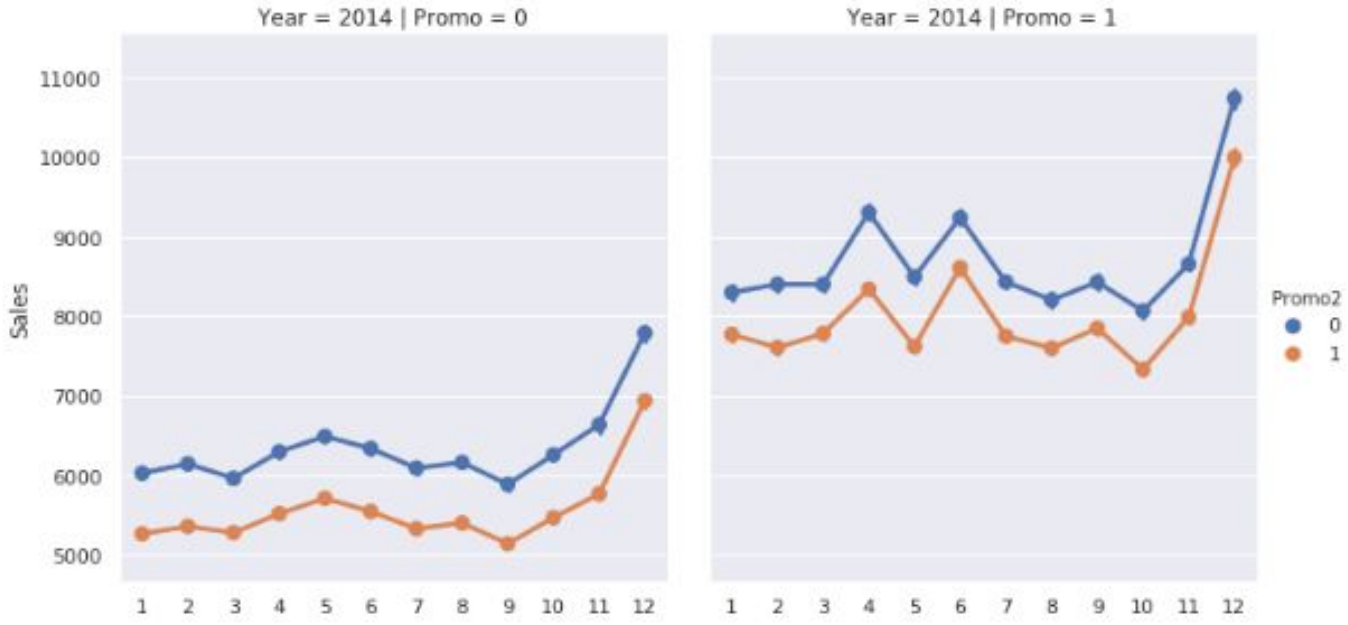
3. Promotions



Les ventes et les promotions sont fortement corrélées.

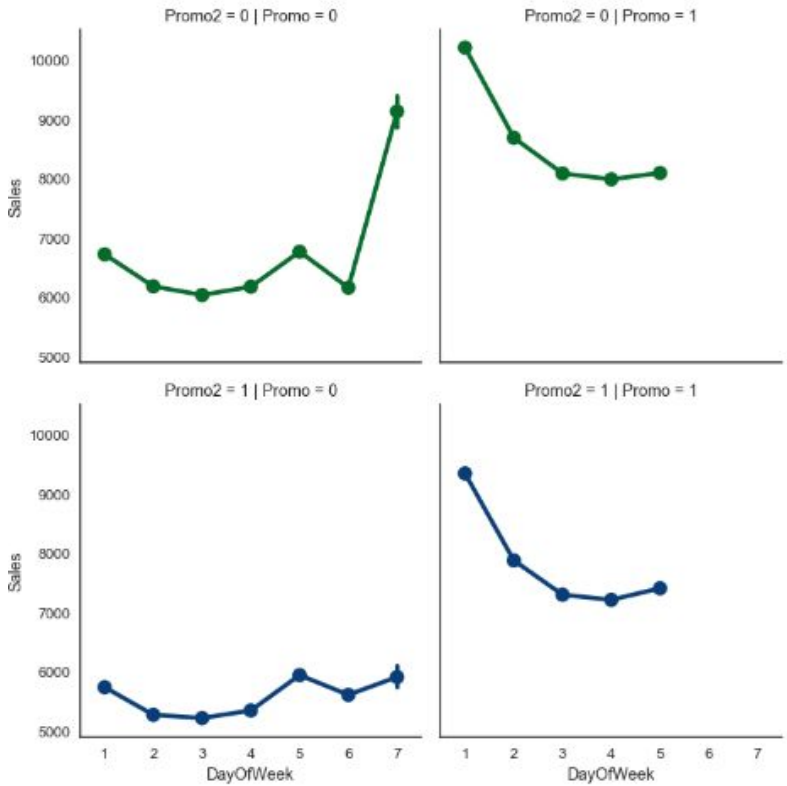
la vente moyenne est 30% plus importante lorsque le magasin propose une promotion.

