



Mémoire de Master

Filière : Mathématiques

Spécialité : Recherche Opérationnelle



Thème

**Etude comparative entre les prévision par lissage
exponentiel et lissage combiné avec le machine learning :
application à la chronique lissage de la citrouille**

Présenté par :

- BOUKERDOUN KARIMA
- CHENAFI OUALID

Devant le jury composé de :

Président	M ^r Omar HAMDOUNI	MAA	U. A/M/O Bouira.
Encadreur	M ^r L'hadi BOUGHANI	MAA	U. A/M/O Bouira.
Examineurs	M ^r Elghani IFTISSENE	MAA	U. A/M/O Bouira.
	M ^r Karim HAMID	MAA	U. A/M/O Bouira.

Dédicaces

De Karima

*Aux êtres les plus chers à mon coeur, **MA MÈRE** et **MON PÈRE**, qui m'ont beaucoup soutenu durant mon cursus d'étude et à qui je serai éternellement reconnaissante, qui ont tout enduré pour que je devienne ce que je suis aujourd'hui.*

*À mon frère **DJAMEL**.*

*À mes soeur **HAFIDHA, MANEL, ANFEL, HANANE**.*

*À mon binôme : **WALID**.*

*À tous les membres des familles : **BOUKERDOUN, HADDAD**.*

Sans oublier tout les professeurs que ce soit du primaire ,du moyen,du secondaire ou de l'enseignement supérieur.

À Tous mes collègues d'étude et surtout la promotion 2022/2023.

Sans oublier tout les professeurs que ce soit du primaire ,du moyen,du secondaire ou de l'enseignement supérieur.

À Toutes et à Tous.

Je vous dédie ce modeste travail

Dédicaces

De Walid

*Aux êtres les plus chers à mon coeur, **MA MÈRE** et **MON PÈRE**, qui m'ont beaucoup soutenu durant mon cursus d' étude et à qui je serai éternellement reconnaissante, qui ont tout enduré pour que je devienne ce que je suis aujourd'hui.*

*À mes frères **MOHAMED ET SOUFIANE**.*

*À mes soeur **LYDIA ET HANANE**.*

*À ma bînome : **KARIMA**.*

*À tous les membres des familles : **CHENAFI, ZAIDI**.*

À Tous mes collègues surtout la promotion 2022/2023.

À Toutes et à Tous.

Je vous dédie ce modeste travail

Remerciements

Nos remerciements vont tout premièrement à Allah le tout puissant pour la volonté, la santé et la patience qu'il nous a donné pour terminer ce travail.

Nous remercions notre promoteur M. L'HADI BOUGHANI pour avoir accepté de diriger ce travail ainsi que pour ses conseils avec beaucoup de patience et d'encouragements.

Nous exprimons nos remerciements aux membres du jury qui ont accepté de juger notre travail, en l'occurrence Mr HAMDOUNI OMAR de présider le jury de soutenance et Messieurs HAMID KARIM ET IFTISSEN ELGHANI d'examiner ce travail.

Nos sincères remerciements s'adressent enfin à tous ceux qui nous ont soutenu de près ou de loin.

Enfin ,nous remercions les professeurs Amel Belhadj,Ouidja Daya ,Demouche Nacer et Bedak Said

Mecri à Tous

Résumé

Le travail réalisé dans ce mémoire consiste, en se basant sur des chroniques obtenues lors du séchage de la citrouille, de créer modèle de prévision à l'aide du lissage exponentiel sur le logiciel R puis appliquer une des méthodes du machine learning dans l'objectif d'améliorer la précision des prévisions obtenues par le lissage.

Mots Clés : Série chronologique, Lissage exponentiel, Holt-Winter, Tendence, saisonnalité, Machine learning, apprentissage supervisé, apprentissage non supervisé, Langage R.

Table des matières

Liste des tableaux	3
Table des figures	4
Introduction Générale	6
1 Méthodes de lissage exponentiel pour la prévision	8
1.1 Généralités sur les séries chronologiques	8
1.1.1 Définition d'une série chronologique	8
1.1.2 Composantes d'une série chronologique	9
1.1.3 Les différents modèles de série chronologique	9
1.2 Série stationnaire	10
1.2.1 Bruit blanc	10
1.2.2 Les séries non Stationnaires	11
1.3 Les méthodes de prévision	11
1.3.1 Méthodes extrapolatives	11
1.3.2 Méthodes explicatives	11
1.4 Lissage exponentiel	12
1.4.1 lissage exponentiel Simple	12
1.4.2 Lissage exponentiel double :	15
1.4.3 Lissage de Holt Winters	16

1.5	Commandes de lissages exponentiel dans R	18
2	Machine Learning	21
2.1	Définition et Histoire	21
2.2	Importance et applications du Machine Learning	22
2.3	Méthodes d'apprentissage	22
2.3.1	Apprentissage supervisé	23
2.3.2	non supervisé machine learning	28
2.3.3	Apprentissage semi-supervisé	31
2.3.4	Apprentissage par Renforcement	32
3	Prévision par lissage exponentiel et lissage-machine learning	34
3.1	Modél de lissage exponentiel	35
3.1.1	Modélisation de la série 200w	35
3.1.2	Résultats du lissage des séries de puissances (400W.600W.800W)	38
3.2	Modélisation par lissage combiné avec le machine learning	39
3.2.1	Modélisation de la série 200w	39
3.3	Modèle Lissage-Machine Learning	39
3.3.1	Réultats et discussion	40
	Conclusion Générale	42
3.4	Annexe3 :Résultat du Holt Winter non saisonnier des séries sur "R" . . .	47
	Annexes	45
	Bibliographie	49

Liste des tableaux

1.1	les composantes de chaque méthode de lissage	12
2.1	Différence entre régression et classification	27
2.2	Différence entre apprentissage supervisé et non supervisé	31
3.1	Les résultats de HLW de chaque série	38
3.2	Somme des carrés des résidus moyenne dans chaque modèle	41
3.3	Les résultats des paramètres HLW de chaque puissance	48

Table des figures

2.1	Type d'apprentissage en ML	23
2.2	Apprentissage supervisé	24
2.3	Régression vs Classification.	25
2.4	l'apprentissage automatique	28
2.5	les types de problèmes d'apprentissage non supervisé	29
2.6	30
2.7	Clustering.	31
2.8	Supervised vs Unsupervised Learning	32
2.9	Apprentissage par renforcement concept	33
3.1	Environnement Générale de " R et RStudio"	35
3.2	Le graphe de la série de puissance 200 W	36
3.3	Le résultat LED de 200 W.	37
3.4	Resultats ADF Residus	37
3.5	Graphe de la série originale de puissance 400W	46
3.6	Graphe de la série originale de puissance 600W	46
3.7	Graphe de la série originale de puissance 800W	47
3.8	Résultat de LHW de 400W	47
3.9	Résultat de HLW de 600w	48
3.10	Résultat de LHW de 800W	48

liste d'abréviations

TS	Trend Stationnary.
DS	Differency Stationnary.
LES	Lissage exponentiel simple.
LED	Lissage exponentiel double.
LHW	Lissage de Holt winters.
LM	Machine learnig.
SSR	Sum of Squared Error .

Introduction Générale

Les méthodes de prévision des séries temporelles sont des techniques utilisées pour analyser et prévoir les tendances et les modèles des données séquentielles dans le temps. Plusieurs approches sont couramment utilisées dans la prévision des séries temporelles et ce suivant la nature de la série et la taille de l'historique des données ; parmi ces approches se situe le lissage exponentiel.

Le lissage exponentiel est une méthode simple mais puissante pour la prévision des séries temporelles. Elle repose sur l'idée de donner plus de poids aux observations récentes et moins de poids aux observations plus anciennes. Le lissage exponentiel est basé sur une combinaison linéaire des observations passées, où les coefficients sont déterminés par un facteur de lissage. Il existe plusieurs variantes de lissage exponentiel, telles que le lissage exponentiel simple, le lissage exponentiel double et le lissage exponentiel triple (ou de Holt-Winters) [6], qui prennent en compte différents niveaux de tendance et de saisonnalité dans les données.

D'autre part, l'apprentissage automatique offre une approche plus complexe pour la prévision des séries temporelles en utilisant des algorithmes d'apprentissage supervisé ou non supervisé. Les modèles d'apprentissage automatique peuvent capturer des relations non linéaires et des interactions complexes entre les variables. Parmi les algorithmes d'apprentissage automatique couramment utilisés pour la prévision des séries temporelles, on trouve les arbres de décision, les forêts aléatoires, les réseaux de neurones et les méthodes de régression linéaire. Ces modèles apprennent à partir des

données historiques et peuvent être utilisés pour prédire les valeurs futures de la série temporelle.

L'avantage de l'utilisation des méthodes de lissage exponentiel est leur simplicité et leur facilité d'interprétation. Elles sont souvent utilisées lorsque les données ne présentent pas de tendance complexe ou de saisonnalité prononcée. En revanche, les modèles d'apprentissage automatique offrent une flexibilité accrue pour modéliser des séries temporelles plus complexes, avec des tendances non linéaires, des sauts, des motifs saisonniers irréguliers, etc.

Il est important de noter que le choix de la méthode de prévision dépendra de la nature des données, de la disponibilité des données historiques, des objectifs spécifiques de prévision et des contraintes du problème. Dans de nombreux cas, il peut être judicieux de combiner différentes méthodes pour obtenir des prévisions plus précises et robustes.

Dans ce mémoire, on essayera de comparer les résultats des prévisions de la série séchage de la citrouille via le lissage seulement puis via le lissage combiné avec les méthodes d'apprentissage du machine learning, le travail sera donc articulé comme suit :

- On commencera par une introduction générale
- Le premier chapitre de ce mémoire comportera l'essentiel de la théorie des prévision par lissage exponentiel
- En deuxième chapitre sera exposé brièvement les notions de base du machine learning et ces méthodes d'apprentissage.
- Le troisième chapitre portera sur l'application du lissage et lissage combiné avec le machine learning dans la modélisation de la série séchage de la citrouille et on terminera le travail par une conclusion générale et quelques perspectives.

