

RAPPORT DE PROJET D'ÉLÈVE INGÉNIEUR DE 2^{ÈME} ANNÉE
FILIÈRE : MODÉLISATION MATHÉMATIQUE ET SCIENCE DES DONNÉES

**Optimisation de la prévision de la consommation
électrique**

Réalisé par : Aymane LEYLI & Mohamed BERRHOU

Tuteur : Dr. Vincent BARRA

Mardi 17 Mars 2020

Novembre 2019 - Février 2020

Campus des Cézeaux. 1 rue de la Chébarde. TSA 60125. 63178 Aubières CEDEX

Remerciements

Nous tenons à exprimer nos sincères remerciements à Monsieur Vincent BARRA, notre tuteur pour avoir accepté de nous encadrer et lui exprimer notre gratitude pour le temps qui nous a accordé afin de nous aider à surmonter toutes les obsacles auxquelles nous avons dû faire face.

Nous tenons aussi à remercier Mme. Violaine ANTOINE ainsi que Mme. Murielle MOUZAT pour leur présence lors de la soutenance du projet.

Résumé

Le but de ce projet est d'étudier plusieurs algorithmes de prédiction pour les séries temporelles (univariées et multivariées) et de les appliquer sur deux tables de données de consommation d'électricité pour pouvoir en tirer les avantages et les limites.

Mots-clés : Apprentissage statistique, Séries temporelles, Prédiction, ARIMA, FbProphet, VAR, VARMA

Abstract

This project aims studying several forecasting algorithms for time series (univariate and multivariate) and testing them on two electricity consumption data frames so as to be able to draw the advantages and the disadvantages of each algorithm.

Keywords : Machine learning, Time series, Forecasting, ARIMA, FbProphet, VAR, VARMA

Table des figures

1	Analytics Vidhya	8
2	train.csv	8
3	test.csv	9
4	Kaggle	9
5	RU_Electricity_Market_PZ_dayahead_price_volume.csv	10
6	consumption_eur	10
7	Anaconda	11
8	Jupyter	11
9	Pandas	11
10	Pandas	11
11	Matplotlib	11
12	Scikit Learn	12
13	Prophet	12
14	Diagramme de Gantt	12
15	Exemple - La consommation d'électricité pour Juillet 2013	13
16	Décomposition de la série de la consommation électrique Juillet 2013	14
17	Exemple - Série temporelle non stationnaire	14
18	Exemple - Série temporelle stationnaire	15
19	Exemple - Fonction d'autocorrélation pour la consommation de Juillet 2013	16
20	Exemple - Fonction d'autocorrélation partielle pour la consommation de Juillet 2013	17
21	Exemple - Détermination du paramètre q à l'aide du graphe de la fonction d'auto-corrélation	21
22	Exemple - Détermination du paramètre q à l'aide du graphe de la fonction d'auto-corrélation	21
23	Comparaison des résultats obtenus par ARIMA avec les données réelles pour le jeu de données Electrolysis.	31
24	Comparaison des résultats obtenus par FbProphet avec les données réelles pour le jeu de données Electrolysis.	32
25	Comparaison des résultats obtenus à l'aide du modèle VAR avec les données réelles pour le jeu de données Electrolysis.	33
26	Comparaison des résultats obtenus à l'aide du modèle VARMA avec les données réelles pour le jeu de données Electrolysis.	33
27	Comparaison des résultats obtenus par FbProphet avec les données réelles pour le jeu de données RU_Electricity.	34
28	Comparaison des résultats obtenus par ARIMA avec les données réelles pour la prédiction à long terme pour le jeu de données RU_Electricity.	35

Table des matières

Remerciements	2
Résumé	3
Abstract	3
Table des illustrations	4
Table des matières	6
Introduction	7
1 Outils et répartition des tâches :	8
1.1 Outils :	8
1.1.1 Jeux de données :	8
1.1.2 Langage et bibliothèques	11
1.2 Répartition des tâches :	12
2 Séries temporelles	13
2.1 Définition et décomposition d'une série temporelle	13
2.2 Notion de stationnarité	14
2.3 Autocorrélation - Autocorrélation partielle	15
2.3.1 Autocorrélation	15
2.3.2 Autocorrélation partielle	16
3 Moyenne mobile intégrée autorégressive - ARIMA	17
3.1 Définition	17
3.1.1 Autorégression (AR) :	18
3.1.2 Moyenne Mobile (MA) :	19
3.1.3 Différenciation :	19
3.2 Choix des paramètres :	20
3.2.1 Le choix de d :	20
3.2.2 Le choix de q :	20
3.2.3 Le choix de p :	21
3.3 Avantages et limites :	22
3.3.1 Avantages :	22
3.3.2 Limites :	22
4 Facebook Prophet :	23
4.1 Définition :	23
4.2 Principe :	23
4.2.1 Le modèle de la tendance :	23
4.2.2 La saisonnalité :	25
4.2.3 Vacances et événements :	26
4.3 Avantages et Limites	26
4.3.1 Avantages	26
4.3.2 Limites	27

5	VAR - VARMA	28
5.1	Vecteur Autoregressif - VAR	28
5.1.1	Définition	28
5.1.2	Causalité au sens de Granger	28
5.1.3	Choix du paramètre p	29
5.1.4	Avantages du modèle VAR	29
5.1.5	Limites du modèle VAR	29
5.2	Vecteur Autoregressif Moyenne Mobile - VARMA	29
5.2.1	Définition	29
5.2.2	Avantages du modèle VARMA	30
5.2.3	Limites du modèle VARMA	30
6	Resultats obtenus	31
6.1	Electrolysia	31
6.2	RU_Electricity	34
	Conclusion	36
	Références	37
	Glossaire	38

Introduction

Depuis quelques années, la consommation d'électricité en France a connu une augmentation au niveau des pointes. Celles-ci augmentent chaque année de 3%. Ainsi, la consommation d'électricité augmente de 0,6% par an. En particulier, cette consommation a été multipliée par 3,2 de 1973 à 2010. La raison principale des pointes de consommation provient de l'utilisation massive du chauffage électrique dans l'habitat. Les nouveaux usages des appareils électriques sont aussi mis en avant (équipements électroménagers, informatique, recharge multiple). S'inscrivant comme un contributeur majoritaire dans la dépense énergétique, le chauffage électrique n'a fait qu'augmenter au fil des années pour passer de 24% en 1990 à 34% en 2013. Lors des pointes de consommation, l'électricité consommée peut doubler par rapport à la consommation moyenne. Dans ces cas, il est nécessaire de produire plus d'électricité très rapidement. C'est alors dans ces moments que des moyens de production très polluants sont utilisés. Par exemple, on peut citer les centrales utilisant du charbon ou du fioul.

Lors des consommations les plus extrêmes, des coupures d'électricité peuvent se produire. Ces pointes de consommation sont d'autant plus difficiles à gérer que la part des énergies renouvelables augmente. Ces sources d'énergies sont intermittentes, par conséquent il est difficile d'ajuster les horaires de production d'électricité. Ces dernières années, la gestion de la consommation d'électricité est cruciale pour garantir l'équilibre entre l'offre et la demande.

Dans le cadre de notre projet de deuxième année en école d'ingénieur en informatique, nous avons essayé d'étudier quelques algorithmes de prévision afin de pouvoir prédire la consommation future le plus précisément possible.

1 Outils et répartition des tâches :

1.1 Outils :

1.1.1 Jeux de données :

Le premier jeu de données dont nous disposons est extrait d'un hackathon d'Analytics Vidya. Le sujet du concours qui est clos actuellement était comme suit :

La compagnie '**Electrolysia**' distribue l'électricité dans la ville. Elle cherche à optimiser sa production électrique en se basant sur les anciennes données de la consommation électrique des habitants d'Electrovanya.

Cette compagnie vous a employé comme un 'Data Scientist' pour fouiller les données de la consom-



FIGURE 1 – Analytics Vidhya

mation électrique ainsi que celles de la température et de la vitesse du vent afin de trouver un modèle qui capte la tendance le plus précisément possible. Il faut garder en esprit qu'il y a plusieurs facteurs qui peuvent impacter la consommation de l'électricité et on ne peut pas les mesurer tous. Electrolysia vous a confié ces données avec une fréquence horaire qui s'étendent sur cinq années. Pour cette compétition les données d'apprentissage sont comprises entre le 1er et le 23 de chaque mois alors que les données de test s'étendent du 24 jusqu'à la fin du mois. Votre tâche sera de prédire la consommation électrique avec une fréquence horaire pour la période de test. Veuillez noter que les données de la consommation ne sont disponibles que dans la base d'apprentissage et qu'il n'est pas possible en aucun cas pouvoir prédire la consommation d'une certaine période en se basant sur la consommation d'une période ultérieure.

Les données d'apprentissage sont sous forme d'une table de 8 colonnes et 26496 lignes. Ci-dessous un exemple des 10 premiers éléments de la table train.csv :

```
In [7]: train.head(10)
```

Out[7]:

	ID	datetime	temperature	var1	pressure	windspeed	var2	electricity_consumption
0	0	2013-07-01 00:00:00	-11.4	-17.1	1003.0	571.910	A	216.0
1	1	2013-07-01 01:00:00	-12.1	-19.3	996.0	575.040	A	210.0
2	2	2013-07-01 02:00:00	-12.9	-20.0	1000.0	578.435	A	225.0
3	3	2013-07-01 03:00:00	-11.4	-17.1	995.0	582.580	A	216.0
4	4	2013-07-01 04:00:00	-11.4	-19.3	1005.0	586.600	A	222.0
5	5	2013-07-01 05:00:00	-10.7	-19.3	1013.0	2.790	A	216.0
6	6	2013-07-01 06:00:00	-13.6	-17.1	1006.0	4.380	A	219.0
7	7	2013-07-01 07:00:00	-11.4	-17.9	997.0	7.635	A	222.0
8	8	2013-07-01 08:00:00	-10.7	-18.6	1012.0	12.055	A	219.0
9	9	2013-07-01 09:00:00	-10.0	-16.4	1002.0	17.975	A	222.0

FIGURE 2 – train.csv

Les données de test sont sous forme d'une table de 7 colonnes et 8568 lignes.

```
In [10]: test.head(10)
```

```
Out[10]:
```

	ID	datetime	temperature	var1	pressure	windspeed	var2
0	552	2013-07-24 00:00:00	-10.0	-16.4	1011.0	263.280	A
1	553	2013-07-24 01:00:00	-10.0	-20.7	1011.0	267.175	A
2	554	2013-07-24 02:00:00	-10.7	-17.1	1003.0	269.555	A
3	555	2013-07-24 03:00:00	-13.6	-20.7	1008.0	273.060	A
4	556	2013-07-24 04:00:00	-10.7	-17.1	1006.0	1.765	A
5	557	2013-07-24 05:00:00	-10.7	-20.0	1009.0	2.915	A
6	558	2013-07-24 06:00:00	-12.1	-20.7	997.0	6.295	A
7	559	2013-07-24 07:00:00	-15.7	-17.9	1003.0	8.675	A
8	560	2013-07-24 08:00:00	-12.9	-17.1	996.0	12.180	A
9	561	2013-07-24 09:00:00	-10.0	-19.3	999.0	2.140	A

FIGURE 3 – test.csv

Pour ce projet, on a pas exploité les colonnes (variables) 'var1' et 'var2' car aucune explication n'a été fournie avec la documentation des données. Donc on a choisi de ne pas travailler sur des données qui n'ont pas de signification physique pour nous.

