

République Algérienne Démocratique et Populaire
Université Abou Bakr Belkaid– Tlemcen
Faculté des Sciences
Département d'informatique

Mémoire de fin d'études pour l'obtention du diplôme de
Master en Informatique

Option : Génie logiciel(G.L)

Thème

**Développement des modèles de prévision
de la consommation et de la production
éolienne et photovoltaïque**

Réalisé par :

MALIKA BERRAHOU

Présenté le 23 Septembre 2023 devant le jury composé de MM.

M. MOURTADA BENAZZOUZ	Président (Université de Tlemcen)
M. SALIM ZIANI CHERIF	Examineur (Université de Tlemcen)
M. HOUCINE MATAALLAH	Encadrant (Université de Tlemcen)
M. SEIF-EDDINE BENKABOU	Encadrant (Université de Poitiers)
M. EMMANUEL GROLLEAU	Co-encadrant (Université de Poitiers)

Année Universitaire : 2022 - 2023

Je dédie ce modeste travail à : Ma mère et mon oncle, et je souhaite que mon père repose en paix. Vos sacrifices et votre soutien inconditionnel ont fait de moi ce que je suis aujourd'hui. «Merci infiniment »

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

« قُلْ لَنْ يُصِيبَنَا إِلَّا مَا كَتَبَ اللَّهُ لَنَا هُوَ مَوْلَانَا وَعَلَى اللَّهِ فَلْيَتَوَكَّلِ الْمُؤْمِنُونَ »

REMERCIEMENTS

Mes sincères remerciements s'adressent à M. BENKABOU Seif Eddine et Emmanuel GROLLEAU, mes encadrants de stage, pour leur soutien inestimable tout au long de mon projet de fin d'études. Leur expertise avérée et leurs conseils éclairés ont grandement contribué à mon apprentissage et à la réussite de ce travail. Je tiens également à souligner l'accompagnement précieux de M. MATALLAH Houcine et Amin MESMOUDI dans toutes les démarches liées à ce stage.

Je saisis cette opportunité pour remercier chaleureusement les membres du jury : M. BENZAOUZ Mourtaoua, maître de conférences à l'université Aboubekr Belkaid, ainsi que M. ZIANI CHERIF Salim, maître de conférences au département d'informatique de l'université Aboubekr Belkaid.

Je souhaite adresser ma reconnaissance à l'ensemble des membres du laboratoire LIAS, notamment Kamni Soulimane, Saidi Boumedienne, avec qui j'ai eu le privilège de collaborer. Leur environnement propice à l'apprentissage et leur esprit collaboratif m'ont offert une expérience enrichissante au cours de laquelle j'ai pu acquérir de précieuses connaissances et compétences.

Je veux adresser un sincère merci à ma mère, un pilier inébranlable de soutien tout au long de mon parcours. Son encouragement constant et son amour inconditionnel ont été la source de ma détermination. Mon oncle, Berrahou Mokhtar, mérite également toute ma reconnaissance pour ses précieux conseils et son accompagnement bienveillant.

Ma famille, sans exception, a toujours été présente pour me soutenir dans les moments clés. Leur encouragement a été une source de motivation sans pareille.

Un remerciement spécial également à mes amies Bentoumi Hanane, Berrahma Djazia, Bouanani Hanane, Hamda Nour El Imane dont le soutien indéfectible a illuminé les jours les plus difficiles.

Enfin, je veux exprimer ma reconnaissance envers toutes les personnes qui m'ont apporté leur soutien. Vos gestes, paroles et encouragements ont joué un rôle déterminant dans ma réussite.

RÉSUMÉ

La société SRD se démarque par son réseau de distribution d'électricité hautement technologique, axé sur l'efficacité, l'intégration des énergies renouvelables et la fiabilité de l'énergie fournie. En partenariat avec le laboratoire LIAS, SRD accède aux dernières avancées en modélisation éolienne et en apprentissage automatique, renforçant ainsi la crédibilité de ses projets et favorisant un échange de connaissances fructueux entre l'industrie et le milieu académique.

Notre travail consiste à développer des modèles de prévision avancée pour la production éolienne en utilisant des techniques de traitement des données temporelles, notamment les séries temporelles. Ces modèles sont conçus pour anticiper avec précision la production éolienne sur deux échelles temporelles spécifiques : à court terme (2 heures) pour la gestion en temps réel de l'équilibre production-demande d'électricité, et à moyen terme (1 mois) pour l'ajustement de la planification opérationnelle en fonction des saisons et des conditions météorologiques.

Pour atteindre ces objectifs, nous avons mis en œuvre divers modèles de réseaux de neurones, tels que les réseaux LSTM et GRU, spécialement adaptés pour capturer les dépendances temporelles complexes présentes dans les données éoliennes.

Mots clés : Apprentissage automatique, Modèles de prévision, Traitement des données temporelles, Séries temporelles.