Méthodes de lissage exponentiel pour la prévision

Introduction

La prédiction des réalisations futures est l'un des principaux objectifs de l'étude d'une série temporelle. Les méthodes de lissage exponentiel, introduite d'abord par Holt en 1958, Winters en 1960, et rendu populaire par le livre de Brown en 1963[6], comprennent l'ensemble des approches de prédiction empirique à court terme qui supposent que le phénomène étudié dépend seulement de ses valeurs passées. Ces techniques sont largement utilisées et diffusées. Son succès est le résultat à la fois de leur simplicité et de l'exactitude de leurs prévisions

1.1 Généralités sur les séries chronologiques

1.1.1 Définition d'une série chronologique

Une série chronologique (ou encore une série temporelles ou chronique) est une suite réelle finie $(x(t))_{0 \leq t \leq n}$ où t représente le temps et peut être en minute, heure, jour ..., et 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. Les points ainsi obtenus sont reliés par des segments de droite et le graphe apparaît donc comme une ligne brisée

1.1.2 Composantes d'une série chronologique

En général, une série chronologique comporte 4 composantes décrivant son évolution dans le temps et la mise en oeuvre de ces composantes permettra de mettre un modèle mathématique pour prévoir des valeurs futures de x_t . Ces composantes sont :

- **La tendance** : Elle décrit l'évolution à court terme de la série x_t : l'évolution peut être à la hausse ou bien à la baisse et traduit le comportement moyen de la série, elle est notée T_t
- **Saisonnalité** : Elle décrit une partie de la série qui est répétée sur des périodes de temps homogènes. Graphiquement cette composante est identifiée par des pics qui se répètent dans des intervalles de temps réguliers dit période P , elle est notée s_t
- **La composante cyclique** : Elle décrit l'évolution de la série à long terme. Cette composante se manifeste dans le cas d'un historique de données très important de x_t , elle est notée C_t
- **Composante résiduelles** : ou variations accidentelles est la partie non structurée du phénomène. elle est notée ϵ_t

1.1.3 Les différents modèles de série chronologique

Généralement une série chronologique x_t est modélisée mathématiquement suivant deux schémas :

schéma additif

Il suppose que l'évolution de x_t est définie comme la somme de ses composantes :

$$x_t = T_t + S_t + C_t + \epsilon_t$$

Ce modèle est utilisé dans le cas de séries à variations constante dans le temps.

schéma multiplicatif

Il suppose que l'évolution de la série est le produit de ces composantes :

$$x_t = T_t \times S_t \times C_t \times \epsilon_t$$

Ce modèle est utilisé pour modéliser les séries à variations proportionnelles dans le temps : graphiquement les fluctuations évoluent coniquement avec le temps.

1.2 Série stationnaire

Définition[14] Une série x_t est dite stationnaire si et ssi elle ne comporte ni tendance ni saisonnalité.

théorème[14]

Une série x_t est stationnaire si et ssi :

1. $E(x_t) = E((x_t)^2) = \alpha, \forall t \in Z$ (donc ne dépend pas de temps)
2. $var(x_t) = \sigma^2 < \infty$
3. $cov(x_t, x_{t-h}) = \gamma(h), \forall t \in Z, \forall h \in Z$ (ne dépend pas du temps)

1.2.1 Bruit blanc

Définition

On dit que la suite de variables aléatoires ϵ_t constitue un bruit blanc si elle possède les propriétés suivantes :

$$\begin{cases} E(\epsilon_t) = 0 \quad \forall t \in Z \\ E((\epsilon_t)^2) = \sigma^2 \\ cov(\epsilon_s, \epsilon_t) = 0, \text{ si } s \neq t \end{cases} \quad (1.1)$$

1.2.2 Les séries non Stationnaires

Un processus est non stationnaire s'il présente au moins l'une des composantes tendancielle ou saisonnière.

L'analyse des processus stationnaires est plus facile à établir et donne de meilleurs résultats par rapport aux processus non stationnaires, c'est pour cela que tout processus non stationnaire se doit d'être stationnarisé.

En pratique, les cas de non stationnarité sont analysés à partir de deux types de processus. Les processus TS (Trend Stationary) caractérisés par une non stationnarité de nature déterministe, et les processus DS (Difference Stationary) présentant une non stationnarité de nature stochastique. Dans le cas de processus TS, les données suivent une tendance qui a une fonction définie (linéaire, quadratique, ...).

Plusieurs tests ont été développés à fin de vérifier un processus non stationnaire est de type TS ou DS. Le plus commun est le Test de Dickey et Fuller et Dickey et Fuller augmenté [11].

1.3 Les méthodes de prévision

. Plusieurs méthodes de prévision existent, elles peuvent être regroupées en deux grandes classes :

1.3.1 Méthodes extrapolatives

Ces méthodes utilisent le passé de la variable elle même. Seul le passé de la variable est utilisé en vue de la prévoir sans apport d'information extérieure (lissage exponentielle et modélisation ARMA ...). C'est le cas de séries temporelles.

1.3.2 Méthodes explicatives

Celles ci utilisent les valeurs passées et présentes d'une ou plusieurs variables exogènes pour prévoir y (régression linéaire simple et multiple ...).

1.4 Lissage exponentiel

Les Méthodes de lissage reposent sur l'idée que le passé le plus récent a plus de poids dans la prévision du futur et donc le passé loin a moins d'importance.

la diversité des fonctions mathématiques utilisées dans l'extrapolation (pouvant être des fonctions linéaires, exponentielles ou même sinusoidales) implique une distinction entre les différentes méthodes du lissage exponentiel (lissage simple, double, triple). La fonction qui est choisie empiriquement dépend le plus souvent d'un certain nombre de paramètres, a_0, a_1, \dots, a_k qui sont calculés sur la base des données historiques disponibles et mis à jour avec l'arrivée de nouvelles observations.

Nous présentons trois types de lissage exponentiel : Lissage exponentiel simple qui ajuste la série si elle ne présente ni tendance ni saisonnalité. Le lissage exponentiel Double qui ajuste une tendance. Le lissage de Holt Winters qui ajuste une série avec tendance et saisonnalité.

Tendance	Saisonnalité	Méthode de lissage
Non	Non	Simple
oui	Non	Double
oui	oui	Holt winters

TABLE 1.1 – les composantes de chaque méthode de lissage

1.4.1 lissage exponentiel Simple

Le lissage exponentiel simple permet d'effectuer des prévisions pour des séries chronologiques dont la tendance ne varie pas avec le temps (constante) et sans saisonnalité. Soit y_t une telle série dont on observe les T premiers instants y_1, \dots, y_T Pour $T \in \mathbb{N}^*$. Le modèle s'écrit :

$$y_t = \alpha + \varepsilon_t$$

Avec : α est une constante dite constante de lissage.

ε_t est la composante irrégulière (bruit).

On cherche à prévoir la valeur y_{t+h} c'est à dire faire une prévision de la série à l'horizon h . étant donné un réel α tel que $\alpha \in [0,1]$ comme la tendance est constante, on cherche une prévision $\hat{y}_T(h)$ sous la forme d'une constante qui s'ajuste le mieux au sens des moindres carrés pondérés au voisinage de T , c'est -à-dire la solution du problème de minimisation :

$$\min_{\alpha} \sum_{j=0}^{T-1} \alpha^j (y_{T-j} - \alpha)^2 \quad (1.2)$$

Définition 1.1 La prévision $\hat{y}_T(h)$ est construite :

- en prenant en compte tout l'historique de la chronique,
- plus on s'éloigne de la base T de la prévision, moins l'influence des observations correspondantes est importante, cette décroissance de l'influence est de type exponentiel. De là vient le nom de la technique,
- on se donne α appelé constante de lissage, avec $0 < \alpha < 1$,

Ainsi

$$\hat{y}_T(h) = (1 - \alpha) \sum_{j=0}^{T-1} \alpha^j y_{T-j} \quad (1.3)$$

$\hat{y}_T(h)$ est une moyenne des observations passées où le poids de chaque observation décroît de façon exponentielle avec la tendance

formule de mise à jour :

La définition précédente vérifie les formules suivantes :

$$\hat{y}_T(h) = \alpha y_{T-1}(h) + (1 - \alpha)\hat{y}_T = \hat{y}_{T-1} + (1 - \alpha)(y_T - \hat{y}_{T-1}(h)) \quad (1.4)$$

La prévision au temps T , telle décrite dans la première partie de l'équation, est une moyenne pondérée de la prévision au temps $T - 1$ et de l'observation de la série au temps T . Elle permet de faire une mise à jour très simple, la deuxième partie de la relation définit la prévision au temps T comme la prévision au temps $T - 1$ corrigée de l'erreur de prévision commise au temps $T - 1$ amortie d'un facteur α

L'initialisation :

La série lissée est initialisée par la valeur $\hat{y}_1(h) = y_1$, ainsi pour T assez grand, la valeur initiale a peu d'influence.

Choix de la constante de lissage :

Un problème important en pratique est celui du choix de la constante de lissage α qui est en générale très subjectif et varie selon le contexte de l'étude et/ou le type de prévision souhaité. En pratique, si on souhaite faire une prévision rigide, on choisira $\alpha \in [0.7; 0.99]$, et si au contraire on souhaite une prévision souple, on choisira $\alpha \in [0.01; 0.3]$, une autre solution dictée par les données, consiste à choisir α comme la solution du problème des moindres carrés ordinaires suivant :

$$\sum_{t=1}^{T-h} (y_{t+h} - \hat{y}_T(h))^2 = (y_{1+h} - \hat{y}_1(h))^2 + \dots + (y_T - \hat{y}_{T-h}(h))^2 \quad (1.5)$$

C'est -à-dire de minimiser la somme des carrés des erreurs de prévision aux dates $1, \dots, T - h$.