Le deuxième jeu de données a été extrait depuis la plateforme Kaggle et est intitulé RU_Electricity_Market_PZ_dayahead_price_volume.csv.



FIGURE 4 – Kaggle

Il est composé de 5 colonnes et 45816 lignes. Les colonnes représentent la date, la consommation électrique de la partie européenne de la Russie, la consommation électrique de la Sibérie, ainsi que les prix moyens du mégawatt sur chaque partie.

Les données sont fournies avec une fréquence horaire.
Ci-dessous un exemple des 10 premiers éléments de cette table :

```
In [4]: df= pd.read_csv('RU_Electricity_Market_PZ_dayahead_price_volume.csv')
df.head(10)
```

```
Out[4]:
```

	timestep	consumption_eur	consumption_sib	price_eur	price_sib
0	2006-09-01 00:00	62341.0	17916.0	275.22	0.00
1	2006-09-01 01:00	60625.0	17467.0	0.00	0.00
2	2006-09-01 02:00	59685.0	17161.0	0.00	0.00
3	2006-09-01 03:00	59500.0	17061.0	0.00	0.00
4	2006-09-01 04:00	60617.0	17074.0	0.00	0.00
5	2006-09-01 05:00	62262.0	17273.0	29.96	0.00
6	2006-09-01 06:00	64503.0	17625.0	343.73	0.00
7	2006-09-01 07:00	68316.0	18074.0	440.01	0.00
8	2006-09-01 08:00	71802.0	18698.0	462.64	0.00
9	2006-09-01 09:00	73390.0	19195.0	471.93	0.00

FIGURE 5 – RU_Electricity_Market_PZ_dayahead_price_volume.csv

On s'est contenté uniquement des données de la consommation sur la partie européenne, donc la table est devenue sous cette forme :

```
In [5]: df=df[['timestep','consumption_eur']]
df.head(10)
```

```
Out[5]:
```

	timestep	consumption_eur
0	2006-09-01 00:00	62341.0
1	2006-09-01 01:00	60625.0
2	2006-09-01 02:00	59685.0
3	2006-09-01 03:00	59500.0
4	2006-09-01 04:00	60617.0
5	2006-09-01 05:00	62262.0
6	2006-09-01 06:00	64503.0
7	2006-09-01 07:00	68316.0
8	2006-09-01 08:00	71802.0
9	2006-09-01 09:00	73390.0

FIGURE 6 – consumption_eur

1.1.2 Langage et bibliothèques

Principalement, le langage de programmation utilisé le long du projet est **Python** sous **Anaconda** qui est une distribution libre et open source avec **Jupyter** comme éditeur de code.



FIGURE 7 – Anaconda



FIGURE 8 – Jupyter

Les bibliothèques Python utilisées sont :

— Pandas : Avec ses structures de données, elle permet des opérations de manipulation et d'exploration de tableaux alphanumériques et de séries temporelles.



FIGURE 9 – Pandas

— Numpy : Elle permet d'effectuer des calculs numériques.



FIGURE 10 – Pandas

— Matplotlib : Elle est destinée à visualiser les données à l'aide des graphiques.



FIGURE 11 – Matplotlib

— Scikit-Learn : Utilisée ici principalement pour la partie prétraitement et test de précision.



FIGURE 12 – Scikit Learn

— Statsmodels : Elle permet d’explorer les données, les transformer en série temporelle pour pouvoir estimer des modèles statistiques ainsi que d’appliquer des tests statistiques.

— Prophet : Bibliothèque open source conçue par Facebook pour la prédiction des séries temporelles



FIGURE 13 – Prophet

1.2 Répartition des tâches :

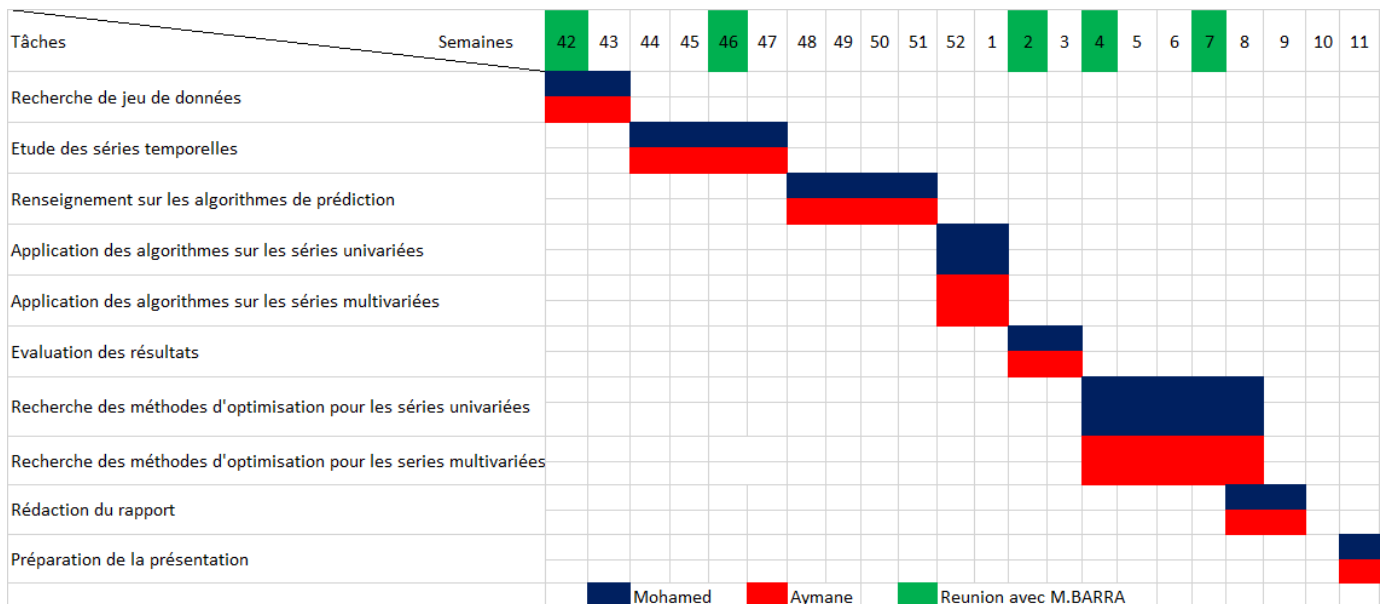


FIGURE 14 – Diagramme de Gantt

2 Séries temporelles

2.1 Définition et décomposition d'une série temporelle

Une série temporelle (ou encore une série chronologique) est une suite (Y_1, \dots, Y_n) de données indexées par le temps. L'indice temps peut être selon les cas la minute, l'heure, le jour, l'année etc... Le nombre n est appelé la longueur de la série. Il est la plupart du temps bien utile de représenter la série temporelle sur un graphe construit de la manière suivante : en abscisse le temps, en ordonnée la valeur de l'observation à chaque instant.

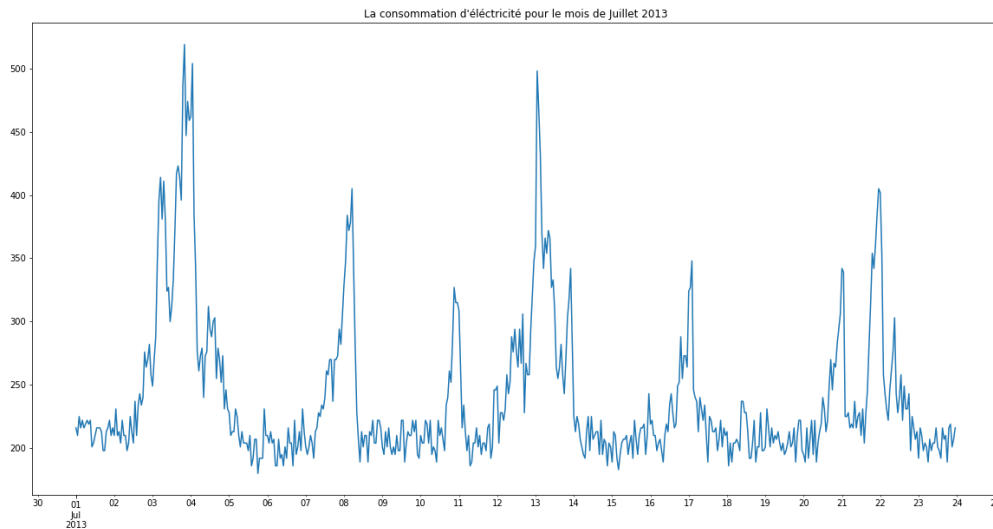


FIGURE 15 – Exemple - La consommation d'électricité pour Juillet 2013

Une série temporelle est généralement constituée de plusieurs éléments :

- a)– Tendence : représente l'évolution à long terme de la série (échelle interannuelle). Exemples : croissance économique, évolution climatologique à long terme.
- b)– Saisonnalité : évolution se répétant régulièrement tous les ans. Exemples : – En météorologie, température plus faible en hiver qu'en été. – En économie, saisonnalité induite par les périodes de vacances, les périodes de fêtes, le climat...
- c)– Composante stationnaire (ou résiduelle) : ce qui reste lorsque l'on a enlevé les autres composantes. Décrit l'évolution à court terme de la série (échelle journalière).

