



MODULE D'APPRENTISSAGE
STATISTIQUE - MODÉLISATION
STATISTIQUE

Projet d'Apprentissage Automatique

Lilian LECLERC, Samuel SABO, Adrien BOLLING, Abdoulaye SAKHO,



Présentation
du problème

Présentation des données

Trois fichiers de données : **train.csv**, **test.csv**, **store.csv**

Les différentes variables présentes dans test.csv et train.csv sont les suivantes :

- Store : un ID unique propre à chaque magasin
- DayOfWeek : le jour de la semaine (numéroté de 1 à 7)
- Date : la date du jour (format YYYY-MM-DD)
- Sales : uniquement présente dans test.csv (en effet, c'est la variable à prédire dans train.csv), représentant le nombre de ventes effectuées par le couple (Store, Date)
- Customers : le nombre de clients dans un magasin à un jour donné
- Open : un indicateur permettant de savoir si le magasin est ouvert (1) ou fermé (0)
- Promo : indique si un magasin effectue ou non des promotions ce jour
- StateHoliday : indique si des vacances nationales ont lieu ou non (0 ou 1). En général tous les magasins, hormis quelques exceptions, sont fermés pendant ces vacances. De plus, toutes les écoles sont fermées lors des jours fériés et des week-ends. On suit la notation suivante : jour férié (a), vacances de Pâques (b), vacances de Noël (c), pas de vacances (0)
- SchoolHoliday : indique si le couple (Store, Date) est affecté ou non par la fermeture des écoles publiques

Ensuite, voici les différentes variables de store.csv :

- Store : une ID unique propre à chaque magasin
- StoreType : différencie 4 types de magasins (a, b, c ou d)
- Assortment : décrit un niveau d'assortiment (basique (a), supplémentaire (b), étendu (c))
- CompetitionDistance : distance en mètres par rapport au concurrent le plus proche
- CompetitionOpenSince[Month/Year] : donne l'année et le mois approximatifs depuis lequel le premier concurrent a été ouvert
- Promo2 : Promo2 est une promotion continue et consécutive pour certains magasins, si le magasin participe (1) sinon (0)
- Promo2Since[Year/Week] : indique l'année et la semaine calendaire correspond au début de la participation à cette promotion
- PromoInterval : décrit les intervalles consécutifs pour lesquels Promo2 se déroulent, en nommant les mois durant lesquels la promotion est relancée. Par exemple, "Feb,May,Aug,Nov" signifie que pour chaque année donnée, Promotion2 commence à ces périodes