On peut aussi ne considérer que les écarts obtenus sur la deuxième moitié de la série, an de ne pas tenir compte de l'initialisation ; On cherche alors α qui minimise :

$$\sum_{t=1}^{T-h} (y_{t+h} + \hat{y}_T(h))^2 \quad (1.6)$$

remarque On peut calculer \hat{y}_T à partir de la formule de mise à jour de LES en initialisant $\hat{Y}_1 = \hat{Y}_0(1) = \alpha Y_1$

1.4.2 Lissage exponentiel double :

Holt [11] a étendu le lissage exponentiel simple au cas du lissage exponentiel linéaire. l'idée est d'ajuster la tendance par une droite au lieu d'une constante dans l'approximation locale de la série .

Définition 1.2 On ajuste le voisinage de T par une droite d'équation :

$$y_t = a + b \times (t - T) = \epsilon_t \quad (1.7)$$

Brown a réécrit le modèle sous la forme [12] :

$$\hat{y}_T = \hat{a}_T + \hat{b}_T \times h \quad (1.8)$$

Les coefficients \hat{a}_T et \hat{b}_T sont solution du problème d'optimisation :

$$\inf_{a \in \mathbf{R}, b \in \mathbf{R}} \sum_{j=0}^{T-1} \alpha^j (y_{T-j} - (a + b_j))^2 \quad (1.9)$$

La solution de ce problème repose sur les termes s_1 (dit série lissée et s_2 dit série doublement lissée définies par :

$$\begin{cases} s_1 = (1 - \alpha) \sum_{i=0}^{t-1} \alpha^i y_{t-i} \\ s_2 = (1 - \alpha) \sum_{i=0}^{t-1} \alpha^i s_1(t-i) \end{cases} \quad (1.10)$$

On a la définition suivant :

Définition 1.3 La prévision de la série $h.\hat{y}_T(h)$ fournie par la méthode de lissage exponentiel double est donnée par :

$$\hat{y}(h) = \hat{a}_T + \hat{b}_T \times h \quad (1.11)$$

le couple (\hat{a}_T, \hat{b}_T) est donnée par :

$$\begin{cases} \hat{a}_T = 2s_1(T) - s_2(T) \\ \hat{b}_T = ((1 - \alpha)/\alpha)(s_1(T) - s_2(T)) \end{cases} \quad (1.12)$$

Formule de mise à jour

Les formules de mise à jour s'obtiennent à partir de ces expressions :

$$\begin{cases} \hat{a}_T = (\hat{a}(T-1)) + \hat{b}(T-1) + (1 - a^2)(y_T - \hat{y}(T-1)) \\ \hat{b}_T = \hat{b}(T-1) + (1 - a^2)(y_T - \hat{y}(T-1)) \end{cases} \quad (1.13)$$

1.4.3 Lissage de Holt Winters

La méthodes de lissage exponentiel double permet de traiter des séries présentant une tendance linéaire mais sans saisonnalité, on peut également définir des lissages exponentiels généralisés sur le même principe que les techniques de LES et LED permettant de traiter des séries avec saisonnalité et sans saisonnalité.

Séries non saisonnière

La méthode de Holt Winters sans saisonnalité est fondée comme la méthode de lissage double, On ajuste au voisinage de l'instant n une droite : $y_t = a + b(t - n)$. La variante proposée par Holt-Winters porte sur les formules de mise à jour dans l'estimation de a et b par les formules(2.12) qui prennent la forme suivant :

Soient deux constantes de lissages $0 < \gamma < 1$ et $0 < \beta < 1$ Les formules de mise à jour sont :

$$\begin{cases} \widehat{a}(n) = \beta x_n + (1 - \beta)(\widehat{a}(n-1) + \widehat{b}(n-1)) \\ \widehat{b}(n) = \gamma(\widehat{a}(n) - \widehat{b}(n-1)) + (1 - \gamma)\widehat{b}(n-1) \end{cases} \quad (1.14)$$

La prévision à l'horizon h faite à la date n est donnée par :

$$\widehat{y}_{n,h} = \widehat{a}(n) + \widehat{b}(n)h \quad (1.15)$$

Cette méthode est plus flexible que la méthode du lissage exponentiel double dans la mesure où elle fait intervenir deux constantes ($\beta; \gamma$) au lieu (α).

Cette formule peut être simplifiée au cas où

$$\beta = 1 - \alpha^2 \text{ et } \gamma = \frac{1 - \alpha}{1 + \alpha}$$

Ainsi les formules de mise à jour deviennent :

$$\begin{cases} \widehat{a}(n) = \alpha^2(\widehat{a}(n-1) + \widehat{b}(n-1)) + (1 - \alpha^2)y_n \\ \widehat{b}(n) = \widehat{b}(n-1) + \frac{(1 - \alpha)^2}{1 - \alpha^2}(\widehat{a}(n) - \widehat{a}(n-1) - \widehat{b}(n-1)) \end{cases} \quad (1.16)$$

Séries saisonnières

Modèle additif

On cherche maintenant à ajuster au voisinage de l'instant n une droite d'équation

$$y_t = a + b(t - n) + S_t$$

où S_t est une composante périodique de période T . Les formules récursives de mise à jour sont :

$$\begin{cases} \widehat{a}(n) = \beta(x_n - \widehat{S}_{n-T}) + (1 - \beta)(\widehat{a}(n-1) + \widehat{b}(n-1)) \\ \widehat{b}(n) = \gamma(\widehat{a}(n) - \widehat{a}(n-1)) + (1 - \gamma)\widehat{b}(n-1) \\ \widehat{S}_n = \sigma(x_n - \widehat{a}(n)) + (1 - \sigma)\widehat{S}_{n-T} \end{cases} \quad (1.17)$$

Les prévisions sont de la forme :

$$\begin{cases} \widehat{x}_{n,h} = \widehat{a}(n) + h\widehat{b}(n) + \widehat{S}_{n+h-T} & 1 \leq h \leq T \\ \widehat{x}_{n,h} = \widehat{a}(n) + h\widehat{b}(n) + \widehat{S}_{n+h-2T} & T+1 \leq h \leq 2T \end{cases} \quad (1.18)$$

et ainsi pour $2T \leq h$ dans cette procédure la difficulté de choisir les constante de lissage est encore plus grande pour les autres situations.

Modèle multiplicatif

On ajuste au voisinage de l'instant n une droite d'équation :

$$y_t = (a + b(t - n)) \times S_t$$

où S_t est une composante périodique de période T . Les formules récursives de mise à jour sont :

$$\begin{cases} \widehat{a}(n) = (1 - \beta) \frac{x_t}{S_{n-T}} + \beta(\widehat{a}(n-1) + \widehat{b}(n-1)) \\ \widehat{b}(n) = (1 - \gamma)(\widehat{a}(n) - \widehat{a}(n-1)) + \gamma\widehat{b}(n-1) \\ \widehat{S}_n = (1 - \sigma) \frac{x_n}{\widehat{a}(n)} + \sigma\widehat{S}_{n-T} \end{cases} \quad (1.19)$$

Les prévisions sont de la forme :

$$\begin{cases} \widehat{x}_{n,h} = (\widehat{a}(n) + h\widehat{b}(n)) \times \widehat{S}_{n+h-T} & 1 \leq h \leq T \\ \widehat{x}_{n,h} = (\widehat{a}(n) + h\widehat{b}(n)) \times \widehat{S}_{n+h-2T} & T+1 \leq h \leq 2T \end{cases} \quad (1.20)$$

1.5 Commandes de lissages exponentiel dans R

En R, il existe plusieurs commandes et fonctions pour effectuer des lissages exponentiels. L'une des approches couramment utilisées est basée sur la méthode Holt-Winters, qui combine des composantes de tendance et de saisonnalité. Voici une explication des principales commandes de lissage exponentiel disponibles dans R :

1. **La fonction HoltWinters()** : Cette fonction permet d'appliquer le modèle de Holt-Winters pour le lissage exponentiel. Elle prend en entrée une série temporelle et estime les paramètres nécessaires pour le lissage exponentiel.

Voici les différents paramètres de la fonction `HoltWinters()` :

- **x** : La série temporelle à modéliser, généralement un objet de type `ts`.
 - **alpha** : Le coefficient de lissage pour la composante de niveau (entre 0 et 1).
 - **beta** : Le coefficient de lissage pour la composante de tendance (entre 0 et 1).
 - **gamma** : Le coefficient de lissage pour la composante de saisonnalité (entre 0 et 1).
 - **seasonal** : La période saisonnière de la série temporelle. Par exemple, pour des données mensuelles, vous pouvez spécifier `seasonal = 12`.
 - **initial** : Les valeurs initiales pour les composantes de niveau, de tendance et de saisonnalité. Si vous ne spécifiez pas cette valeur, le modèle tentera de les estimer automatiquement.
 - **optim.control** : Les options de contrôle pour l'optimisation du modèle. Par exemple, vous pouvez spécifier le critère d'optimisation, le nombre d'itérations, etc.
2. **La fonction ets()** : Cette fonction est utilisée pour effectuer une modélisation et un lissage exponentiel des données temporelles en utilisant le modèle ETS (Error, Trend, Seasonality). Le modèle ETS est plus flexible que le modèle de Holt-Winters, car il peut gérer différentes configurations de composantes de tendance et de saisonnalité.

Voici les différents paramètres de la fonction `ets()` :

- **y** : La série temporelle à modéliser, généralement un objet de type `'ts'`.
- **model** : Le type de modèle à utiliser, qui peut prendre les valeurs "ZZZ", "AAN", "MAM", etc. Chaque lettre (A, M, N) représente la présence ou l'absence d'une composante spécifique (Additive, Multiplicative, None).