Le modèle le plus usuel consiste à supposer que la série initiale s'écrit sous la forme (modèle additif)

$$Y_t = T_t + S_t + R_t \quad (2.1)$$

Avec :

- Y_t : La série temporelle
- T_t : La tendance de la série temporelle
- S_t : La saisonnalité série temporelle
- R_t : Le résidu

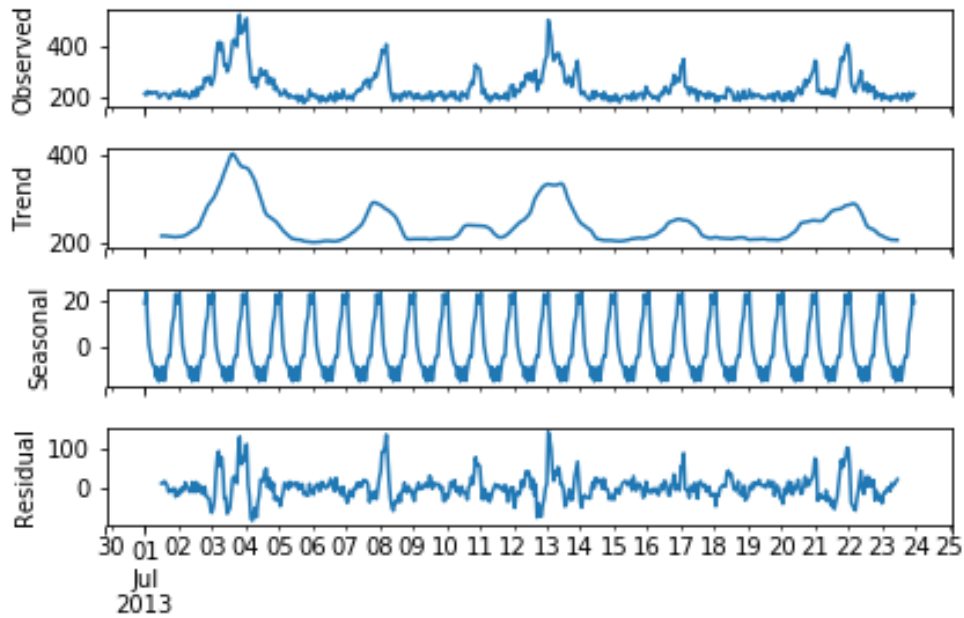


FIGURE 16 – Décomposition de la série de la consommation électrique Juillet 2013

2.2 Notion de stationnarité

Définition

La stationnarité signifie que les propriétés statistiques d'un processus générant une série chronologique ne changent pas avec le temps. Cela ne signifie pas que la série ne change pas avec le temps, mais simplement que la façon dont elle évolue ne change pas elle-même avec le temps. L'équivalent algébrique est donc une fonction linéaire. La valeur d'une fonction linéaire change à mesure que x croît, mais la façon dont elle évolue reste constante (sa pente est constante) .

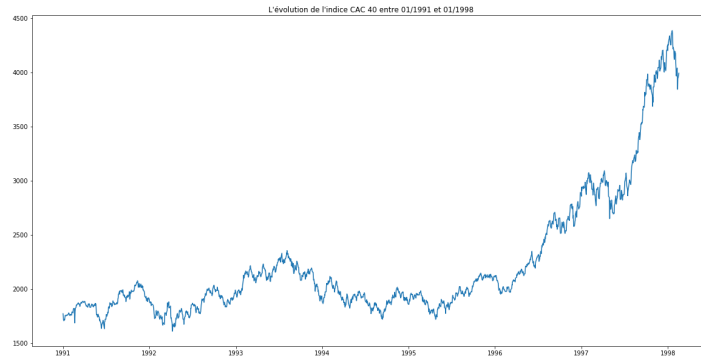


FIGURE 17 – Exemple - Série temporelle non stationnaire

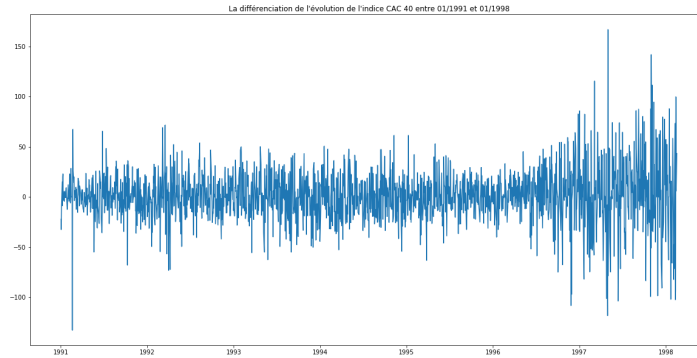


FIGURE 18 – Exemple - Série temporelle stationnaire

Test de Stationnarité

Le test Dickey Fuller augmenté (ADF) est un test de racine unitaire pour la stationnarité . Les hypothèses du test :

- L' hypothèse nulle pour ce test est qu'il existe une racine unitaire.
- L' hypothèse alternative diffère légèrement selon l'équation utilisée. L'alternative de base est que la série chronologique est stationnaire (ou tendance-stationnaire). Soit (Y_t) une série temporelle. Les trois modèles de régression de base sont :

- Sans constante, sans tendance : $\Delta Y_t = \alpha Y_{t-1} + \epsilon_t$
- Avec constante, sans tendance : $\Delta Y_t = C + \alpha Y_{t-1} + \epsilon_t$
- Avec constante et avec tendance : $\Delta Y_t = C + \alpha Y_{t-1} + \lambda_t + \epsilon_t$

Avec :

- α : le coefficient du modèle.
- C : constante du modèle.
- ϵ_t : le bruit blanc de la série.
- λ : la tendance de la série.

Le Dickey Fuller augmenté ajoute des différences décalées à ces modèles :

- Sans constante, sans tendance : $\Delta Y_t = \alpha Y_{t-1} + \sum_{s=1}^m a_s \Delta Y_{t-s} + \epsilon_t$
- Avec constante, sans tendance : $\Delta Y_t = C + \alpha Y_{t-1} + \sum_{s=1}^m a_s \Delta Y_{t-s} + \epsilon_t$
- Avec constante et avec tendance : $\Delta Y_t = C + \alpha Y_{t-1} + \sum_{s=1}^m a_s \Delta Y_{t-s} + \lambda_t + \epsilon_t$

Avec :

- m : le nombre des différences décalées.
- a_s : les coefficients associés à chaque différenciation.

La longueur de décalage doit être choisie de sorte que les résidus ne soient pas corrélés en série. Plusieurs options pour choisir les décalages : minimisation du critère d'information d'Akaike (AIC) ou le critère d'information bayésien (BIC) , ou la suppression des décalages jusqu'à ce que le dernier décalage soit statistiquement significatif .

2.3 Autocorrélation - Autocorrélation partielle

2.3.1 Autocorrélation

La corrélation statistique résume la force de la relation entre deux variables. Nous pouvons calculer la corrélation pour les observations de séries chronologiques avec les pas de temps précédents, appelés décalages. Étant donné que la corrélation des observations de séries chronologiques est cal-

culée avec des valeurs de la même série à des moments antérieurs, cela s'appelle une corrélation en série, ou une autocorrélation.

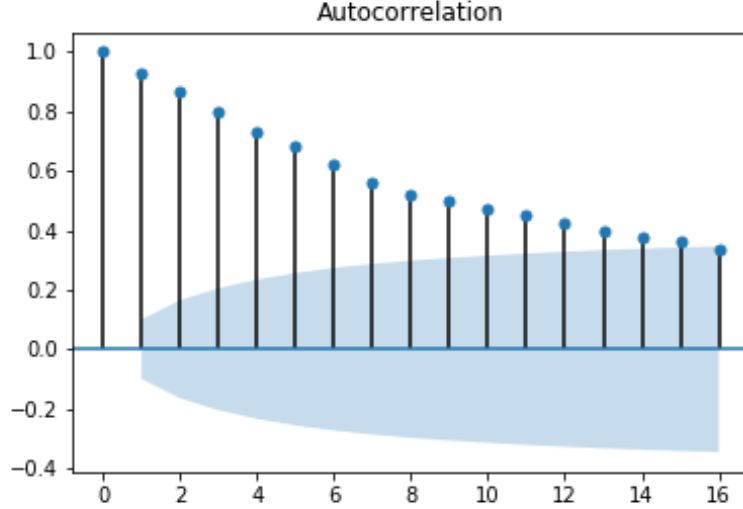


FIGURE 19 – Exemple - Fonction d'autocorrélation pour la consommation de Juillet 2013

Pour définir mathématiquement la fonction d'autocorrélation, nous devons citer premièrement la fonction de covariance.

Soit Y_t une série temporelle. Sa fonction d'autocovariance $\gamma Y()$ est la fonction définie sur \mathbb{Z} par :

$$h \rightarrow \gamma Y(h) = \text{cov}(Y_h, Y_0) = \text{cov}(Y_{t+h}, Y_t), \forall t \in \mathbb{Z} \quad (2.2)$$

La fonction d'autocorrélation $\rho Y()$ est définie par :

$$\rho Y : \mathbb{Z} \rightarrow [-1, 1]$$

$$h \rightarrow \rho Y(h) = \gamma Y(h) \gamma Y(0) = \text{corr}(Y_{t+h}, Y_t) = \frac{\text{cov}(Y_{t+h}, Y_t)}{\sqrt{\text{var}(Y_t) \text{var}(Y_{t+h})}}, \forall t \in \mathbb{Z} \quad (2.3)$$

Ces deux fonctions mesurent le degré de dépendance entre les valeurs d'une série à des instants différents ou bien de la série avec elle-même avec un décalage de temps (Lag)

2.3.2 Autocorrélation partielle

Une autocorrélation partielle est la relation entre une observation dans une série chronologique avec des observations à des pas de temps antérieurs, en supprimant les relations des observations intermédiaires.