Restructuration primaire

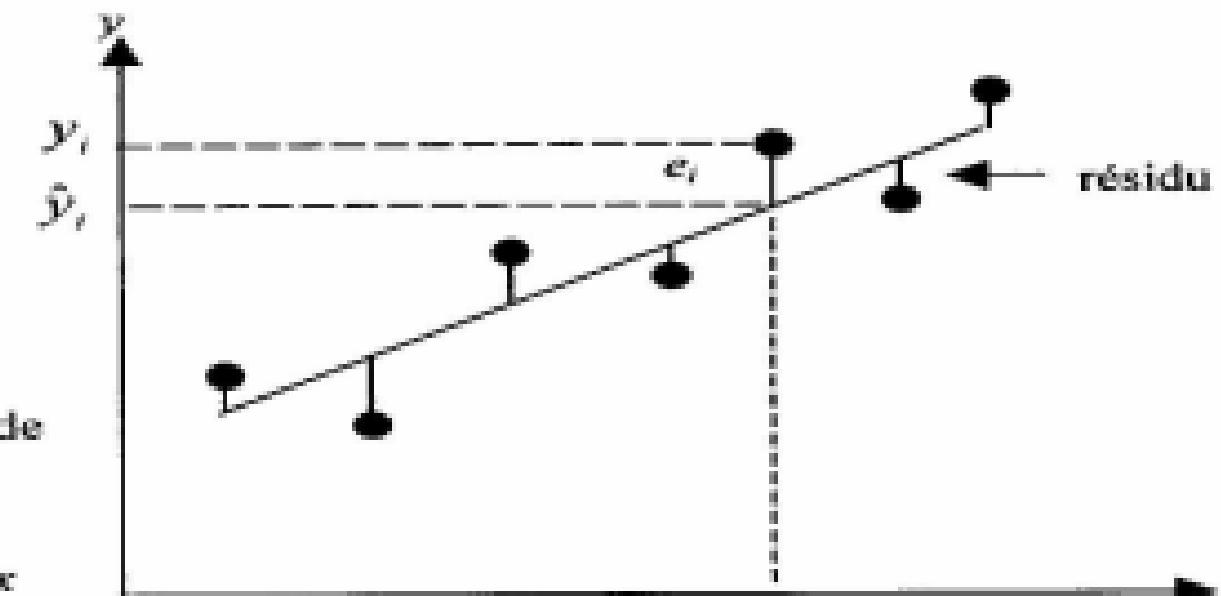
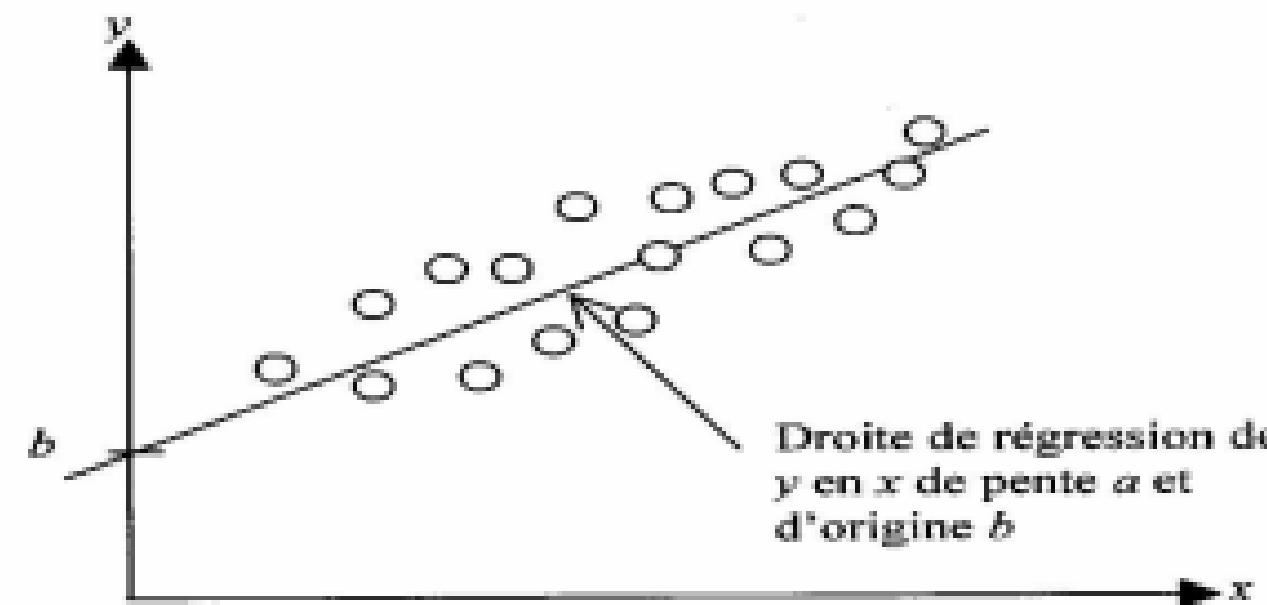


```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1017209 entries, 0 to 1017208
Data columns (total 16 columns):
 #   Column           Non-Null Count  Dtype  
--- 
 0   Store            1017209 non-null   int64  
 1   DayOfWeek        1017209 non-null   int64  
 2   Sales            1017209 non-null   int64  
 3   Open             1017209 non-null   int64  
 4   Promo            1017209 non-null   int64  
 5   StateHoliday     1017209 non-null   int64  
 6   SchoolHoliday    1017209 non-null   int64  
 7   StoreType         1017209 non-null   int64  
 8   Assortment       1017209 non-null   int64  
 9   CompetitionDistance  1017209 non-null   float64 
 10  CompetitionOpenSinceMonth 1017209 non-null   float64 
 11  CompetitionOpenSinceYear 1017209 non-null   float64 
 12  Promo            1017209 non-null   int64  
 13  Year              1017209 non-null   int64  
 14  Month             1017209 non-null   int64  
 15  Day               1017209 non-null   int64  
dtypes: float64(3), int64(13)
memory usage: 131.9 MB
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 41088 entries, 0 to 41087
Data columns (total 16 columns):
 #   Column           Non-Null Count  Dtype  
--- 
 0   Id               41088 non-null   int64  
 1   Store            41088 non-null   int64  
 2   DayOfWeek        41088 non-null   int64  
 3   Open             41088 non-null   float64 
 4   Promo            41088 non-null   int64  
 5   StateHoliday     41088 non-null   int64  
 6   SchoolHoliday    41088 non-null   int64  
 7   StoreType         41088 non-null   int64  
 8   Assortment       41088 non-null   int64  
 9   CompetitionDistance  41088 non-null   float64 
 10  CompetitionOpenSinceMonth 41088 non-null   float64 
 11  CompetitionOpenSinceYear 41088 non-null   float64 
 12  Promo            41088 non-null   int64  
 13  Year              41088 non-null   int64  
 14  Month             41088 non-null   int64  
 15  Day               41088 non-null   int64  
dtypes: float64(4), int64(12)
memory usage: 5.3 MB
```