- **damped** : Un booléen indiquant si le modèle doit être amorti (damped) ou non.

conclusion

l'étude des séries chronologiques offre un arsenal important d'analyse et de modélisation de phénomène complexes. Ce qui est présenté dans ce chapitre n'est qu'une infime partie de cet arsenal, et le chapitre suivant portera sur l'application du lissage pour modéliser un phénomène complexe qui est le séchage d'aliments en se base sur des données expérimentales.

Machine Learning

Introduction

Dans ce chapitre, nous allons explorer le domaine passionnant du Machine Learning (apprentissage automatique). Le Machine Learning [10] est une branche de l'intelligence artificielle qui permet aux machines d'apprendre à partir de données et d'améliorer leurs performances sans être explicitement programmées. Il offre des outils et des techniques puissantes pour résoudre des problèmes complexes, prendre des décisions basées sur les données et découvrir des modèles cachés dans les ensembles de données.

2.1 Définition et Histoire

L'apprentissage automatique ou apprentissage automatique (ML) est une sous-section du domaine intelligence artificielle ou intelligence artificielle en informatique, visant à enseigner aux machines effectuer des tâches sans programmation explicite, et en utilisant l'un des Nous l'appellerons l'algorithme pour le modèle et les données. Le concept d'intelligence artificielle a émergé lors d'une conférence dans les années 1950 Réunissant un grand nombre de scientifiques reconnus dans les domaines de l'informatique et des mathématiques, leur Alan Turing. Ce scientifique de génie a également prédit le

développement de l'apprentissage automatique comme nous le savons. Le ML a refait surface entre les années 70 et 80, et l'idée derrière le concept était de créer Algorithmes avec la capacité d'accumuler de l'expérience et des connaissances vingt-deux A la fin des années 1980, on assiste au retour des réseaux de neurones et à leur réinvention algorithme de rétropropagation, mais sans succès car le domaine Finalement, il sera jeté en raison du manque de capacité de calcul et de stockage suffisante. Il a fallu attendre le milieu des années 2000 pour le grand retour des réseaux de neurones pour le deep learning.

2.2 Importance et applications du Machine Learning

Le Machine Learning (apprentissage automatique) est une branche de l'intelligence artificielle qui permet aux ordinateurs d'apprendre à partir de données et de prendre des décisions ou de faire des prédictions sans être programmés. Médecine et santé : Le Machine Learning est utilisé dans de nombreux domaines de la médecine et de la santé, tels que le diagnostic médical, la découverte de médicaments, l'analyse d'images médicales, la prédiction de l'évolution des maladies, etc. Voici quelques-unes des raisons pour lesquelles le Machine Learning est important et ses applications dans divers domaines : Prédiction et prévision : Le Machine Learning est utilisé pour effectuer des prédictions et des escomptes dans de nombreux domaines, tels que la finance, les sciences sociales, la météorologie, le marketing, la santé, etc.

2.3 Méthodes d'apprentissage

Il existe de nombreux sous-domaines ou sous-sections. On note les 3 principales, sa dernière Selon le type de problème que nous voulons traiter, nous constatons que :

- 1• apprentissage supervisé
- 2• Apprentissage Non Supervisé

- 3• Apprentissage par Renforcement.
- 4• Apprentissage Semi-Supervisé.

2.3.1 Apprentissage supervisé

L'apprentissage supervisé est une méthode d'apprentissage automatique qui se caractérise par la création d'algorithmes qui apprennent des fonctions prédictives. Ceci est possible car formé à partir d'exemples annotés, qui incluent un ensemble de variables d'entrée, avec leurs variables de sortie respectives. Ce processus de formation est répété jusqu'à ce que des performances satisfaisantes soient obtenues, à chaque itération, la machine crée de nombreuses règles qui relient les variables d'entrée aux variables de sortie. Ce processus permet au modèle d'apprendre à partir des données et d'ap-

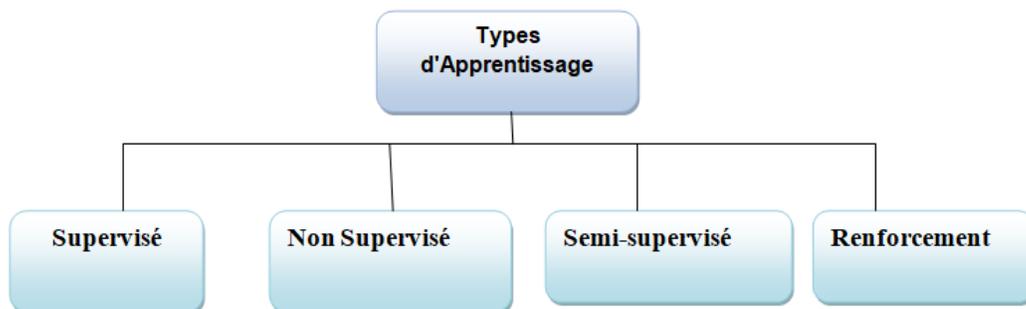


FIGURE 2.1 – Type d'apprentissage en ML

pliquer Lorsque la valeur d'entrée est une règle qui prédit avec précision la valeur de sortie donner.

L'apprentissage supervisé peut être divisé en deux sous-catégories : classification et régression. Nous nous concentrons maintenant sur ce dernier. Plus Informations sur les modèles de régression, la classification et l'utilisation de ceux-ci Compétences Linedata

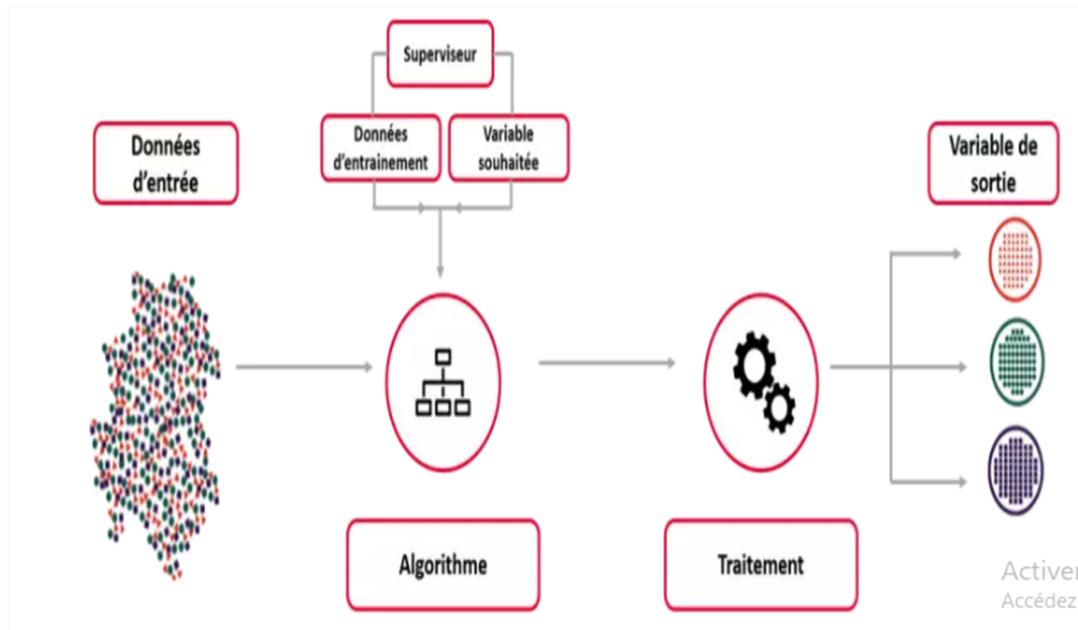


FIGURE 2.2 – Apprentissage supervisé

Régression vs Classification dans Machine Learning

Les algorithmes de régression et de classification sont des outils précieux dans le domaine de l'apprentissage automatique. Ils sont tous deux utilisés dans des problèmes de prédiction, mais avec une approche différente en fonction du type de données et des objectifs spécifiques. Lorsqu'il s'agit d'algorithmes de régression, ceux-ci sont utilisés pour prédire des valeurs continues telles que le prix d'un bien, un salaire, l'âge d'une personne, et bien d'autres. Ils permettent d'obtenir des estimations quantitatives en se basant sur des données étiquetées. D'un autre côté, les algorithmes de classification sont utilisés pour prédire ou classer des valeurs discrètes. Par exemple, ils peuvent être employés pour différencier entre un homme et une femme, déterminer si un e-mail est un spam ou non, ou encore pour distinguer entre des options binaires telles que vrai ou faux. Ces algorithmes permettent de prendre des décisions qualitatives en se basant sur des caractéristiques spécifiques des données. En résumé, la différence fondamentale entre les algorithmes de régression et de classification réside

dans leur capacité à prédire des valeurs continues ou discrètes respectivement. Chacun de ces outils est adapté à des problèmes spécifiques et offre une approche unique pour l'apprentissage automatique, etc. Considérez le schéma ci-dessous :

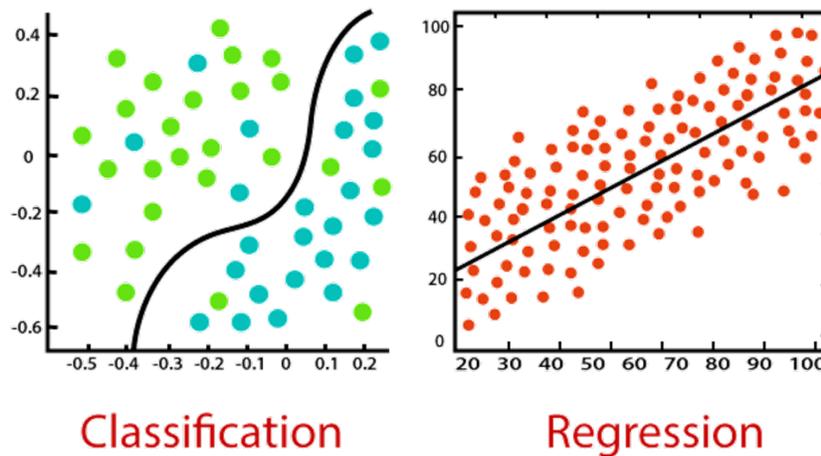


FIGURE 2.3 – Régression vs Classification.