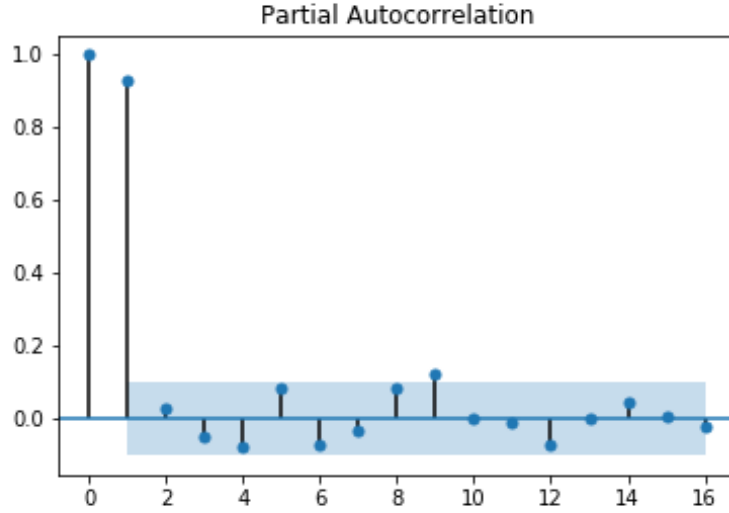


FIGURE 20 – Exemple - Fonction d'autocorrélation partielle pour la consommation de Juillet 2013

Mathématiquement parlant, l'autocorrélation partielle d'ordre k désigne la corrélation entre Y_t et Y_{t-k} obtenue lorsque l'influence des variables Y_{t-k-i} , avec $0 < i < k$, a été retirée.

L'autocorrélation partielle d'ordre k d'une série $(Y_t)_{t \in \mathbb{Z}}$, de moyenne $E(Y_t) = m$, notée $p(k)$ ou p_k , est définie par le dernier coefficient de la projection linéaire de Y_{t+1} sur ces k plus récentes valeurs.

$$\forall k \in \mathbb{Z} \quad Y_{t+1} - m = c_1(Y_t - m) + c_2(Y_{t-1} - m) + \dots + c_{k-1}(Y_{t-k+1} - m) + p_k(Y_{t-k} - m) \quad (2.4)$$

Ou de façon équivalente par :

$$\begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ p_k \end{pmatrix} = \begin{pmatrix} \gamma_0 & \gamma_1 & \cdot & \gamma_{k-1} \\ \gamma_1 & \gamma_0 & \cdot & \gamma_{k-2} \\ \cdot & \cdot & \cdot & \cdot \\ \gamma_{k-1} & \gamma_{k-2} & \cdot & \gamma_0 \end{pmatrix}^{-1} \begin{pmatrix} \gamma_1 \\ \gamma_2 \\ \vdots \\ \gamma_k \end{pmatrix} = \begin{pmatrix} 1 & \rho_1 & \cdot & \rho_{k-1} \\ \rho_1 & 1 & \cdot & \rho_{k-2} \\ \cdot & \cdot & \cdot & \cdot \\ \rho_{k-1} & \rho_{k-2} & \cdot & 1 \end{pmatrix}^{-1} \begin{pmatrix} \rho_1 \\ \rho_2 \\ \vdots \\ \rho_k \end{pmatrix}$$

3 Moyenne mobile intégrée autorégressive - ARIMA

3.1 Définition

Un modèle ARIMA est étiqueté comme modèle ARIMA (p, d, q), dans lequel :

p : est le nombre de termes autorégressifs.

d : est le nombre de différences.

q : est le nombre de moyennes mobiles.

3.1.1 Autorégression (AR) :

Les processus autorégressifs supposent que chaque point peut être prédit par la somme pondérée d'un ensemble de points précédents, plus un terme aléatoire d'erreur, par exemple un processus autorégressif d'ordre p peut être donné par l'équation :

$$X_t = \mu + \sum_{i=1}^p \alpha_i X_{t-i} + \epsilon_t \quad (3.1)$$

Avec :

X_t : La valeur de la composante autorégressive de la série chronologique à l'instant t .

μ : Constante du processus.

α_i : Coefficient d'auto-régression associé à l'instant $t-i$ pour $i \in \{1, \dots, p\}$.

ϵ_t : La composante aléatoire associée à l'observation.

L'équation (3.1) peut s'écrire à l'aide de l'opérateur de retard L par :

$$(1 - \sum_{i=1}^p \alpha_i \mathcal{L}^i) X_t = \mu + \epsilon_t \quad (3.2)$$

Avec : $\mathcal{L}X_t = X_{t-1}$.

Comment assurer la stabilité du processus ?

Théorème :

Un processus AR(p) est stationnaire si le module des solutions (les racines) de son équation caractéristique est à chaque fois strictement supérieure à 1 en valeur absolue.

Exemple pour $p=1$:

L'équation pour un processus AR(1) peut s'écrire :

$$(1 - \alpha_1 L) X_t = \mu + \epsilon_t \quad (3.3)$$

La résolution de l'équation caractéristique donne :

$$1 - \alpha_1 x = 0 \iff x = \frac{1}{\alpha_1} \quad (3.4)$$

Alors :

$$|x| > 1 \Rightarrow |\alpha_1| < 1. \quad (3.5)$$

Exemple pour $p=2$:

L'équation pour un processus AR(2) peut s'écrire :

$$(1 - \alpha_1 L - \alpha_2 L^2) X_t = \mu + \epsilon_t \quad (3.6)$$

La résolution de l'équation caractéristique $1 - \alpha_1 x - \alpha_2 x^2$, amène aux conditions suivantes :

$$\alpha_1 - \alpha_2 < 1.$$

$$\alpha_1 + \alpha_2 < 1.$$

$$|\alpha_2| < 1.$$

3.1.2 Moyenne Mobile (MA) :

Les modèles à moyenne mobile suggèrent que la série présente des fluctuations autour d'une valeur moyenne. On considère alors que la meilleure estimation est représentée par la moyenne pondérée d'un certain nombre de valeurs antérieures. Ceci revient en fait à considérer que l'estimation est égale à la moyenne vraie, auquel on ajoute une somme pondérée des erreurs ayant entaché les valeurs précédentes. Un modèle à moyenne mobile d'ordre q peut être présenté par l'équation :

$$Z_t = \mu + \sum_{i=1}^q \theta_i \epsilon_{t-i} + \epsilon_t \quad (3.7)$$

Avec :

Z_t : La composante du processus à l'instant t .

μ : La moyenne de la série.

θ_i : Coefficient de moyenne mobile du modèle pour $i \in \{1, \dots, q\}$.

ϵ_{t-i} : erreurs aléatoires passées en instant $t-i$ pour $i \in \{1, \dots, q\}$.

ϵ_t : composante d'erreur aléatoire.

Ainsi, un modèle à moyenne mobile est conceptuellement une régression linéaire de la valeur actuelle de la série par rapport aux termes d'erreur de bruit blanc actuels et précédents (observés) ou aux chocs aléatoires. Les chocs aléatoires à chaque point sont supposés être mutuellement indépendants et provenir de la même distribution, généralement une distribution normale.

3.1.3 Différenciation :

L'estimation des modèles ARIMA suppose que l'on travaille sur une série stationnaire. Ceci signifie que la moyenne de la série est constante dans le temps, ainsi que la variance. La meilleure méthode pour éliminer toute tendance est de différencier, c'est-à-dire de remplacer la série originale par la série des différences adjacentes. Une série temporelle qui a besoin d'être différenciée pour atteindre la stationnarité est considérée comme une version intégrée d'une série stationnaire (d'où le terme Integrated).

Une différenciation d'ordre 1 suppose que la différence entre deux valeurs successives de y est constante :

$$y_t - y_{t-1} = \mu + \epsilon_t \quad (3.8)$$

Avec :

y_t : La valeur de la série chronologique à l'instant t .

μ : La constante du modèle.

ϵ_t : La composante aléatoire associée à l'observation.

Un tel modèle est un ARIMA(0,1,0). Il peut être représenté comme un accroissement linéaire en fonction du temps.

Les modèles d'ordre 2 travaillent non plus sur les différences brutes, mais sur les différences de différence. La seconde différence de y au moment t est égale à $(y_t - y_{t-1}) - (y_{t-1} - y_{t-2})$. Un modèle ARIMA(0,2,0) obéira à l'équation de prédiction suivante :

$$y_t - 2y_{t-1} + y_{t-2} = \mu + \epsilon_t \quad (3.9)$$

Exemples :

Application du modèle d'autorégression sur une série différenciée ARIMA(1,1,0) :

$$y_t - y_{t-1} = \mu + \alpha(y_{t-1} - y_{t-2}) + \epsilon_t \quad (3.10)$$

Application du modèle de moyenne mobile sur une série différenciée ARIMA(0,1,1) :

$$y_t - y_{t-1} = \mu + \theta\epsilon_{t-1} + \epsilon_t \quad (3.11)$$

3.2 Choix des paramètres :

L'enjeu auquel font face la plupart des data scientists lors de l'utilisation d'ARIMA est de trouver les meilleurs paramètres (p, d, q) qui sont convenables avec la série chronologique. Si ce n'est pas le cas, le modèle ajusté donnera des résultats indésirables.

3.2.1 Le choix de d :

Le choix de paramètre (d) s'appuie sur l'application du test de Dicky-Fuller expliqué ultérieurement pour lequel nous avons créé une fonction python qui retourne la valeur de p-value qui indique si la série est stationnaire (expliqué dans le chapitre précédent). Nous appliquons la fonction sur notre série chronologique, si $p\text{-value} > 0.05$ alors nous différencions (à l'ordre 1) la série et on applique à nouveau le test de stationnarité, si elle n'est toujours pas stationnaire nous différencions la série jusqu'à trouver l'ordre de différenciation qui donne une valeur de p-value inférieur à 0.05 . En pratique, à partir d'ordre 2 de différenciation la série est toujours stationnaire.

3.2.2 Le choix de q :

Le choix de paramètre q s'appuie sur l'autocorrélation de la série. Nous trouvons les caractéristiques optimales ou l'ordre du processus MA en utilisant le graphique ACF, car étant un processus MA il n'a pas de composantes saisonnières et de tendance, nous obtenons donc uniquement la relation résiduelle avec les décalages des séries chronologiques dans le graphique ACF.