La modélisation linéaire

$$E = \sum_{i=0}^n \varepsilon_i^2 = \sum_{i=0}^n (y_i - (ax_i + b))^2.$$



Les modèles de pénalisation

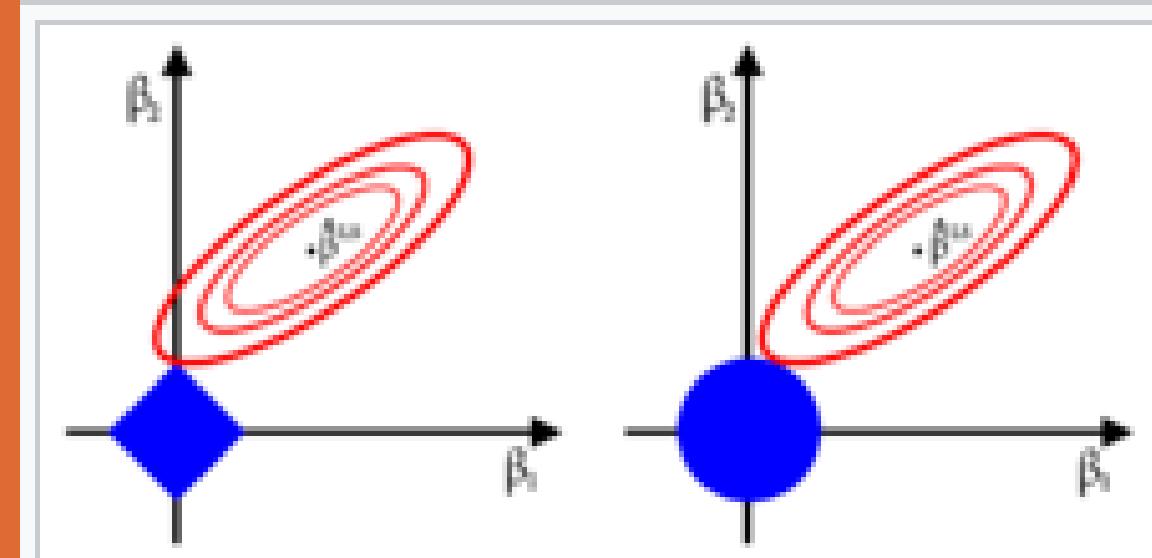