3. Promotions



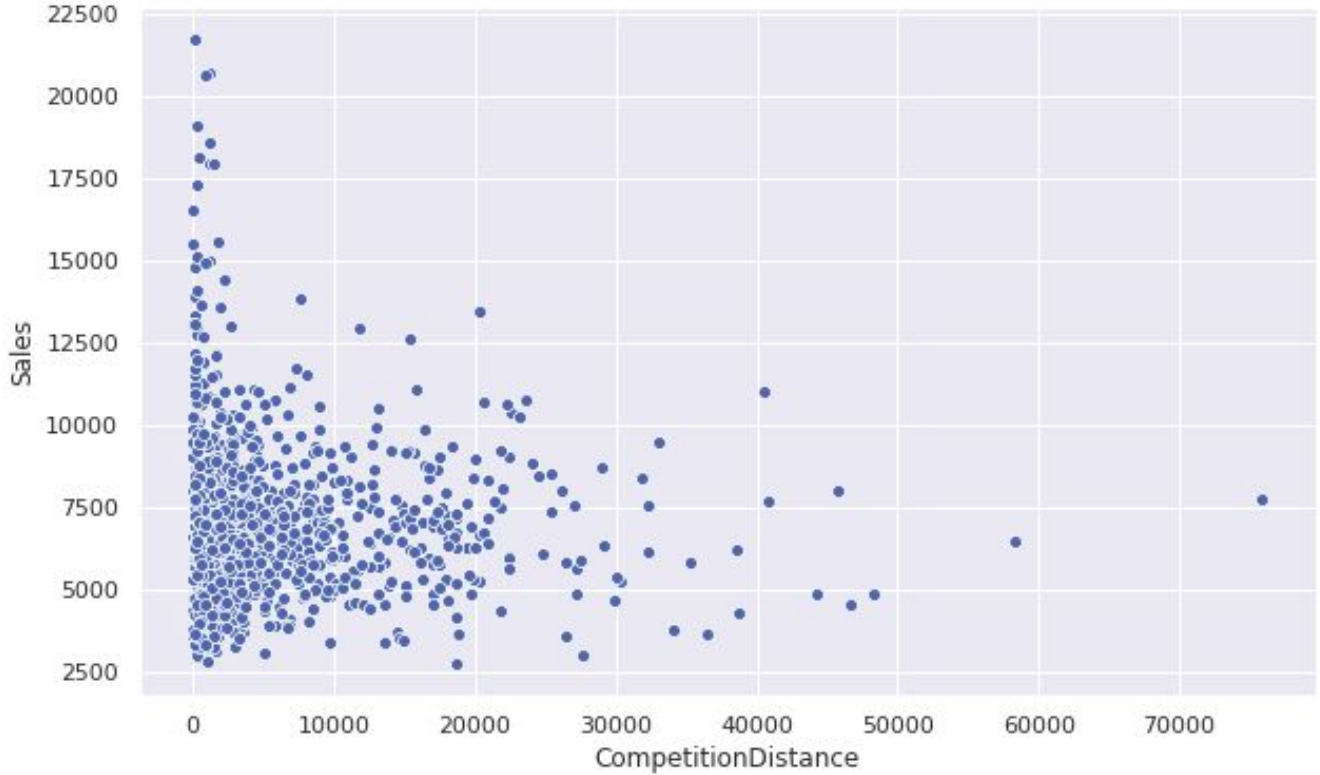
Les promotions augmentent les ventes. Mais les magasins qui ne participent pas à des promotions consécutives engendrent plus de profit !