Dans la figure 21, on remarque que les valeurs de la fonction d'autocorrélation ne rentrent dans la zone de confiance (zone en bleu-ciel) qu'à partir du 16 ème décalage. Donc, théoriquement la valeur adéquate pour q est 16. En pratique, cette valeur peut ne pas être la meilleure mais elle donne une information sur la valeur maximale que pourrait avoir le paramètre.

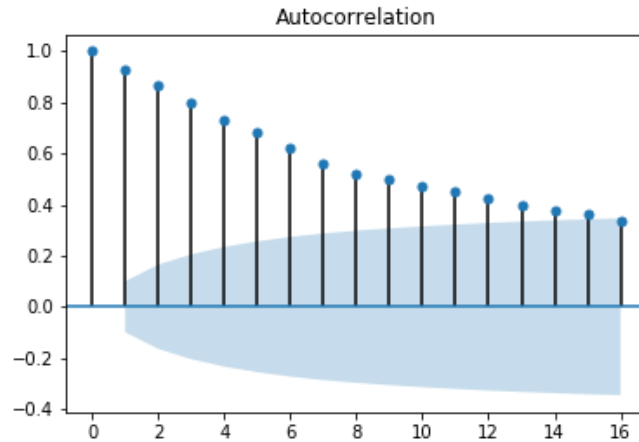


FIGURE 21 – Exemple - Détermination du paramètre q à l'aide du graphe de la fonction d'autocorrélation

3.2.3 Le choix de p :

Nous trouvons les caractéristiques optimales ou l'ordre du processus AR en utilisant le tracé PACF, car il supprime les variations expliquées par les décalages précédents afin que nous n'obtenions que les caractéristiques pertinentes. Dans la figure 22, on remarque que les valeurs de la fonction d'autocorrélation partielle rentrent dans la zone de confiance (zone en bleu-ciel) à partir du 1^{er} décalage. Donc, théoriquement la valeur adéquate pour p est 1. En pratique, cette valeur n'est qu'une estimation de la valeur maximale du paramètre.

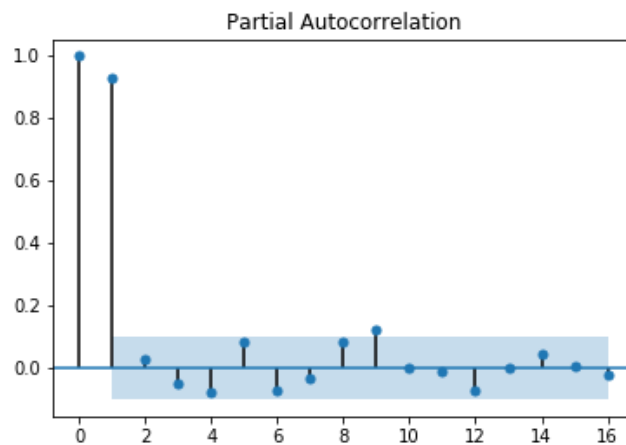


FIGURE 22 – Exemple - Détermination du paramètre q à l'aide du graphe de la fonction d'autocorrélation

Ces 2 valeurs maximales (pour p et q) peuvent être utiles lors de l'utilisation d'une grille de recherche pour déterminer avec précision les meilleurs paramètres pour ajuster le modèle.

3.3 Avantages et limites :

3.3.1 Avantages :

1) Les modèles ARIMA se sont avérés relativement robustes pour la prediction à court terme. Dans notre cas, ARIMA a réussi à indiquer les pointes de la consommation même si il n'a pas donné leurs valeurs avec une grande précision.

2) Ce modèle ne nécessite que les données préalables d'une série chronologique pour générer une prévision, dans notre cas quelques heures pour donner une bonne prévision pour les 6h suivantes.

3.3.2 Limites :

1) ARIMA donne des mauvaises predictions à long terme, car il repose sur les valeurs précédentes (par exemple 10 dernières valeurs précédentes) pour estimer la valeur d'une observation. Alors, si nous voulons prédire la prédiction pour 2 mois il faut s'assurer que le résultat pour le premier mois est très proche de la réalité (ce qui n'est pas toujours le cas).

2) Le modèle ARIMA ne peut être appliqué que sur les series univariées, ici nous ne pouvons pas traiter l'effet de la température sur la consommation d'électricité dans notre jeu de données.

4 Facebook Prophet :

4.1 Définition :

Prophet est un algorithme de prévision open-source de Facebook. Il a été développé pour les problèmes typiques de Facebook, tels que la prédiction de l'activité des utilisateurs dans différentes parties de leur application ou la priorisation du développement de fonctionnalités.

4.2 Principe :

Facebook prophet s'appuie sur un modèle de décomposition additive pour les séries chronologiques (Harvey Peters 1990) avec 3 composantes principales : la tendance, la saisonnalité et les vacances. En combinant ces 3 composantes, nous obtenons l'équation suivante :

$$y(t) = g(t) + s(t) + h(t) + \epsilon_t \quad (4.1)$$

Avec :

y : représente les valeurs de la série chronologique.

g : la fonction de tendance qui modélise les changements non-périodique de la série.

s : la fonction de saisonnalité qui modélise les changements périodiques de la série. (changements : hebdomadaires, mensuelles ...).

ϵ_t : la fonction d'erreur représente tout changement idiosyncratique qui n'est pas pris en compte par le modèle.

4.2.1 Le modèle de la tendance :

Le modèle de Facebook Prophet utilise 2 modèles pour la modélisation de la tendance : modèle de croissance saturante et un modèle linéaire par morceaux.

a) modèle de croissance saturante :

Pour les séries chronologiques qui modélisent des phénomènes qui connaissent une croissance - par exemple : nombre d'utilisateurs de Facebook dans une zone géographique - où il y a une croissance non linéaire qui sature à une capacité de charge - dans notre exemple, le nombre des habitants de cette zone qui ont accès à l'internet - la modélisation de la tendance dans ce cas doit prendre en compte la façon dans laquelle le phénomène s'évalue et la façon dans laquelle elle continue à s'évaluer. Elle peut être modélisée par une fonction logistique de la forme :

$$g(t) = \frac{C}{1 + \exp -k(t - m)} \quad (4.2)$$

Avec :

C : la capacité de charge

K : le taux de croissance.

m : le paramètre de décalage.

Dans (4.2), il y a 2 aspects qui ne sont pas pris en compte. Premièrement, la capacité de charge n'est pas constante. Par exemple si nous voulons estimer le nombre d'utilisateurs de Facebook dans une zone géographique, nous savons que le nombre maximal est bien le nombre de personnes qui ont accès à l'internet dans cette zone, mais ce nombre maximal n'est pas constant. Pour cela nous remplaçons la capacité fixe par une capacité variable $C(t)$.

Deuxièmement, le taux de croissance n'est pas constant. Si nous prenons l'exemple précédent, le nombre de personnes qui ont accès à l'internet est lié à plusieurs facteurs (par exemple le développement de la technologie dans la zone...), donc le modèle doit être capable d'incorporer un taux variable afin d'ajuster les données historiques. Pour cela, on définit un ensemble qui contient les points où le taux de croissance peut changer. Supposons qu'on a S points de changements accordés aux instants s_j , $j \in \{1, \dots, S\}$. On définit ainsi le vecteur d'ajustement de taux $\delta \in \mathbb{R}^s$ où δ_j représente le changement de taux qui se produit au temps s_j . Alors le taux de changements a un instant t est le taux de croissance k plus tous les ajustements jusqu'à ce point :

$$k(t) = k + \sum_{j:t < s_j} \delta_j \quad (4.3)$$

Cela peut être représenté par un produit scalaire en définissant le vecteur α tel que $\alpha(t) \in \{0, 1\}^s$ avec :

$$\alpha_j(t) = \begin{cases} 1, & t \geq s_j \\ 0, & \text{sinon} \end{cases} \quad (4.4)$$

Le taux de changement à l'instant t est donné par : $k + \alpha(t)^\top \delta$. Lorsque le taux k est ajusté, le paramètre de décalage m doit également l'être aussi afin d'atteindre les extrémités des segments des instants. L'ajustement de m est donné par la formule :

$$\gamma_j = (s_j - m - \sum_{l < j} \gamma_l) \left(1 - \frac{k + \sum_{l < j} \delta_l}{k + \sum_{l \leq j} \delta_l}\right) \quad (4.5)$$

Le modèle de croissance logistique par morceaux est alors :

$$g(t) = \frac{C(t)}{1 + \exp(-(k + \alpha(t)^\top \delta)(t - (m + \alpha(t)^\top \gamma)))} \quad (4.6)$$

Le paramètre le plus important dans ce modèle est $C(t)$ la capacité de charge du système car elle empêche le modèle de diverger.

b)Modèle linéaire par morceaux :

Pour les séries chronologiques qui représentent des phénomènes qui ne suivent pas une croissance saturée, un taux constant par morceaux de croissance peut être utile pour modéliser la tendance de la série chronologique. Ce modèle est donné par l'équation :

$$g(t) = (k + \alpha(t)^\top \delta)(m + \alpha(t)^\top \gamma) \quad (4.7)$$

c) Incertitude des prévisions de tendance :

La plus grande source d'incertitude dans les prévisions est le potentiel de changements de tendance futurs. Il est impossible de le savoir avec certitude, le prophète fait donc la chose la plus raisonnable possible, il suppose que l'avenir verra des changements de tendance similaires à ceux du passé. En particulier, il suppose que la fréquence et l'amplitude moyennes des changements de tendance dans le futur seront les mêmes que celles qu'il a observées dans le passé. Il projette ces changements de tendance dans le futur, calcule leur distribution et il obtient des intervalles d'incertitude.