Pénalité de la méthode Lasso

$$\lambda \sum_{i \leq p} |\beta_i|$$

Pénalité de la méthode Ridge

$$\lambda \sum_{i \leq p} \beta_i^2$$

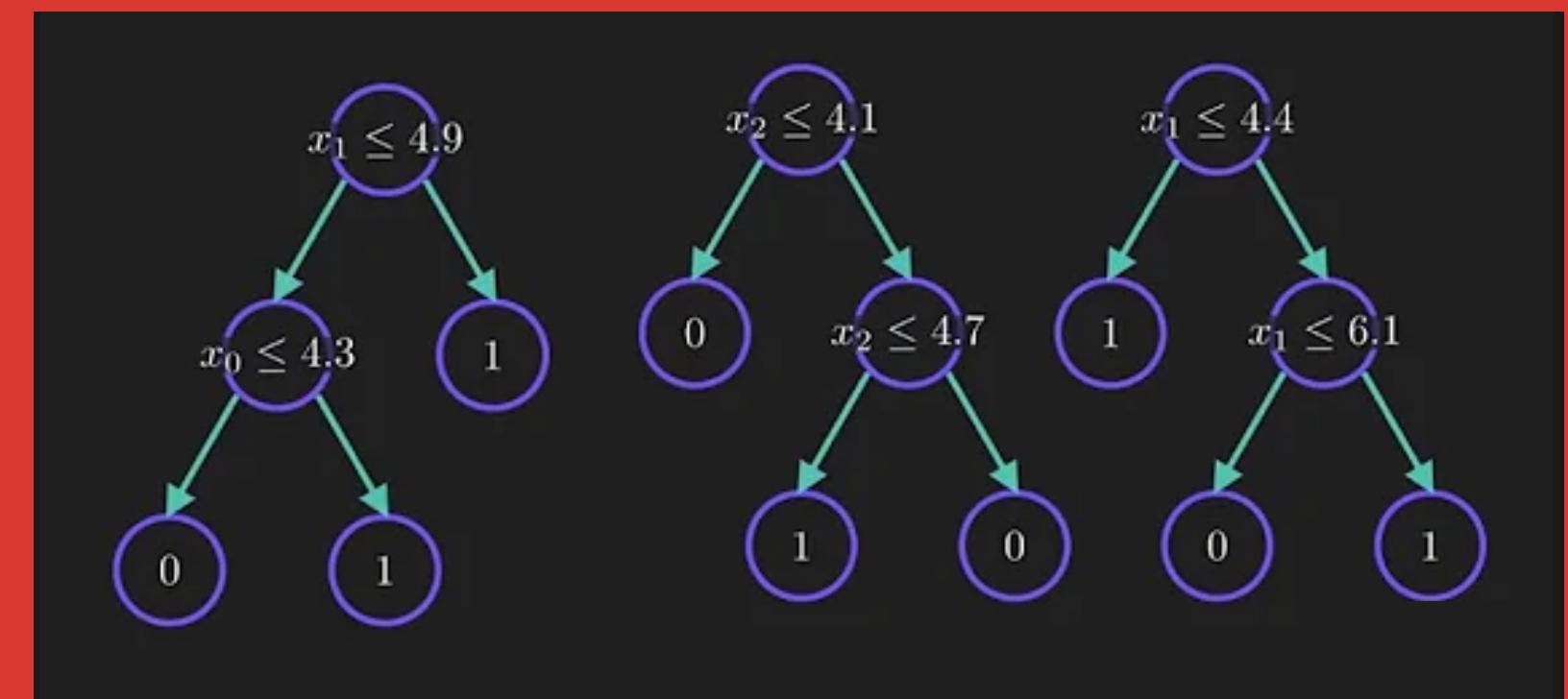