Classification :

La classification est une tâche d'apprentissage supervisé où les sorties sont étiquetées avec des valeurs définies (valeurs discrètes), appelées classes ou catégories. On peut avoir des classifications binaires ou multiclasse. Dans le cas de la classification binaire, le modèle prédit soit 0 soit 1, vrai ou faux. En ce qui concerne la classification multiclasse, le modèle prédit plusieurs classes. Peu importe le nombre de classes, l'objectif reste le même : prédire si les valeurs discrètes appartiennent à une classe particulière

Exemple 2.1 *Le meilleur exemple pour comprendre le problème de classification est la détection du Spam par courrier électronique. Le modèle est entraîné sur la base de millions d'e-mails sur différents paramètres, et chaque fois qu'il reçoit un nouvel e-mail, il identifie si l'e-mail est du spam ou non. Si l'e-mail est un spam, il est déplacé vers le dossier Spam.*

Types d'Algorithmes de Classification ML

Les algorithmes de classification peuvent être divisés en types suivants :

- 1• Régression logistique
- 2• K-Voisins les plus proches
- 3• Machines Vectorielles de Soutien
- 4• SVM du noyau
- 5• Bayes Naïfs
- 6• Classification de l'Arbre décisionnel
- 7• Classification aléatoire des Forêts

Régression :

La régression est une tâche d'apprentissage supervisé où l'objectif est de prédire une valeur continue. L'idée ici est de trouver une estimation aussi proche que possible de la vraie valeur de sortie en utilisant notre modèle. Ensuite, nous évaluons la performance en calculant l'erreur. Plus l'erreur est faible, plus notre modèle de régression est précis

Exemple 2.2 *Supposons que nous voulions faire des prévisions météorologiques, donc pour cela, nous utiliserons l'algorithme de régression. Dans la prévision météorologique, le modèle est entraîné sur les données passées et, une fois l'entraînement terminé, il peut facilement prédire le temps qu'il fera pour les jours à venir.*

Types d'algorithme de régression

- 1• Régression linéaire simple.
- 2• Régression linéaire multiple.
- 3• Régression polynomiale.

- 4• Prise en charge de la régression vectorielle.
- 5• Régression d'arbre de décision.
- 6• Régression forestière aléatoire.

Différence entre régression et classification

Algorithme de Régression	Algorithme de Classification
En régression, la variable de sortie doit être de nature continue ou de valeur réelle.	En classification, la variable de sortie doit être une valeur discrète.
La tâche de l'algorithme de régression est de mapper la valeur d'entrée (x) avec la variable de sortie continue(y).	La tâche de l'algorithme de classification est de mapper la valeur d'entrée(x) avec la variable de sortie discrète(y).
Les algorithmes de régression sont utilisés avec des données continues.	Les algorithmes de classification sont utilisés avec des données discrètes.
Dans la régression, nous essayons de trouver la meilleure ligne , d'ajustement qui peut prédire la sortie avec plus de précision.	En classification, nous essayons de trouver la limite de décision, qui peut diviser l'ensemble de données en différentes classes.
Les algorithmes de régression peuvent être utilisés pour résoudre les problèmes de régression tels que la prévision météorologique, la prévision du prix de l'immobilier, etc.	Les algorithmes de classification peuvent être utilisés pour résoudre des problèmes de classification tels que l'identification des spams, la reconnaissance vocale, l'identification des cellules cancéreuses, etc.
L'algorithme de régression peut être divisé en régression linéaire et Non linéaire.	Les algorithmes de classification peuvent être divisés en Classificateur binaire et Classificateur multiclasse.

TABLE 2.1 – Différence entre régression et classification

2.3.2 non supervisé machine learning

L'apprentissage non supervisé, également connu sous le nom d'apprentissage non étiqueté, est un type d'algorithme d'apprentissage automatique utilisé pour tirer des conclusions à partir de ensembles de données sans réponses étiquetées. Ce type d'apprentissage est principalement utilisé lorsque l'on souhaite analyser un ensemble de données non étiquetées

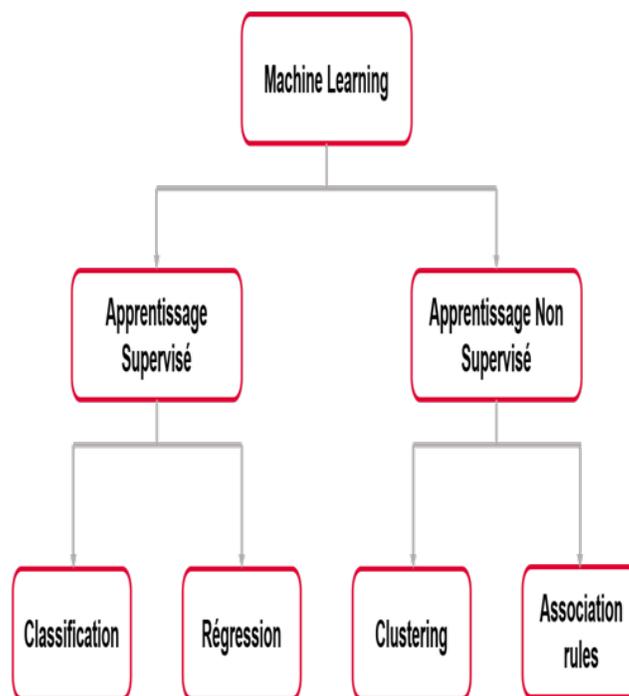


FIGURE 2.4 – l'apprentissage automatique

Deux types de problèmes d'apprentissage non supervisé

On peut considérer l'apprentissage non supervisé comme étant séparé en deux catégories : le clustering et l'association.

Le clustering :

Le clustering est une technique d'apprentissage non supervisé qui regroupe des données non étiquetées en fonction de leurs similarités et de leurs différences. L'objectif est de former des groupes (ou des clusters) où les points au sein d'un même groupe sont le plus similaires possible, tandis que les points dans des groupes différents sont les plus différents possible. Pour cela, une analyse de cluster identifie les caractéristiques présentes ou absentes dans les données et regroupe les points en conséquence. Parmi les méthodes de clustering, on trouve la méthode des k-moyennes (ou k-means).

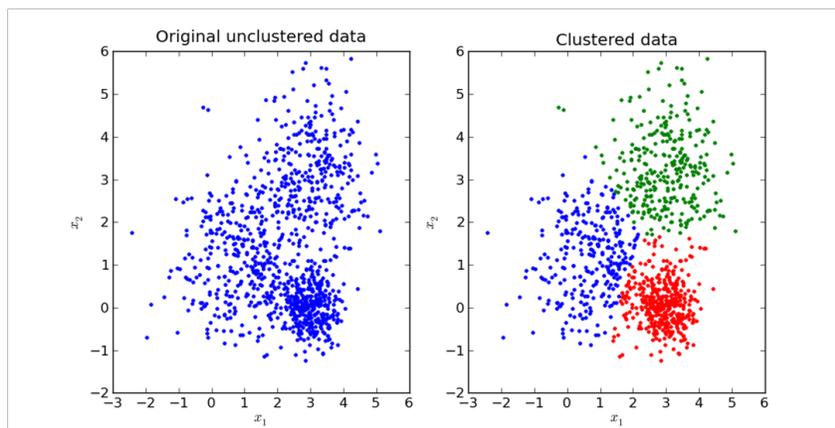


FIGURE 2.5 – les types de problèmes d'apprentissage non supervisé

La classification

La classification hiérarchique, également connue sous le nom de clustering hiérarchique, et la classification probabilistique, ou clustering probabilistique, sont deux approches de l'apprentissage non supervisé.

D'un autre côté, les règles d'association représentent une autre forme d'apprentissage non supervisé qui identifie des relations entre les points de données. En d'autres termes, ces algorithmes découvrent des associations entre les points qui se retrouvent ensemble dans les données. Ces méthodes sont souvent utilisées dans l'analyse du panier d'achat, permettant aux entreprises de comprendre les liens entre l'achat de dif-

férents produits. Par exemple, cela peut révéler des associations telles que : "Les personnes qui achètent le produit X ont également tendance à acheter le produit Y". Parmi les algorithmes d'association fréquemment utilisés, on trouve l'algorithme Apriori, l'algorithme Eclat et l'algorithme FP-growth. **Clustering** : Le clustering permet de

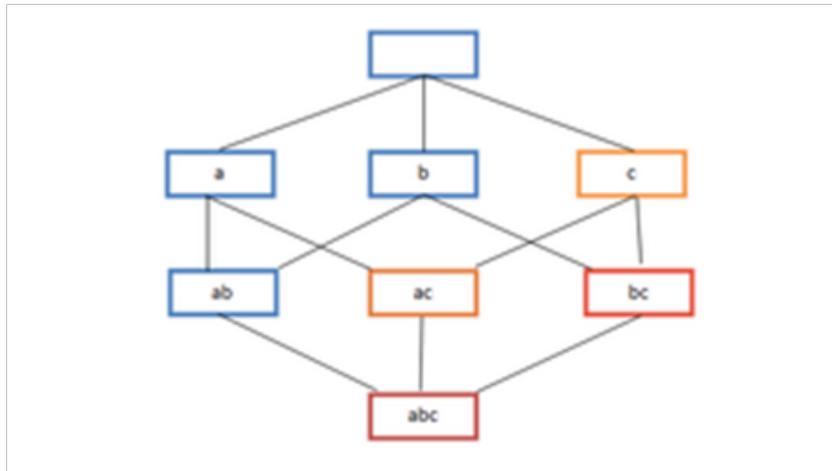


FIGURE 2.6

regrouper les données d'entrée en ensembles ou groupes de données présentant des caractéristiques similaires, et de les attribuer à un cluster. Contrairement à la classification dans l'apprentissage supervisé, ces différents clusters ou groupes ne sont pas connus à l'avance. C'est l'algorithme lui-même qui va séparer les données en un nombre approprié de clusters. Comme on peut le voir dans l'exemple illustré dans la Figure II.3, les points du jeu de données ont été divisés en groupes identifiables par les couleurs rouge, vert et bleu.