Par exemple, supposons qu'on a S points de changements sur un intervalle de temps $[0, T]$, et chaque point de changement a un taux $\delta_j \sim \text{Laplace}(0, \tau)$. Nous simulons les changements de taux

futurs qui imitent ceux du passé en remplaçant τ par une variance déduite des données, en utilisant l'estimation du maximum de vraisemblance du paramètre d'échelle de taux :

$$\rho = \frac{1}{S} \sum_{j=1}^S |\delta_j| \quad (4.8)$$

Les futurs points de changement sont échantillonnés au hasard de manière à ce que la fréquence moyenne des points de changement corresponde à celle du passé :

Ainsi, l'incertitude dans la tendance prévue est mesurée en supposant que l'avenir verra la même fréquence et l'ampleur moyennes des changements de taux que celles observées dans l'histoire. Une fois que τ a été déduit des données, ce modèle génératif est déployé pour simuler les tendances futures possibles et utiliser les tendances simulées pour calculer les intervalles d'incertitude.

L'hypothèse de Prophète selon laquelle la tendance continuera à changer avec la même fréquence et la même ampleur que dans l'histoire est assez forte, alors il ne faut pas compter sur les intervalles d'incertitude ayant une couverture exacte. Au fur et à mesure que τ augmente, le modèle a plus de souplesse pour ajuster l'historique et donc l'erreur d'apprentissage diminue. Même ainsi, lorsqu'elle est projetée vers l'avant, cette flexibilité est susceptible de produire de larges intervalles. Les intervalles d'incertitude sont cependant une indication utile du niveau d'incertitude, et en particulier un indicateur de sur-ajustement .

4.2.2 La saisonnalité :

La composante saisonnière $s(t)$ offre une adaptabilité au modèle en permettant des changements périodiques basés sur la saisonnalité sous-quotidienne, quotidienne, hebdomadaire et annuelle. Par exemple, une semaine de travail de 5 jours peut produire des effets sur une série chronologique qui se répètent chaque semaine, tandis que les horaires de vacances et les pauses scolaires peuvent produire des effets qui se répètent chaque année. Pour ajuster et prévoir ces effets, nous devons spécifier des modèles de saisonnalité qui sont des fonctions périodiques de t .

Le prophète s'appuie sur la série de Fourier pour fournir un modèle qui prend en compte les effets périodiques. Soit P est la période régulière de la série chronologique (par exemple $P = 365,25$ pour les données annuelles ou $P = 7$ pour les données hebdomadaires, lorsque le temps est échelonné en jours).

Donc, il fait une approximation arbitraire pour les effets saisonniers :

$$s(t) = \sum_{n=1}^N (a_n \cos(\frac{2\pi nt}{P}) + b_n \sin(\frac{2\pi nt}{P})) \quad (4.9)$$

Pour ajuster la saisonnalité, il faut estimer les $2N$ paramètres $\mu = [a_1, b_1, \dots, a_N, b_N]^\top$. Cela se fait en construisant une matrice de vecteurs de saisonnalité pour chaque valeur de t dans les données historiques et futures.

Exemple Pour N=10 :

$$H(t) = \left[\cos\left(\frac{2\pi(1)t}{P}\right), \sin\left(\frac{2\pi(1)t}{P}\right), \dots, \cos\left(\frac{2\pi(10)t}{P}\right), \sin\left(\frac{2\pi(10)t}{P}\right) \right] \quad (4.10)$$

Ce qui signifie que la composante saisonnière est :

$$s(t) = H(t)\mu \quad (4.11)$$

La troncature de la série à N applique un filtre passe-bas à la saisonnalité, donc, bien qu'avec un risque augmenté de sur-ajustement, l'augmentation de N permet d'ajuster des modèles saisonniers qui changent plus rapidement.

4.2.3 Vacances et événements :

L'impact d'un jour férié particulier sur la série chronologique est souvent similaire année après année, ce qui en fait une incorporation importante dans les prévisions. La composante $h(t)$ parle des événements prévisibles de l'année, y compris ceux sur des horaires irréguliers. Pour utiliser cette fonctionnalité, l'utilisateur doit fournir une liste d'événements personnalisée. La fusion de cette liste de vacances dans le modèle est simple en supposant que les effets des vacances sont indépendants. Pour chaque jour férié i , soit D_i l'ensemble des dates passées et futures de ce jour férié. Ensuite, il doit ajouter une fonction d'indicateur représentant si le temps t est pendant les vacances i et il doit affecter à chaque jour férié un paramètre ϕ_i qui est le changement correspondant dans la prévision. Cela se fait de la même manière que la saisonnalité en générant une matrice de régression :

$$Z_j(t) = \begin{cases} 1, & t \in D_j \\ 0, & \text{sinon} \end{cases} \quad (4.12)$$

$\forall j \in \{1, \dots, L\}$ avec L le nombre des événements et jours fériés. On aura donc :

$$h(t) = Z(t)\phi \quad (4.13)$$

Il est souvent important d'inclure des effets pour une fenêtre de jours autour d'un jour férié particulier. Pour tenir compte de cela, le prophète inclue des paramètres supplémentaires pour les jours entourant le jour férié, en traitant essentiellement chacun des jours dans la fenêtre autour du jour férié comme un jour férié lui-même. Ce que nous avons montré avec des exemples sur Python où nous essayons d'appliquer toutes les fonctionnalités citées précédemment sur nos jeux de données.

4.3 Avantages et Limites

4.3.1 Avantages

1)Obtention d'une prévision raisonnable sur des données en désordre sans effort manuel. Prophet résiste aux valeurs aberrantes, aux données manquantes et aux changements spectaculaires dans la série chronologique.

2)La procédure Prophet comprend de nombreuses possibilités pour les utilisateurs de modifier et d'ajuster les prévisions. Il est possible d'utiliser des paramètres interprétables pour améliorer les prévisions en ajoutant les connaissances de domaine.

4.3.2 Limites

1) Si il faut prédire des centaines ou des milliers de cibles simultanément, Prophet calculera lentement. Il doit être adapté à la prévision multivariée en codant plusieurs boucles et en stockant plusieurs prédictions. Prophet ne recoupera pas les schémas potentiellement utiles à partir de plusieurs cibles.

2) Avec des fonctionnalités plus significatives que la saisonnalité ou des événements spéciaux, Prophet n'aidera pas. C'est une grosse limitation dans beaucoup de situations. Même les défis simples de prédiction de la demande impliquent souvent une hiérarchie de produits à plusieurs niveaux et certaines données contextuelles.

5 VAR - VARMA

Dans cette partie, on travaillera sur des séries temporelles multivariées, c-à-d, des séries à plusieurs colonnes. On suppose que ces variables sont toutes stationnaires. Soit $n \in \mathbb{N}^*$ le nombre de ces variables.

5.1 Vecteur Autoregressif - VAR

5.1.1 Définition

Soit $(\Omega, \mathcal{F}, \mathbb{P})$ un espace probabilisé. En notant \mathcal{L} l'opérateur sur l'ensemble des processus stationnaires à valeurs dans \mathbb{R}^n défini par $\mathcal{L}((Y_t)_{t \in \mathbb{Z}}) = (Y_{t-1})_{t \in \mathbb{Z}}$, on dit que Y suit un modèle $VAR(p)$, avec $p \in \mathbb{N}^*$ quelconque, s'il existe une famille $(\Phi_k)_{k \in \llbracket 1, p \rrbracket}$ de matrices de $\mathcal{M}_n(\mathbb{R})$ et un bruit blanc $(\epsilon_t)_{t \in \mathbb{Z}}$ défini sur $(\Omega, \mathcal{F}, \mathbb{P})$ à valeurs dans \mathbb{R}^n , tels que pour tout $t \in \mathbb{Z}$:

$$Y_t = \sum_{k=1}^p \Phi_k \mathcal{L}^k(Y)_t + \epsilon_t \quad (5.1)$$

C'est-à-dire, pour tout $t \in \mathbb{Z}$:

$$\begin{pmatrix} y_{1,t} \\ y_{2,t} \\ \vdots \\ y_{n,t} \end{pmatrix} = \sum_{k=1}^p \begin{pmatrix} \phi_{k,11} & \cdots & phi_{k,1n} \\ \phi_{k,21} & \cdots & phi_{k,2n} \\ \vdots & \ddots & \vdots \\ \phi_{k,n1} & \cdots & phi_{k,nn} \end{pmatrix} \begin{pmatrix} y_{1,t-k} \\ y_{2,t-k} \\ \vdots \\ y_{n,t-k} \end{pmatrix} + \begin{pmatrix} \epsilon_{1,t-k} \\ \epsilon_{2,t-k} \\ \vdots \\ \epsilon_{n,t-k} \end{pmatrix} \quad (5.2)$$

5.1.2 Causalité au sens de Granger

On dit qu'une série temporelle X cause une autre série Y lorsque la connaissance du passé de Y conduit à une prévision de Y différente de celle fondée uniquement sur le passé de Y . Ainsi, la connaissance de la série X n'est pas neutre pour la prévision de Y .

Dans le cadre des modèles paramétriques VAR, Granger a mis en place un test statistique simple permettant de tester si une série en cause une autre. Soit l'hypothèse nulle \mathcal{H}_0 : « X ne cause pas Y au sens de Granger » et l'alternative \mathcal{H}_1 : « X cause Y au sens de Granger ». Le test proposé par Granger est le test de nullité globale des coefficients des $(X_{t-k})_{k \in \llbracket 1, p \rrbracket}$ dans la régression linéaire suivante :

$$Y_t = \sum_{k=1}^p \Phi_{k,11} Y_{t-k} + \sum_{k=1}^p \Phi_{k,12} X_{t-k} \quad (5.3)$$

Ainsi, pour tester si X cause Y , Granger propose de régresser la valeur présente de Y sur le passé de Y et X , et de tester si les coefficients devant les valeurs passées de X sont nuls (auquel cas X ne cause pas Y au sens de Granger). Les méthodes standards (moindres carrés ordinaires) permettent de mener à bien ce test de causalité.

5.1.3 Choix du paramètre p

Afin de sélectionner le paramètre optimal pour un modèle VAR, l'approche pratique consiste à :

- Imposer un maximum p_{max} en se basant sur une connaissance préalable du domaine des données.
- Sélectionner la valeur de p qui minimise des critères comme AIC ou HQ.
- Vérifier la qualité du modèle à partir du résidu en s'assurant qu'il ne contient plus d'information (autocorrélation) et qu'il suit bien une loi normale (cas optimal).