$$Y = X\beta + \epsilon$$



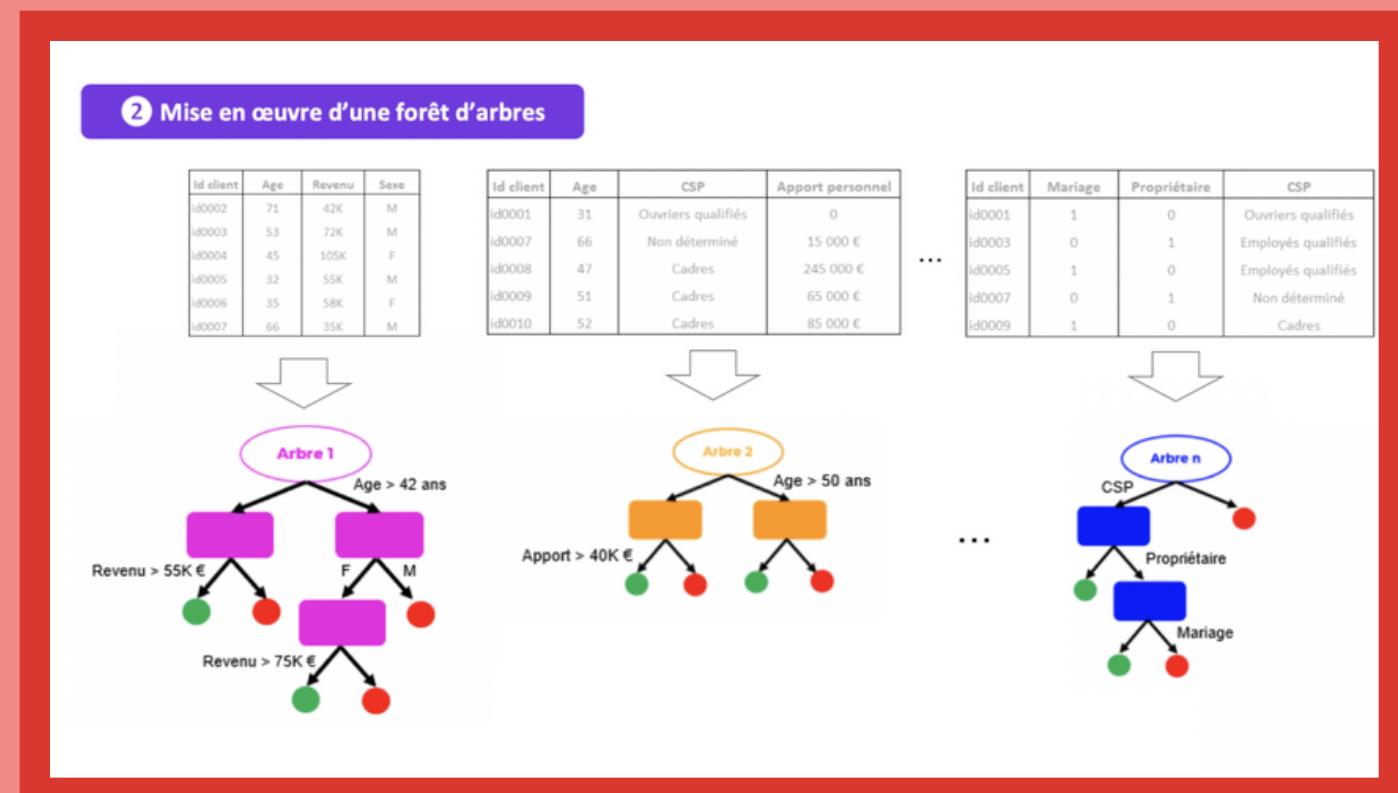
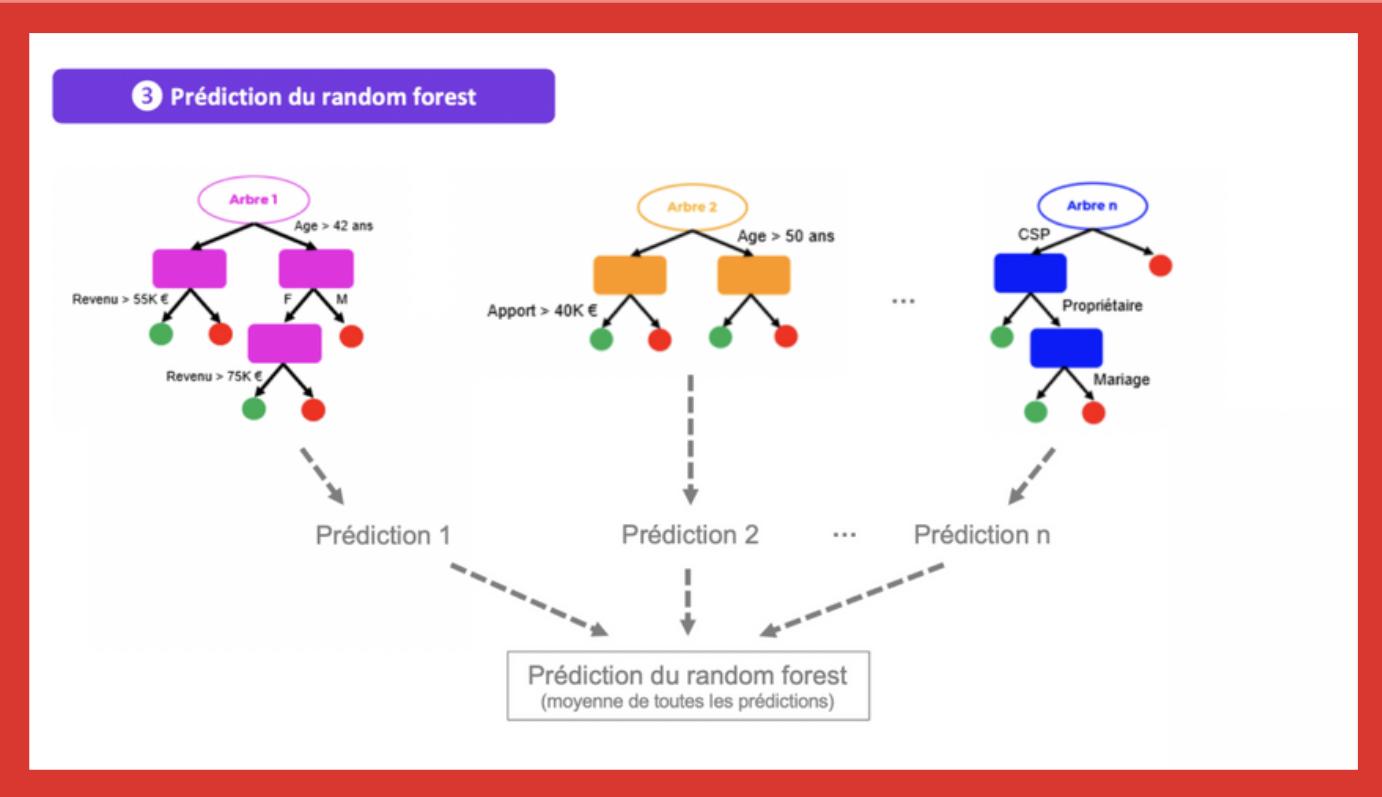
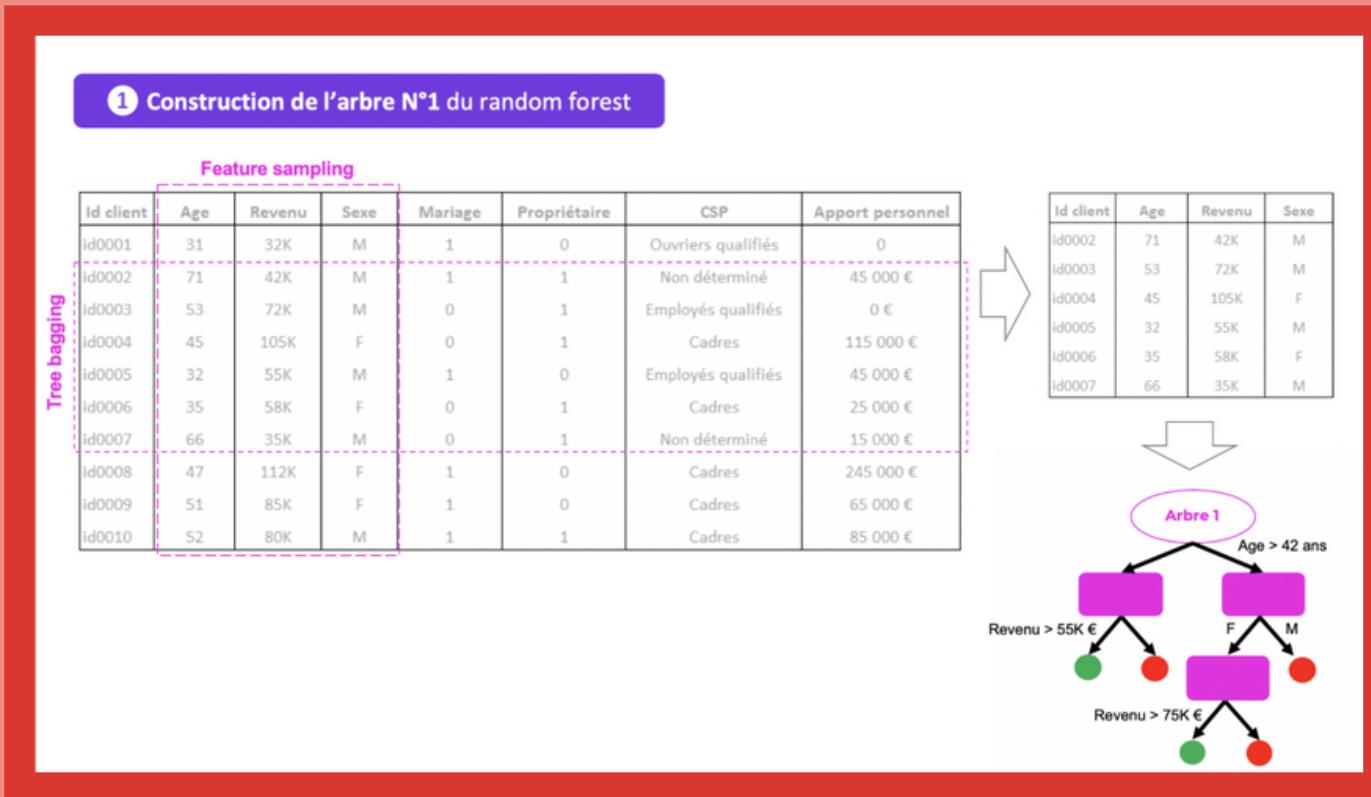
En bleu, zones de contraintes de la pénalité lasso (à gauche) et de la pénalité ridge (à droite). En rouge, les contours de la fonction d'erreur des moindres carrés.