Différence entre apprentissage supervisé et non supervisé :

La majeure différence qu'on peut trouver entre ces 2 types d'apprentissages est la disposition des données d'entrée, mais il existe tout de même d'autres différences qu'on peut trouver dans le tableau ci-dessous. Ce tableau résume la différence entre apprentissage supervisé et non supervisé :

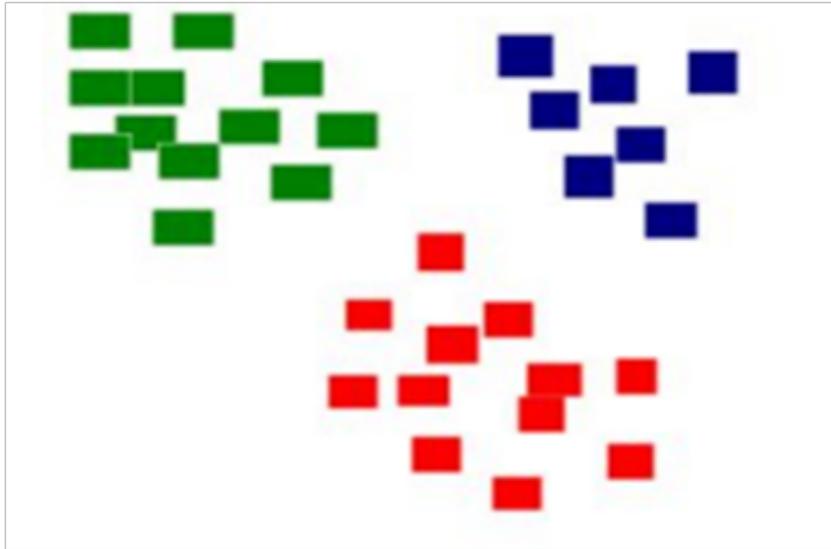


FIGURE 2.7 – Clustering.

	Supervised Learning	Unsupervised Learning
Les données d'entrée	Utilise des données connues et étiquetées en tant qu'entrée	Utilise des données inconnues en tant qu'entrée Non étiquetées
Nombre de Classes	Connues	Inconnues
Précision du résultat	Résultats précis et fiables	précision et fiabilité Modéré

TABLE 2.2 – Différence entre apprentissage supervisé et non supervisé

2.3.3 Apprentissage semi-supervisé

Nous avons précédemment abordé l'apprentissage supervisé et non supervisé, qui diffèrent principalement par l'étiquetage ou non des données, ainsi que par les méthodes appropriées utilisées pour traiter ces données. L'apprentissage semi-supervisé regroupe ces deux principes. Il utilise un ensemble réduit de données étiquetées ainsi qu'un autre ensemble de données non étiquetées du même type. L'avantage de ce type d'apprentissage réside principalement dans le fait que le processus d'étiquetage des données est souvent long et coûteux. Paradoxalement, le non-étiquetage devient bénéfique pour le processus d'apprentissage, et la construction du modèle est moins coûteuse.

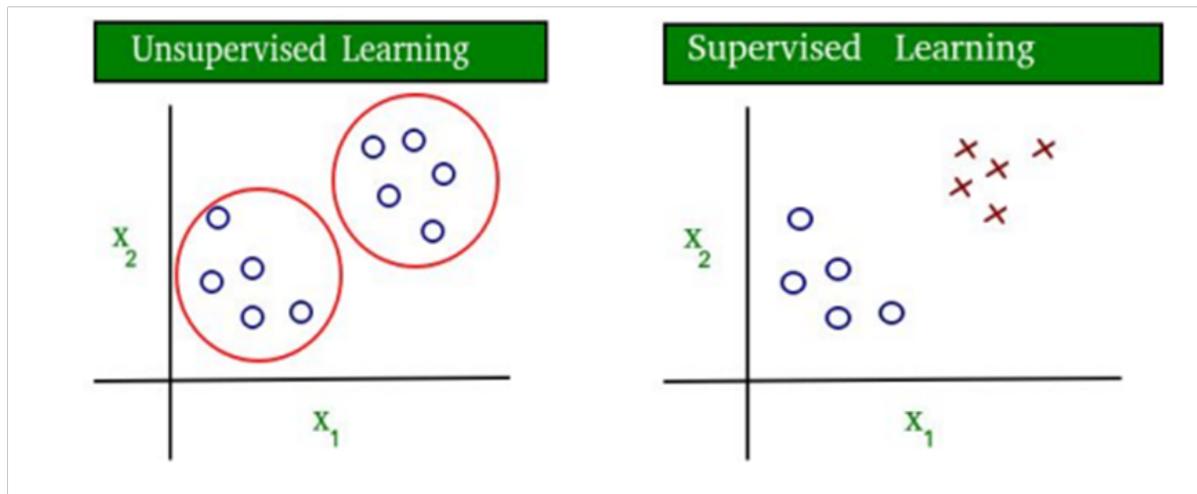


FIGURE 2.8 – Supervised vs Unsupervised Learning

2.3.4 Apprentissage par Renforcement

L'entraînement par renforcement est une technique dans laquelle un agent est immergé dans un environnement où il interagit dans le but d'apprendre. L'agent est équipé de capteurs pour percevoir les informations de son environnement, ainsi que d'actionneurs qui lui permettent d'agir dans cet environnement. L'agent observe les entrées, prend des décisions et effectue des actions. Une fois l'action réalisée, l'agent reçoit des récompenses en conséquence, ce qui renforce le modèle en stockant les informations dans une base de données. Les récompenses peuvent être positives, négatives ou nulles en fonction des actions effectuées.

Exemple 2.3 *AlphaGo, robot domestique, ou encors IA dans les jeux vidéos.*

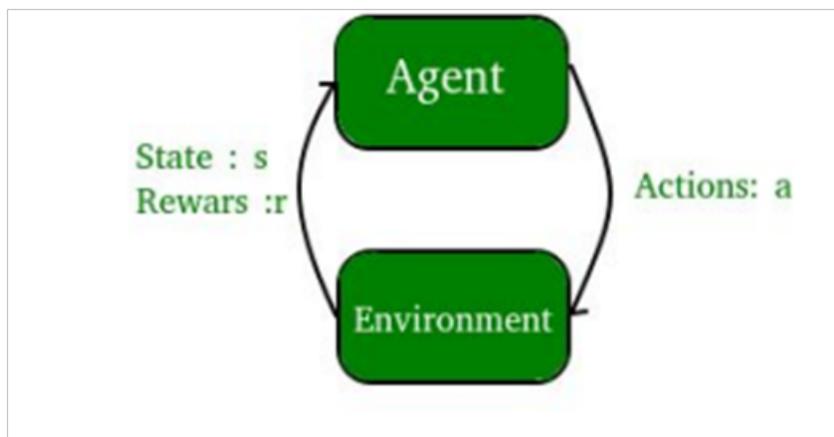


FIGURE 2.9 – Apprentissage par renforcement concept

Prévision par lissage exponentiel et lissage-machine learning

Introduction

ce chapitre comportera deux parties importantes. Dans la première partie, on effectuera un lissage exponentiel aux séries séchage de la citrouille¹. Dans la deuxième partie on proposera un programme R dans lequel on combinera la méthode de lissage exponentiel avec l'une des techniques du machine learning dans le but de voir la possibilité d'améliorer la qualité des prévisions en prenant comme objectif la réduction de l'erreur de prévision.

Le logiciel utilisé "R" Comme mentionné précédemment, notre analyse des données sera effectuée à l'aide du logiciel "R". Ce logiciel offre un large éventail de fonctionnalités dans le domaine de l'analyse statistique et de la programmation, ainsi que pour la réalisation de prévisions, de simulations et de présentations analytiques des données. En plus de sa simplicité d'utilisation, "R" est l'outil idéal pour les étudiants, les professeurs, les chercheurs ou les analystes travaillant avec des séries temporelles, des

1. Ces données sont obtenus au niveau du laboratoire LBBS de l'université de Béjaia, dans un projet d'étude de la technique de séchage d'aliments sur micr-ondes

données de panel ou des données longitudinales.

Avec "R", il est facile de manipuler efficacement les données, d'effectuer des analyses statistiques et économétriques, de générer des prévisions ou des simulations, et de produire des graphiques de qualité pouvant être intégrés dans des articles ou des rapports. Les analyses de séries temporelles et de régression linéaire sont également disponibles dans ce logiciel.