3. Promotions



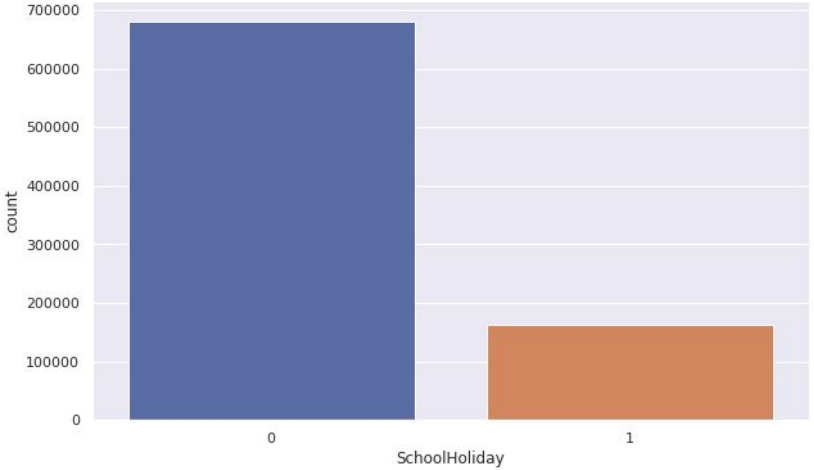
Promo2 n’a pas vraiment un impact significatif sur les ventes, ce qui confirme encore notre hypothèse.

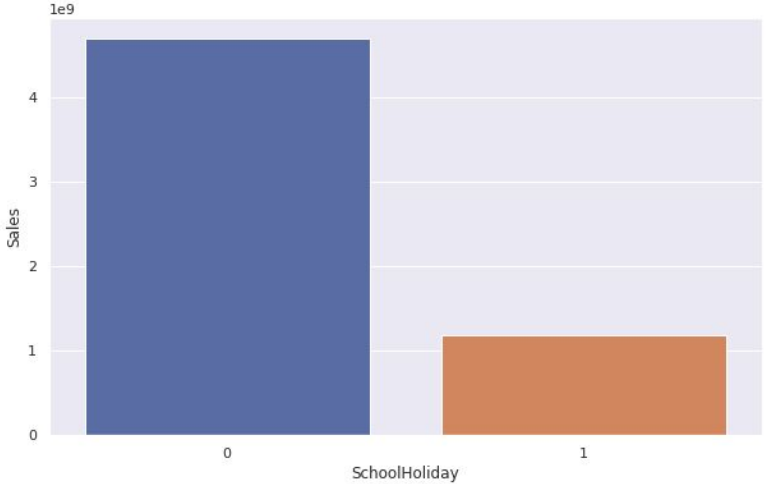
4. Compétition



Plus la compétition est proche, plus les ventes sont élevées ?!