5.1.4 Avantages du modèle VAR

- 1- Le paramètre p est relativement facile à estimer.
- 2- Il a une bonne capacité de prévision.
- 3- L'utilisateur n'a pas besoin de préciser quelles variables sont endogène ou exogène. Elles sont toutes endogènes.
- 4- A l'aide du modèle VAR, il est très facile de tester la non-causalité de Granger.

5.1.5 Limites du modèle VAR

- 1- Le nombre de variables à déterminer peut s'avérer très grand. S'il y a n équations, une pour chaque n variables et p décalages pour chacune des variables dans chaque équation, on aurait à estimer $(n + pn^2)$ paramètres à estimer.
- 2- Le modèle VAR est athéorique car il se base sur peu de théorie économique et deonc peut donner des résultats qui ne sont pas conformes avec les théories économiques.

5.2 Vecteur Autoregressif Moyenne Mobile - VARMA

5.2.1 Définition

Pareillement à la définition du modèle $VAR(p)$, on dit qu'une série temporelle multivariée Y suit un modèle $VARMA(p, q)$ avec $(p, q) \in (\mathbb{N}^*)^2$ s'il existe deux familles $(\Phi_k)_{k \in \llbracket 1, p \rrbracket}$ et $(\Psi_k)_{k \in \llbracket 1, q \rrbracket}$ de matrices de $\mathcal{M}_n(\mathbb{R})$ et un bruit blanc $(\epsilon_t)_{t \in \mathbb{Z}}$ défini sur $(\Omega, \mathcal{F}, \mathbb{P})$ à valeurs dans \mathbb{R}^n , tels que pour tout $t \in \mathbb{Z}$:

$$Y_t = \sum_{i=1}^p \Phi_i \mathcal{L}^i(Y)_t + \sum_{j=1}^q \Psi_j \mathcal{L}^j(\epsilon)_t + \epsilon_t \quad (5.4)$$

avec \mathcal{L} l'opérateur sur l'ensemble des processus stationnaires à valeurs dans \mathbb{R}^n défini par $\mathcal{L}((Y_t)_{t \in \mathbb{Z}}) = (Y_{t-1})_{t \in \mathbb{Z}}$.

On peut exprimer (5.2) avec des notations plus évidentes, pour tout $t \in \mathbb{Z}$:

$$\begin{pmatrix} y_{1,t} \\ y_{2,t} \\ \vdots \\ y_{n,t} \end{pmatrix} = \sum_{i=1}^p \begin{pmatrix} \phi_{i,11} & \cdots & \phi_{i,1n} \\ \phi_{i,21} & \cdots & \phi_{i,2n} \\ \vdots & \ddots & \vdots \\ \phi_{i,n1} & \cdots & \phi_{i,nn} \end{pmatrix} \begin{pmatrix} y_{1,t-i} \\ y_{2,t-i} \\ \vdots \\ y_{n,t-i} \end{pmatrix} + \sum_{j=1}^q \begin{pmatrix} \psi_{j,11} & \cdots & \psi_{j,1n} \\ \psi_{j,21} & \cdots & \psi_{j,2n} \\ \vdots & \ddots & \vdots \\ \psi_{j,n1} & \cdots & \psi_{j,nn} \end{pmatrix} \begin{pmatrix} \epsilon_{1,t-j} \\ \epsilon_{2,t-j} \\ \vdots \\ \epsilon_{n,t-j} \end{pmatrix} + \begin{pmatrix} \epsilon_{1,t} \\ \epsilon_{2,t} \\ \vdots \\ \epsilon_{n,t} \end{pmatrix} \quad (5.5)$$

5.2.2 Avantages du modèle VARMA

- 1- Le modèle VARMA est un outil très puissant pour produire des prévisions linéaires pour une série chronologique multivariée.
- 2- Il utilise les informations non seulement contenues dans les valeurs passées d'une variable particulière, mais permettent également d'exploiter des informations dans d'autres variables.
- 3- Le modèle VARMA peut fournir des prédictions avec une grande précision notamment pour des séries mutivariées avec peu de variables et à court terme.

5.2.3 Limites du modèle VARMA

- 1- Opter pour des prévisions linéaires n'est pas toujours le meilleur choix
- 2- L'ajout de variables dans un système n'augmente pas nécessairement la précision des prévisions. Les systèmes de grande dimension sont généralement plus difficiles à spécifier.
- 3- Ce modèle ne prend pas en charges des prévisions à long terme et donc peut donner de mauvais résultats.

6 Resultats obtenus

6.1 Electrolysia

Ce jeu de données ne permet pas aux algorithmes d'avoir un ensemble d'apprentissage suffisant pour une prédiction à long terme. Sur ce, nous nous sommes contentés uniquement sur une prédiction à court terme.

ARIMA :

Pour ARIMA, nous avons procédé d'une autre manière. Nous avons essayé de prédire la consommation pour chaque 6h afin de trouver les pointes de la consommation.

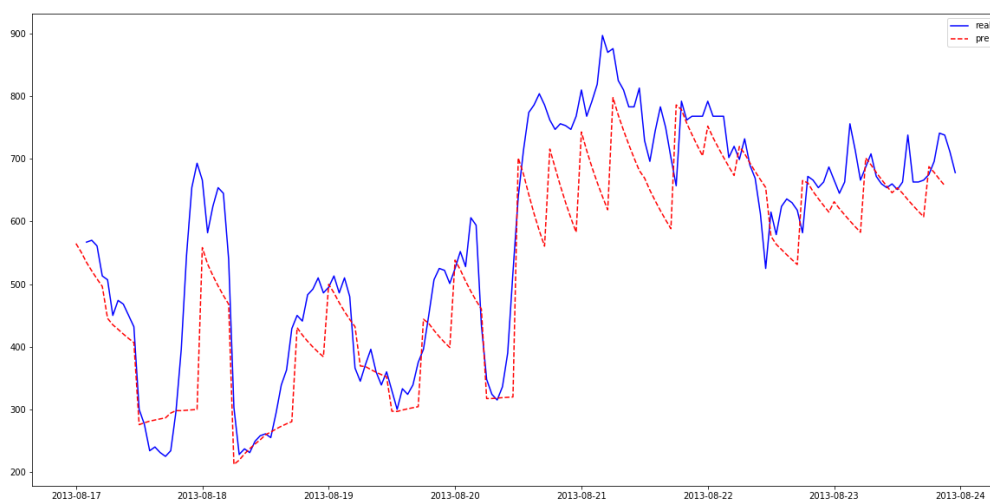


FIGURE 23 – Comparaison des résultats obtenus par ARIMA avec les données réelles pour le jeu de données Electrolysia.

FbProphet :

Nous avons récupéré 22 jours de la consommation électrique du 1^{er} mois de notre jeu de données, pour ajuster le modèle de FbProphet, et nous avons essayé de prédire la consommation pour les 2 jours suivants.

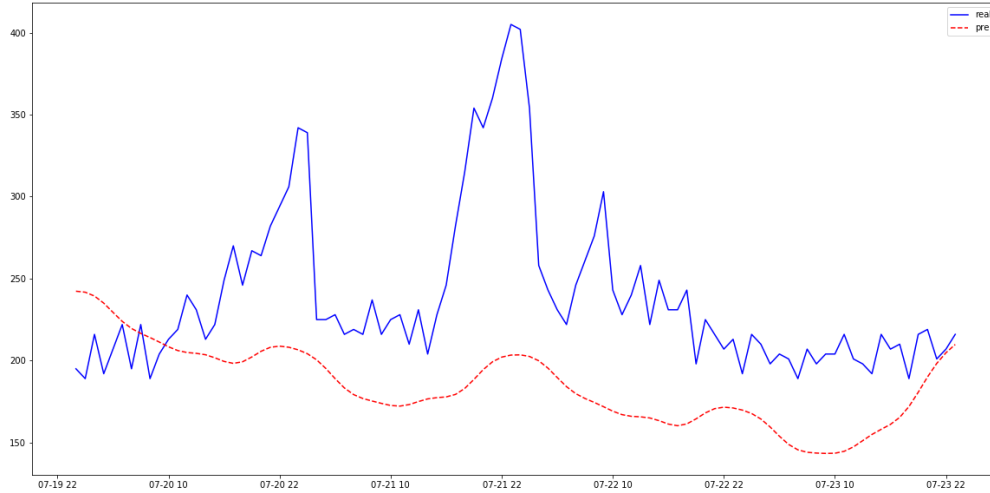


FIGURE 24 – Comparaison des résultats obtenus par FbProphet avec les données réelles pour le jeu de données Electrolysis.

Ce résultat confirme l'une des limites de FbProphet ; nous ne pouvons pas avoir de bons résultats si l'ensemble d'apprentissage n'est pas assez grand.

VAR :

D'une manière pareille à celle utilisée pour le modèle ARIMA, nous avons opté pour une prédiction pour les prochaines 6 heures, c'est-à-dire mettre la base d'apprentissage à jour avec les données réelles une fois toutes les 6 heures. Nous avons utilisé les données de la température uniquement, car seule la série chronologique univariée de la température qui cause au sens de Granger la série de la consommation électrique.

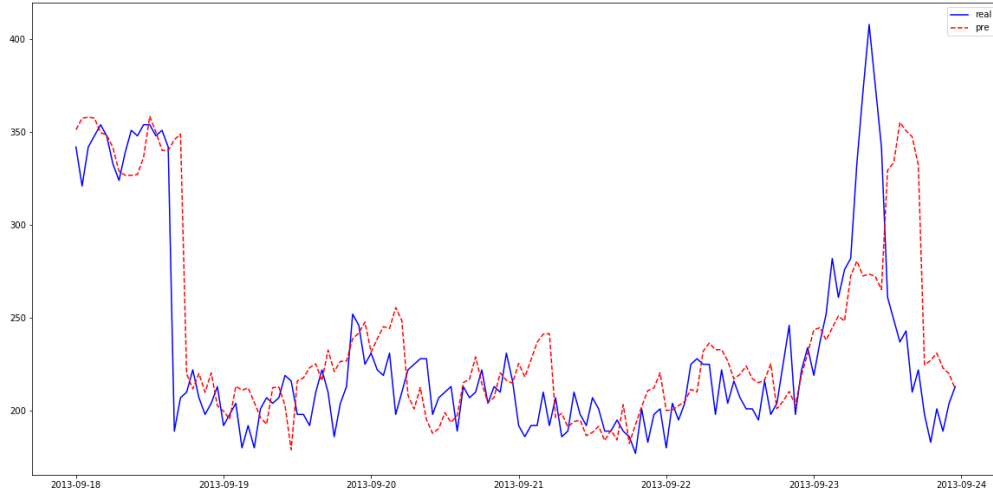


FIGURE 25 – Comparaison des résultats obtenus à l'aide du modèle VAR avec les données réelles pour le jeu de données Electrolysis.