RANDOM FOREST

- Arbres de décision
- Classification
- sklearn Package



RANDOM FOREST



Séries chronologiques

- Modèle MA(q)

$$MA(q) : X_t = \mu + \theta_1 \omega_{t-1} + \theta_2 \omega_{t-2} \dots + \theta_q \omega_{t-q} \text{ avec } \mu = E[X_t], \text{ et } \omega_i \sim \mathcal{N}(0, \sigma^2)$$

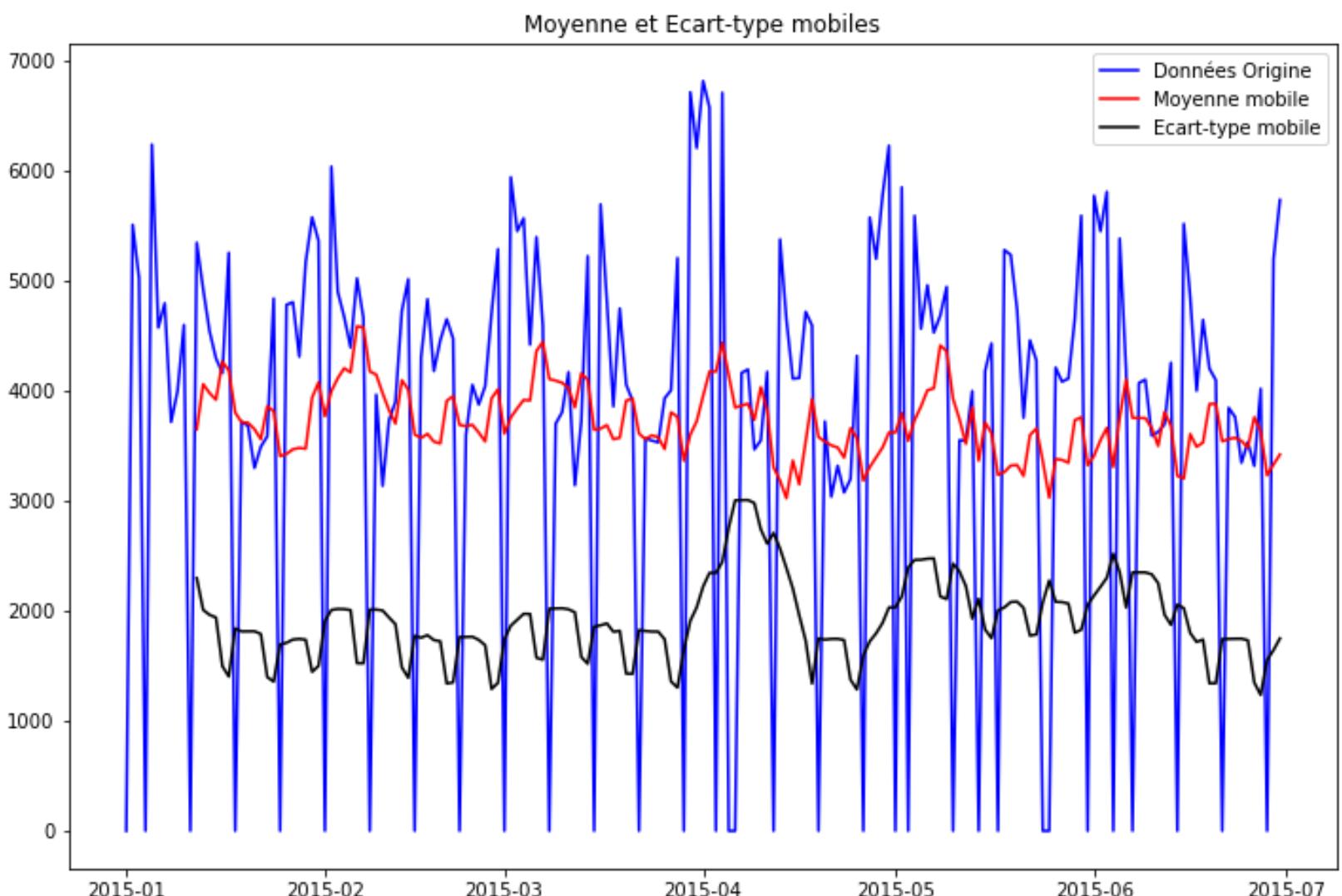
Les ω_i sont les termes d'erreur de bruit (bruit blanc).