5. Vacances scolaires





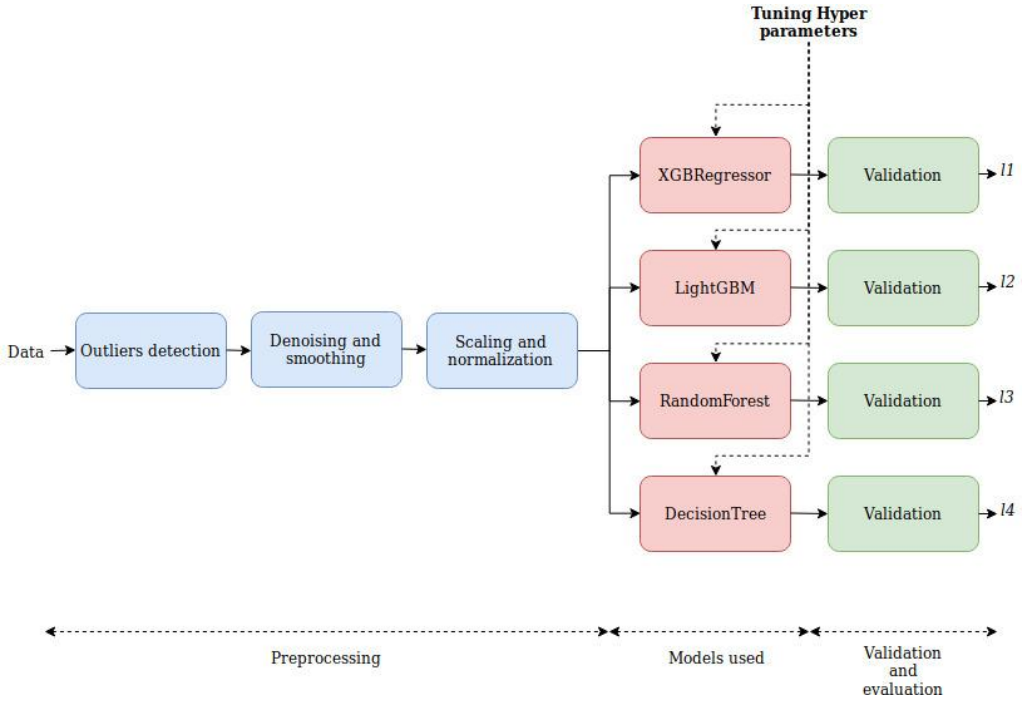
25.06% du total des ventes réalisées en 19.35% du nombre des jours d'ouverture

Pour finir ...

- Les magasins du type ‘a’ sont les plus présents sur le marché et ils réalisent le plus des ventes.
- Les magasins du type ‘b’, malgré leur nombres réduits, attirent le plus de clients et réalisent le plus de ventes en moyenne.
- Les magasins du type ‘d’ ont la moyenne de dépenses des clients la plus élevée, c’est grâce au type ‘c’ de produits qu’ils vendent le plus.
- Pendant les dimanches, le nombre des clients augmentent remarquablement, mais sans effet clair sur les ventes (Phénomène de window-shopping).
- Lancer des promos dans les magasins les moins performants avec une Competitiondistance moins élevée.
- Les magasins ouverts pendant les vacances scolaires sont bien performants.

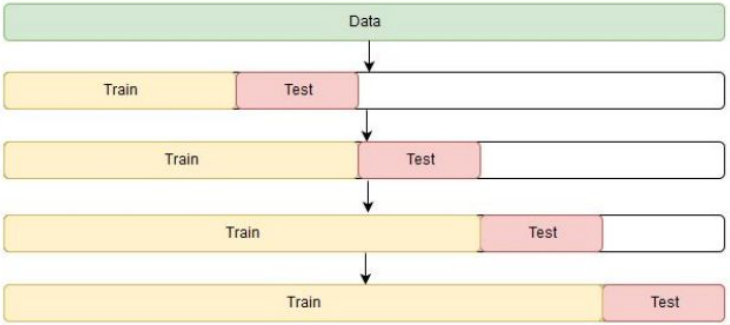
3. Méthodologie

Choix du modèle et des paramètres



- **Modèle choisi** : XGBRegressor
- **La méthode de la détermination des paramètres:** Estimation Bayesienne :
  - Une approche rapide
  - Une approche précise

Stratégie de validation



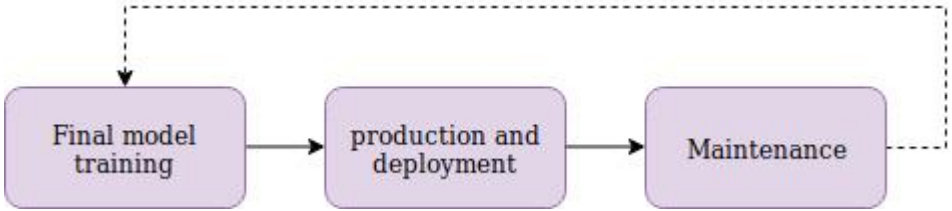
Technique adoptée:

- Effectuer une validation croisée en divisant notre donnée en des paquets d'une durée de 2 mois

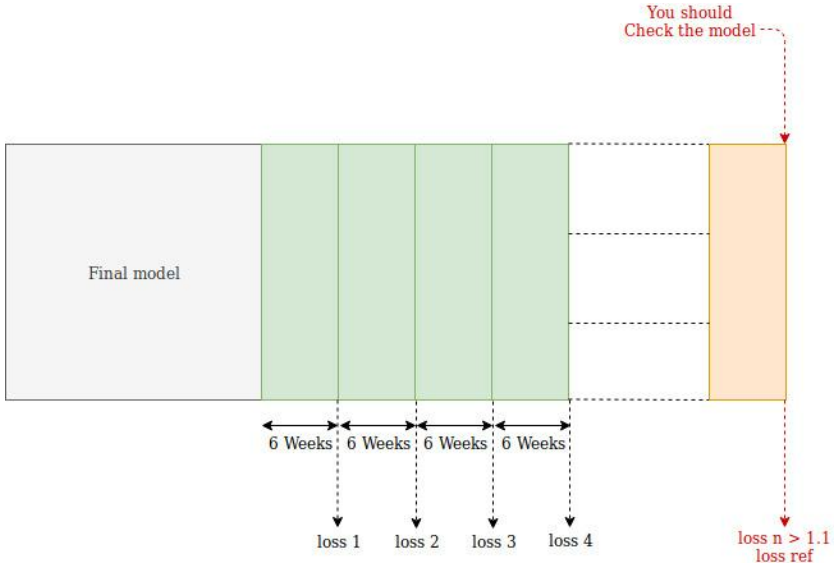
- **Choix du loss:** RMSE (Root Mean Square Error)

- ***Justification:***
  - Pénaliser davantage les grandes erreurs de prédiction.
  ---> Contrainte de stockage.

  - Sensible au changement de distribution entre le training et le test.
  ---> facilite la détection du changement de distribution après la mise en production de l'algorithme.


Final model training
production and deployment
Maintenance

- **Maintenance automatique**



- **Maintenance à la demande du client:**

  - Introduction de nouveaux produits.
  - Changement de stratégie (exemple: expansion à l'international).

4. Recommandations & Next steps

Recommandations au niveau des variables explicatives :

Fournir des données sur :

- Les différents prix des différents produits.
- La localisation du magasin (Ville, Population, …).
- Le nombre d’employés de chaque magasin.
- Le stock de chaque magasin.
- La météo.

Recommandations au niveau des variables explicatives :

- Analyses de sentiments des différents clients sur les différents magasins :

Recommandations au niveau des variables explicatives :

- Prendre en considération les horaires d'affluence fourni par Google Maps :

Recommandations au niveau de modèle

Algorithm 1 : FIND_CHANGE

```
1:  for i = 1...k do
2:      c_0 ← 0
3:      Window_{1,i} ← first m_{1,i} points from time c_0
4:      Window_{2,i} ← next m_{2,i} points in stream
5:  end for
6:  while not at end of stream do
7:      for i = 1...k do
8:          Slide Window_{2,i} by 1 point
9:          if d(Window_{1,i}, Window_{2,i}) > α_i then
10:             c_0 ← current time
11:             Report change at time c_0
12:             Clear all windows and GOTO step 1
13:         end if
14:     end for
15: end while
```

Détection des différents changements dans la distribution des données

Next Steps

- Appliquer l’algorithme de détection de changements de distribution des variables explicatives et re-entrainer le modèle en cas de besoin.
- Essayer d’avoir plus de variables explicatives qui pourront être importants pour la prédiction des ventes des différents magasins.
- Tester le nouveau modèle après avoir fait les différents changements mentionnés ci-dessus et le comparer avec l’ancien modèle.

“Merci pour votre attention.”