VARMA :

En appliquant la même logique à l'aide du modèle VARMA avec $p = 1$ et $q = 4$, on obtient le résultat suivant avec une erreur quadratique inférieure à celle obtenue à l'aide du modèle VAR

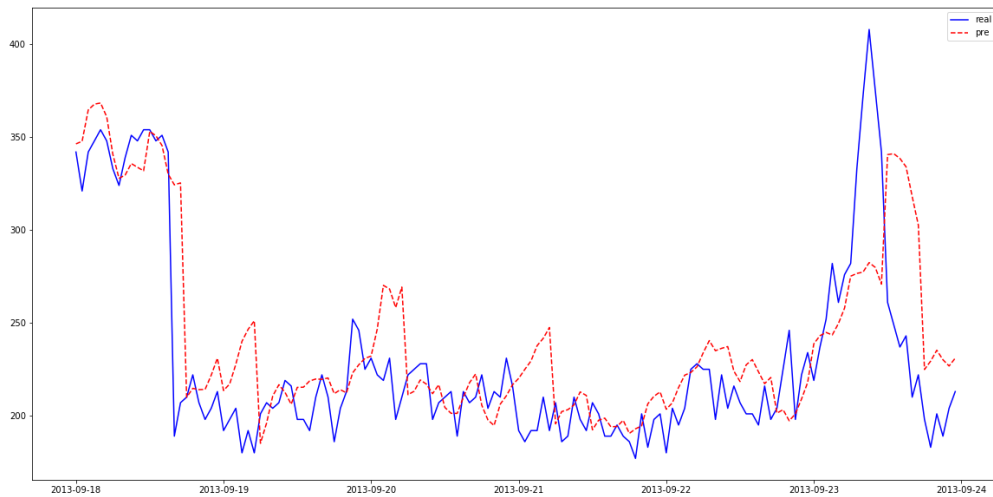


FIGURE 26 – Comparaison des résultats obtenus à l'aide du modèle VARMA avec les données réelles pour le jeu de données Electrolysis.

6.2 RU_Electricity

Contrairement au jeu de données précédent, ce jeu de données permet d'avoir un ensemble d'apprentissage très large, car il ne contient pas des valeurs manquantes.

FbProphet :

Nous avons récupéré 70% de ce jeu de données pour l'apprentissage du modèle, et nous avons prédit la consommation pour 2 mois.

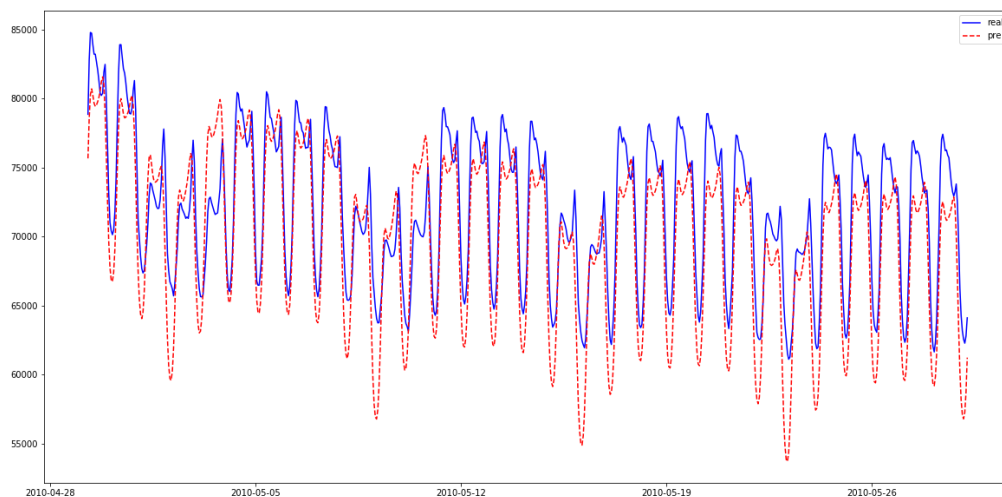


FIGURE 27 – Comparaison des résultats obtenus par FbProphet avec les données réelles pour le jeu de données RU_Electricity.

On remarque que FbProphet n'a pas réussi à prédire la consommation avec une grande précision, mais il a réussi à prédire la tendance avec laquelle la consommation évolue.

ARIMA :

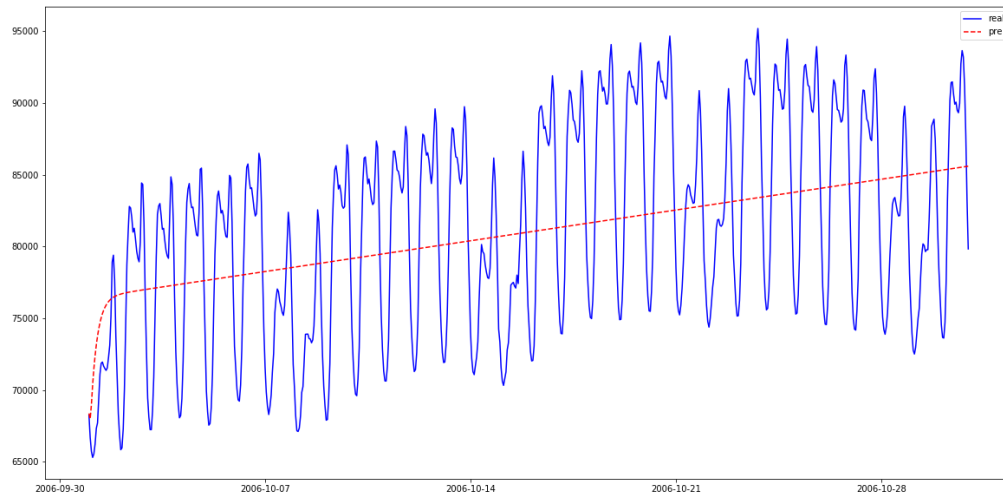


FIGURE 28 – Comparaison des résultats obtenus par ARIMA avec les données réelles pour la prédiction à long terme pour le jeu de données RU_Electricity.

ARIMA n'a pas réussi à prédire la consommation à long terme même si il a eu un ensemble d'apprentissage très important.

Conclusion

Le problème de la prévision de la consommation électrique peut s'avérer très compliqué en terme de choix d'algorithme. En général la famille des modèles ARMAX donne des résultats très puissants à court terme et avec peu de données du passé. Les modèles VAR et VARMA permettent d'exploiter des données d'autres variables pour une prédiction plus précise. Le choix des variables exploitables se fait grâce au test de causalité de Granger. Mais puisque ces algorithmes ne fournissent de bonne prédiction qu'à court terme, on a opté en guise d'optimisation pour mettre à jour la base d'apprentissage une fois toutes les 6 heures, une durée qu'on a estimé suffisante pour pouvoir détecter les pointes de consommation électrique et donc éviter à ce que la demande soit plus importante que l'offre.

Contrairement à cette famille de modèles, le modèle open source de Facebook Fbprophet donne des prédictions à grande précision à long terme. Toutefois, ce modèle nécessite une base d'apprentissage très grande (minimum un an) pour pouvoir fournir de bons résultats, chose qu'on a pu constater en l'essayant sur deux jeu de données complètement différents.

Durant notre travail sur ce projet, on a trouvé beaucoup de difficultés notamment pour le choix des algorithmes et des méthodes d'optimisation surtout que nos connaissances sur les séries temporelles étaient très faibles. On a réussi à surmonter une grande partie de ces difficultés grâce à l'aide de notre tuteur M. Vincent BARRA que nous tenons à remercier encore une fois.

Références

- [1] Gwilym M.Jenkins, Gregory C.Reinsel, George E.P.Box «Time Series Analysis : Forecasting and Control, Fourth Edition», New Jersey, 2008.
- [2] Aileen Nielsen «Practical Time Series Analysis Prediction with Statistics and Machine Learning» O'Reilly, 2019.
- [3] BARADEL Nicolas et CONNEAU Fabien «Indicateur de chocs financiers et stratégies de portefeuille : une méthode de réduction de la Value at Risk à horizon 1 an au niveau 99,5%», Paris, 2012.
- [4] Helmut LÜTKEPOHL «Forecasting with VARMA Models», San Domenico, 2004.
- [5] Vincent Thouvenot, «Estimation et sélection pour les modèles additifs et application à la prévision de la consommation électrique», Université Paris-Saclay, 2015.
- [6] Gilbert COLLETAZ, «LES PROCESSUS VAR notes du cours de séries temporelles multivariées», Université d'Orléans, 2019.
- [7] MOOC Udemy :«Python for Time Series Analysis and Forecasting», (date de consultation 12/2019).
- [8] <https://machinelearningmastery.com/multi-step-time-series-forecasting-with-machine-learning/> (date de consultation 12/2019)
- [9] <https://towardsdatascience.com/prophet-vs-deepar-forecasting-food-demand-2fdebfb8d282> (date de consultation 02/2020)
- [10] <https://www.machinelearningplus.com/time-series/arma-model-time-series-forecasting-python/> (date de consultation 11/2019)
- [11] <https://towardsdatascience.com/a-quick-start-of-time-series-forecasting-with-a-practical-ex> (date de consultation 02/2020)

Glossaire

ACF : La fonction d'autocorrelation.

ADF : Le test de Dickey Fuller Augmenté

AIC : Le critère d'information d'Akaike.

ARIMA : Moyenne mobile intégrée autorégressive.

BIC : Le Le critère d'information bayésien.

cov() : La fonction de covariance.

FbProphet : Le prophète de Facebook.

PACF : La fonction d'autocorrelation partielle.

VARMA : Vecteur Autoregressif Moyenne Mobile.

VAR : Vecteur Autoregressif.

var() : La fonction de la variance.