- Modèle AR(p)

$$X_t = \sum_{k=1}^p a_k X_{t-k} + \varepsilon_t \quad \varepsilon_t \sim \mathcal{N}(\mu, \sigma^2)$$

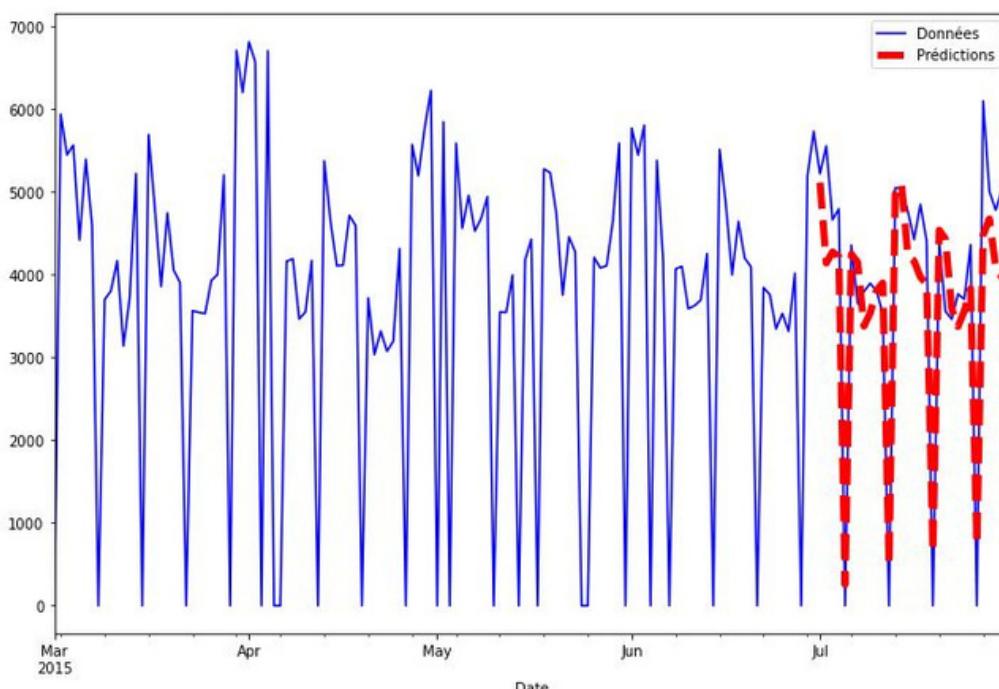
$\varepsilon_1, \dots, \varepsilon_T$ indépendantes

Toutefois, ce type de modèle ne peut être utilisé qu'après s'être assuré que la série étudiée est **stationnaire**!



Store type c:

```
Entrée [26]: 1 model1, rcarré, rmse, rmspe = AR_fonc(data=sales_c, n=50, start_train='2015-03-01', end_train='2015-06-30',  
2 start_test='2015-07-01', end_test='2015-07-31')  
3 print(r'$R^2$ vaut:', rcarré)  
4 print('Le RMSE vaut:', rmse)  
5 print('Le RMSPE vaut:', rmspe)
```



\$R^2\$ vaut: 0.8444927277763012
Le RMSE vaut: 644.3908329753119
Le RMSPE vaut: 0.13354374726823134

Résultats

	Début train	Fin train	Debut validation	Fin Validation	SCORE
Essai 1	'2015-03-01'	'2015-05-31'	'2015-06-01',	'2015-07-31'	0.37365
Essai 2	'2014-03-01'	'2014-07-31'	'2014-08-01'	'2014-09-17'	0.27495
Essai 3	'2014-04-01'	'2014-07-31'	'2014-08-01'	'2014-09-17'	0.24256



Conclusion

Merci