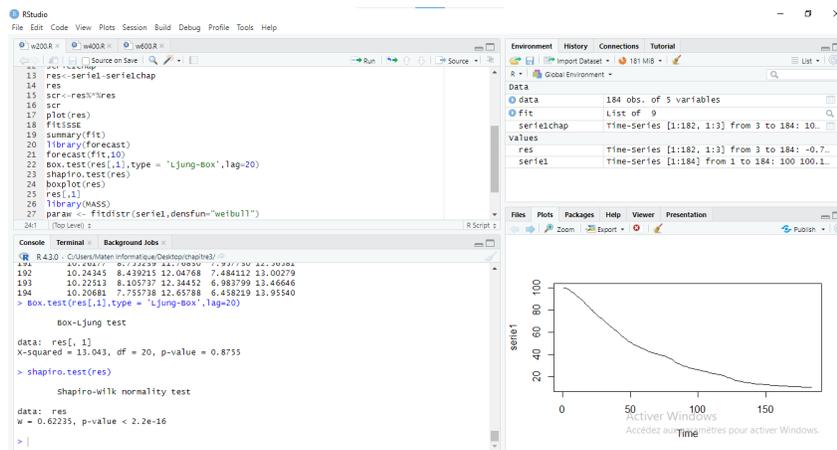


FIGURE 3.1 – Environnement Générale de " R et RStudio"

3.1 Modél de lissage exponentiel

3.1.1 Modélisation de la série 200w

Graphique de la série :

Après introduction des données relatives au séchage de la citrouille (on dispose d'un tableau de plusieurs séries résultats du séchages sur plusieurs périodes de temps et suivants différentes puissances du micro-ondes (200W, 400w, 600w et 800w)), on extrait la première colonne du tableua représentant la chronique "200w". le graphe de cette série sous R est comme suit :

Le graphe de la figure (3.2) montre que la sèrie évolue de manière décroissante en fonction de temps. On souligne aussi que cette décroissance est de type gèométrique ou

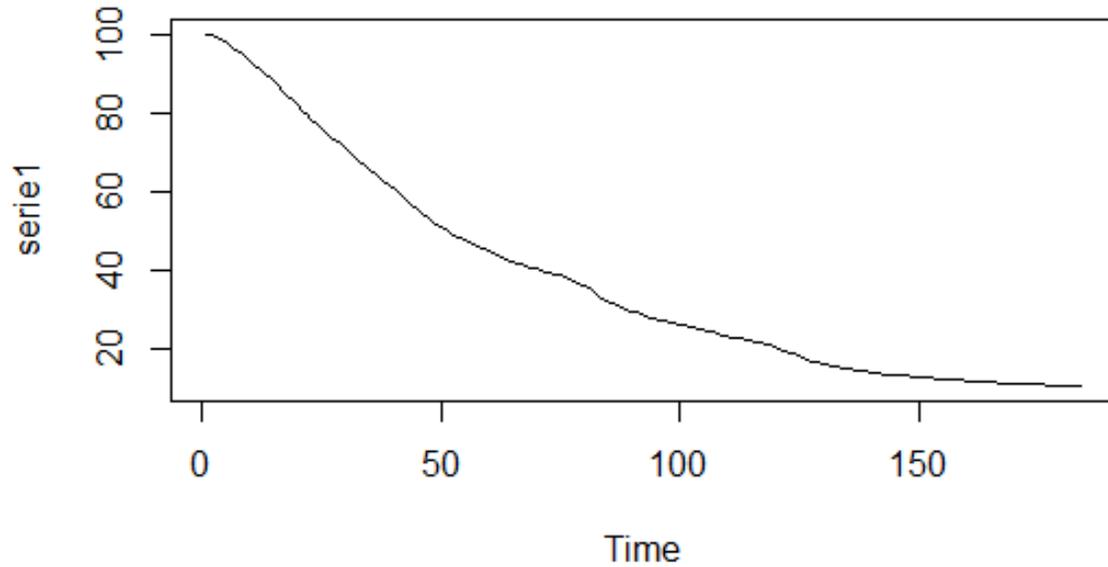


FIGURE 3.2 – Le graphe de la sèrie de puissance 200 W

exponentiel. On conclut alors que la sèrie présente une tendance non linéaire (la sèrie est non stationnaire). Par contre la sèrie ne présente aucune saisonnalité : la méthode la mieux adaptée pour l'étude de cette sèrie est le modèle de Lissage exponentiel Double et celui de Holt - Winters non saisonnier. L'estimation de ce modèle sur le logiciel " R", nous donne les résultats suivants :

Le modèle obtenu pour la sèrie 200 W est alors :

$$\begin{cases} \hat{a} = 2a - b = 20.79 \\ \hat{b} = \frac{1 - \alpha}{\alpha} [a - b] = 1.14 \\ SSE = 3.70 \end{cases} \quad (3.1)$$

```

Holtwinters(x = serie1, gamma = FALSE)

Smoothing parameters:
alpha: 0.8631951
beta : 0.7534253
gamma: FALSE

Coefficients:
      [,1]
a 10.39001449
b -0.01832066

```

FIGURE 3.3 – Le résultat LED de 200 W.

test sur les résidus

Test de Dickey-Fuller augmenté (bruit blanc) : Ce test oppose l'hypothèse nulle " la série est non stationnaire contre l'hypothèse alternative "la série est stationnaire. Pour notre série, on tape sur R : `adf.test(prev$residuals)`

```

data: prev$residuals
Dickey-Fuller = -4.1926, Lag order = 5,
p-value = 0.01
alternative hypothesis: stationary

```

FIGURE 3.4 – Resultats ADF Residus

la $p - \text{value} < 0.05$ on rejette l'hypothèse de non stationnarité, les résidus des prévisions forment bien un bruit blanc.

Test de Shapiro-Wilk : le test de Shapiro–Wilk teste l'hypothèse nulle selon laquelle un échantillon est issu d'une population normalement distribuée.

Ici on teste la normalité des erreurs de prédiction de la série citrouille 200w par ce test en appliquant la commande R : `shapiro.test(200w)`

D'après le test de Shapiro-Wilk, on obtient les résultats suivantes :

$$W = 0.62235 \quad p - \text{value} = 2.2e - 16$$

Les résultats fournis, montre que les résidus sont un bruit blanc gaussien. Le Mdèle obtenu est donc retenu pour les prévisions.

3.1.2 Résultats du lissage des séries de puissances (400W.600W.800W)

On procède de la même manière que la série 200 W. Les modèles validés pour chaque série sont résumés dans le tableau suivant :

La série	lissage exponentiel de Holt Winters
400	$\hat{a}=2a-b=19.99$ $\hat{b} = \frac{1-\alpha}{\alpha}[a-b] = 0$ Résidu : SSE=10.42
600	$\hat{a}=2a-b=17.54$ $\hat{b} = \frac{1-\alpha}{\alpha}[a-b] = 0$ Résidu : SSE=17.60
800	$\hat{a}=2a-b=17.242$ $\hat{b} = \frac{1-\alpha}{\alpha}[a-b] = 0$ Résidu : SSE=12.88

TABLE 3.1 – Les résultats de HLW de chaque sèrie

Examen des Résidu

3.2 Modélisation par lissage combiné avec le machine learning

3.2.1 Modélisation de la série 200w

3.3 Modèle Lissage-Machine Learning

L'idée qu'on propose dans cette partie es comme suit : En se basant sur la technique de machine learning on divisera chaque série temporelle en deux série dite "train" et "test".

La première est utilisée pour estimer le paramètres d'un modèle de lissage exponentiel, puis en entraîne ce modèle avec la commande *train* de la bibliothèque *caret* de R. une fois l'entraînement terminé, on essaye d'améliorer la qualité de la prévision en combinant le résultats de la fonction *train* avec l'un les données dans le vecteur "test", à l'aide d'un algorithme du machine learning supervisé (ici dans notre programme on a appliqué la regression) avec l'objectif de réduire l'erreur de prévision, pour en avoir enfin les prévisions de lissage combiné avec le machine learning.

Le code du programme développé par nos soins est le suivant :

```
library(forecast)
library(caret)
library(Rcmdr)
data <- readXL("C :\\Users\\Dell\\Desktop\\ARIMA\\fichier2\\citrouille.xlsx")
y <- data$X200.W
*****
train_index <- 1 :(length(y) - 10)
train <- y[train_index]
test <- y[-train_index]
*****
```

```

esm_model <- ets(train)
*****
esm_predictions <- forecast(esm_model, h = length(test))$mean
*****

train_data <- data.frame(y = train, lag1 = c(NA, head(train, -1)), lag2 = c(NA, NA,
head(train, -2)))

train_data_complete <- na.omit(train_data)
missing_rows <- which(!complete.cases(train_data))
train_data_clean <- train_data[complete.cases(train_data), ]

lm_model <- train(y ~ ., data = train_data_clean, method = "lm")
*****

test_data <- data.frame(lag1 = c(tail(test, -1), NA), lag2 = c(tail(test, -2), NA, NA))
test_data_complete <- na.omit(test_data)
missing_rowss <- which(!complete.cases(test_data))
test_data_clean <- test_data[complete.cases(test_data), ]
*****

lm_predictions <- predict(lm_model, newdata = test_data_clean)
combined_predictions <- (esm_predictions + lm_predictions) / 2
*****

sse <- mean((combined_predictions - test)^2)

print(paste("SSE du modèle de lissage exponentiel :", mean((esm_predictions - test)^2)))
print(paste("SSE du modèle de régression linéaire :", mean((lm_predictions - test)^2)))
print(paste("SSE du modèle combiné :", sse))

```

3.3.1 Résultats et discussion

Les résultats des résidus obtenus pour les différentes séries est comme suit :

On voit clairement que la somme des carrés de résidus moyenne est minimale avec le modèle de lissage combiné avec le machine learning et ceci sur les quatre séries

série	SSE Lissage	SSE Lissage combiné ML.
200W	0.049015	0.00628
400W	0.030413	0.01857
600W	0.056025	0.018125
800W	5.537663	1.10149

TABLE 3.2 – Somme des carrés des résidus moyenne dans chaque modèle

utilisées dans ce mémoire ce qui implique que la technique de regression offerte par le machine learning supervisé améliore la qualité des prévisions en minimisant les erreurs de prévisions.

conclusion

Le machine learning est un outil très important dans l'amélioration de la qualité des prévisions par lissage exponentiel. la technique de regression linéaire du machine learning supervisé appliquée dans ce mémoire montre clairement ce résultats.

Conclusion Générale

En général, l'application du machine learning au lissage exponentiel peut offrir plusieurs avantages pour les prévisions de séries temporelles, y compris le séchage d'aliments par micro-ondes. Le lissage exponentiel est une méthode couramment utilisée pour estimer les tendances et les modèles de variation dans les données temporelles. L'intégration du machine learning dans ce processus peut permettre d'améliorer la précision des prévisions en utilisant des algorithmes d'apprentissage automatique pour identifier des motifs plus complexes et prendre en compte divers facteurs qui influencent le séchage de la citrouille par micro-ondes.

L'application du machine learning dans ce contexte peut également permettre d'ajuster plus efficacement les paramètres du modèle de lissage exponentiel en fonction des caractéristiques spécifiques des données de séchage de la citrouille par micro-ondes. Cela peut inclure des variables telles que la température, le temps de séchage, la teneur en eau initiale, etc. En utilisant des techniques d'apprentissage automatique, il est possible d'identifier les relations entre ces variables et d'ajuster les paramètres du modèle en conséquence, ce qui peut conduire à des prévisions plus précises.

Cependant, il est important de noter que l'application du machine learning à l'amélioration des prévisions par lissage exponentiel nécessite des données de haute qualité et une modélisation appropriée. Il est essentiel d'avoir un ensemble de données suffisamment grand et représentatif, ainsi qu'une sélection appropriée des algorithmes

d'apprentissage automatique et des techniques de validation pour obtenir des résultats fiables.

En conclusion, l'application du machine learning au lissage exponentiel pour améliorer les prévisions de séchage de la citrouille par micro-ondes peut offrir des avantages significatifs en termes de précision des prévisions. Cependant, cela nécessite une approche rigoureuse dans la collecte et la préparation des données, ainsi que dans la sélection et la mise en oeuvre des techniques d'apprentissage automatique. L'utilisation de ces méthodes peut contribuer à optimiser les processus de séchage de la citrouille par micro-ondes et à prendre des décisions plus éclairées dans ce domaine.

comme perspective de travail, on peu envisager d'appliquer d'autres techniques de machine learning sur les séries utilisées à fin de comparer la meilleure technique à adopter.

On peut même considérer d'étendre l'étude vers des série comportant différentes composantes avec de graphique plus compliqués pour en tirer une conclusion générale.

qdhqhdhd

[15],[12][6][11][3][1][8][4] [5][2][10][14][7][13][9]

Annexes

Annexe1 : Enête des données relatives aux cinétique de séchage de la citrouille par micro ondes

Le temps	200W	400W	600W	800W
0	100	100	100	100
30	100.13	99.0366667	97.9133333	96.7066667
60	99.52	97.3566667	93.58	91.7833333
90	98.8933333	95.7033333	88.2466667	86.4533333
120	98.1333333	93.7233333	82.5833333	81.7366667
150	97.1866667	91.5433333	76.7066667	77.59
180	96.3366667	89.1266667	71.0066667	71.99
210	95.2633333	87.5866667	67.1533333	67.3933333
240	94.2366667	86.2633333	64.2033333	63.8066667
270	93.3533333	83.61	60.02	61.03
300	92.3666667	80.3233333	56.9533333	57.9466667
330	91.3766667	78.51	54.22	55.7866667
360	89.98	76.6033333	50.3766667	53.5366667
390	89.14	74.5466667	47.8133333	50.4366667
420	88.0566667	72.2366667	45.6833333	48.53
450	86.9766667	70.43	43.68	46.7366667
480	85.53	68.36	41.96	44.7433333

Annexe2 : Résultats du lissage R des séries 400W, 600W et 800W

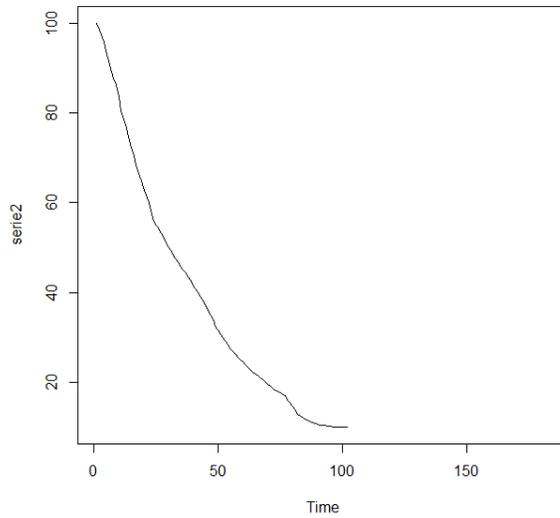


FIGURE 3.5 – Graphe de la série originale de puissance 400W

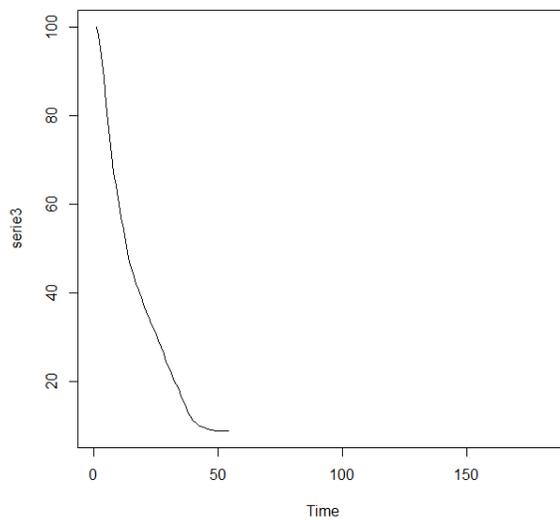


FIGURE 3.6 – Graphe de la série originale de puissance 600W

3.4. ANNEXE3 :RÉSULTAT DU HOLT WINTER NON SAISONNIER DES SÉRIES SUR "R"47

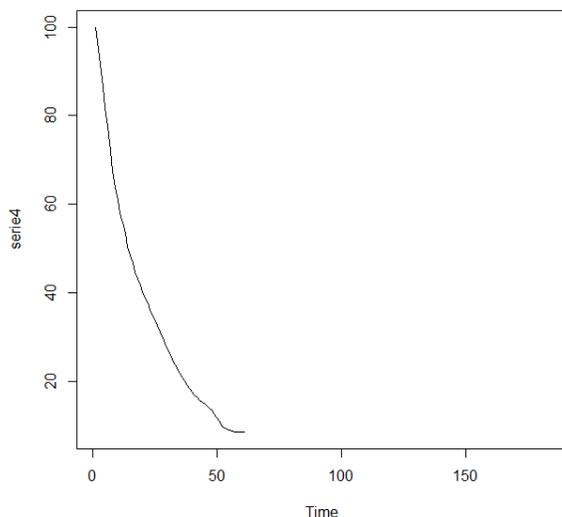


FIGURE 3.7 – Graphe de la série originale de puissance 800W

3.4 Annexe3 :Résultat du Holt Winter non saisonnier des séries sur "R"

```
Holtwinters(x = na.omit (serie2), gamma = FALSE)
Smoothing paramètres:
Alpha: 1
Beta : 0.467313
Gamma: FALSE

Coefficients:
      [,1]
a 9.99000000
b -0.01041923
```

FIGURE 3.8 – Résultat de LHW de 400W

3.4. ANNEXE3 :RÉSULTAT DU HOLT WINTER NON SAISONNIER DES SÉRIES SUR "R"48

```

Call:
Holtwinters(x = serie1, gamma = FALSE)

Smoothing parameters:
alpha: 0.8631951
beta : 0.7534253
gamma: FALSE

Coefficients:
      [,1]
a 10.39001449
b -0.01832066
    
```

FIGURE 3.9 – Résultat de HLW de 600w

```

> Smoothing parameters:
Alpha: 1
Beta : 1
Gamma: FALSE

Coefficients:
      [,1]
a 8.773333
b 0.000000
    
```

FIGURE 3.10 – Résultat de LHW de 800W

La puissance	Alpha	Betta
200w	0.86	0.075
400w	1	0.46
600w	0.56	0.75
800w	1	1

TABLE 3.3 – Les résultats des paramètreras HLW de chaque puissance

Bibliographie

- [1] Djaout A. *Modélisations empirique des séries chronologiques*. Dunod, 20-04-2011.
- [2] A. Bornancin Plantier. *Conception de modèles de prévision des crues éclair par apprentissage artificiel*. PhD thesis, 2013. Thèse de doctorat dirigée par Johannet, Anne Informatique Paris 6 2013.
- [3] L. Bouziane and R. Mouelhi. *Méthode de prévision*. Support de cours, 2006/2007.
- [4] R. Garnier. Machine learning sur les séries temporelles et applications à la prévision des ventes pour l'e-commerce. modélisation et simulation.
- [5] P. C. Goulet, M. Leroux, D. Stevanovic, and S. Surprenant. Prévision de l'activité économique au québec et au canada à l'aide des méthodes machine learning. Technical report.
- [6] J. Julien. *Introduction au Séries Temporelles*. Dunod, 2014.
- [7] Hendrik Blockeel Kristian Kersting and Siegfried Nijssen Filip Železný. *Machine learning and knowledge discovery in databases*. Springer, 2013.
- [8] Kh. Mecheri. Application de la méthodologie de box-jenkins sur la série du nombre des séances d'hémodialyse, mémoire de master biochimie, université de béjaia, 2016.
- [9] G. Pardoux and B. Goldfarb. *Prévision à court terme : méthodes de lissage exponentiel*. 2013.

- [10] K. Phasinam, A.K. Singh, T. Singh, and M.K. Sharma. *Fundamental of Machine Learning*. Blue Rose Publishers, 2022.
- [11] B. Régis and Terraza V. *Analyse des séries temporelles - 5e édition*. Dunod, Avril 2022.
- [12] A. Saf and S. Reddam. Technologies de séchage etude de cas séchage de deux matrices végétales par étuve, mémoire de master, université de béjaia, juin 2018.
- [13] L. SEDDATI. Séries chronologiques.
- [14] V. Stan and G. Saporta. Une étude comparative de logiciels de prévision automatique de séries chronologiques. In *37èmes Journées de Statistique*, 2005.
- [15] M.C. Viano and A. Philippe. *Maîtrise d'économétrie : Cours de Séries Temporelles*. 1999 - 2004.