TABLE DES MATIÈRES

REMERCIEMENTS	iv
RÉSUMÉ	v
TABLE DES MATIÈRES	vi
LISTE DES FIGURES	viii
LISTE DES TABLEAUX	ix
LISTE DES SIGLES ET ACRONYMES	x
INTRODUCTION GÉNÉRALE	1
1 CONTEXTE GÉNÉRAL	3
1.1 INTRODUCTION	4
1.2 LES ÉNERGIES ÉOLIENNE ET PHOTOVOLTAÏQUE : SOURCES D'INNOVATION À POITIERS, FRANCE	4
1.3 FONDEMENTS THÉORIQUES DES SÉRIES TEMPORELLES	5
1.3.1 Définition	5
1.3.2 Propriétés statistiques des séries temporelles	6
1.3.3 Modèle de décomposition	6
1.3.4 Cas d'utilisation des séries temporelles	9
1.4 PROBLÉMATIQUE ET OBJECTIF DU PROJET	9
1.5 CONCLUSION	10
2 PRÉSENTATION DES MODÈLES DE PRÉVISION	12
2.1 INTRODUCTION	13
2.2 MODÈLES CLASSIQUES	13
2.2.1 BOX-Jenkins	13
2.2.2 ARIMA	14
2.2.3 SARIMA	15
2.3 MODÈLES BASÉS SUR LES RÉSEAUX DE NEURONES	17
2.3.1 Modèle RNN	17
2.3.2 Modèle LSTM	20
2.3.3 Modèle GRU	22
2.3.4 Comparaison entre les réseaux de neurones récurrents LSTM, RNN et GRU	23

2.4	COMPARAISON ENTRE LES MODÈLES CLASSIQUES DE PRÉVISION ET LES MODÈLES DE RÉSEAUX DE NEURONES	23
2.5	CONCLUSION	25
3	CONTRIBUTIONS & APPLICATION AUX DONNÉES SRD	26
3.1	INTRODUCTION	27
3.2	OBJECTIF DÉTAILLÉ	27
3.3	GESTION DES DONNÉES	28
3.3.1	Collection de données	28
3.3.2	Description de données	28
3.3.3	Qualité de données	29
3.3.4	Prétraitement de données	29
3.4	MÉTHODOLOGIE DE PRÉVISION	30
3.4.1	Méthodologie d'évaluation	30
3.4.2	Métriques de performance	31
3.4.3	Sélection de l'optimiseur pour l'entraînement de modèles	32
3.4.4	Décomposition des series temporelles	34
3.5	IMPLÉMENTATION DES APPROCHES	35
3.5.1	Prévision court terme	36
3.5.2	Prévision moyen terme	37
3.6	RÉSULTATS	37
3.6.1	Prévision court terme	37
3.6.2	Prévision moyen terme	38
3.7	DISCUSSION	38
3.8	SYNTHÈSE DES RÉSULTATS	41
3.9	CONCLUSION	41
	CONCLUSION GÉNÉRALE & PERSPECTIVES	44

LISTE DES FIGURES

1.1	Logo du laboratoire LIAS	5
1.2	Logo du service SRD	5
1.3	Décomposition de la série temporelle	8
2.1	Comparaison : Réseau de neurones récurrents vs Réseau de neurones nourri vers l'avant	17
2.2	L'architecture d'un réseau de neurones récurrents.	18
2.3	Les différents types du modèle RNN.	20
2.4	L'architecture d'un modèle LSTM.	21
2.5	L'architecture d'un modèle GRU.	24
3.1	Exemple d'une série temporelle	34
3.2	Les sous-séquences d'une série après décomposition.	35
3.3	Schéma représentant l'architecture des modèles LSTM et GRU.	37
3.4	Courbes qui représentent la production éolienne de chaque famille de producteurs.	42
3.5	Comparaison des données réelles et des prédictions des modèles ARIMA, LSTM et GRU : Un Aperçu de la Performance sur 1% des Données.	43

LISTE DES TABLEAUX

2.1	Comparaison des modèles d'apprentissage Box-Jenkins, ARIMA et SARIMA	16
2.2	Tableau des éléments d'un réseau de neurones récurrents	18
2.3	Tableau des éléments d'une cellule LSTM	21
2.4	Comparaison des réseaux de neurones récurrents LSTM, RNN et GRU	25
3.1	Spécifications de l'ordinateur utilisé pour l'évaluation expérimentale.	34
3.2	Répartition des producteurs par puissance	39
3.3	Comparaison des performances moyennes (RMSE) des modèles LSTM, GRU et ARIMA pour différentes familles de producteurs .	40
3.4	Comparaison des Temps d'Exécution Moyens en minutes entre les Modèles LSTM, GRU et Modèle ARIMA pour Différentes Familles de Producteurs	40

LISTE DES SIGLES ET ACRONYMES

PACF	<i>Partial Autocorrelation Function</i>
ACF	<i>Autocorrelation Function</i>
SRD	<i>Service des Réseaux de Distribution d'électricité</i>
LIAS	<i>Laboratoire d'Informatique et d'Automatique pour les Systèmes</i>
AR	<i>Auto Regressive</i>
MA	<i>Moving Average</i>
ARIMA	<i>Auto Regressive Integrated Moving Average</i>
ARMA	<i>Auto Regressive Moving Average</i>
SARIMA	<i>Seasonal Auto Regressive Integrated Moving Average</i>
RNN	<i>Recurrent Neural Network</i>
LSTM	<i>Long Short-Term Memory</i>
GRU	<i>Gated Recurrent Unit</i>
MAE	<i>Mean Absolute Error</i>
MSE	<i>Mean Square Error</i>
RMSE	<i>Root Mean Square Error</i>
ADF	<i>Augmented Dickey-Fuller</i>
ADAM	<i>Augmented Dynamic Adaptive Model</i>
SGD	<i>stochastic gradient descent</i>

Introduction générale

L'utilisation croissante et souvent non durable des ressources énergétiques traditionnelles a conduit à une prise de conscience mondiale de la nécessité de passer à des sources plus durables et respectueuses de l'environnement. Dans ce contexte, les énergies renouvelables ont émergé comme une solution cruciale pour répondre aux défis du changement climatique, de la sécurité énergétique et du développement durable. Ces formes d'énergie, puisées directement à partir des sources naturelles telles que le soleil, le vent et l'eau, offrent un potentiel considérable pour réduire les émissions de gaz à effet de serre, limiter la dépendance aux combustibles fossiles et favoriser une croissance économique plus résiliente. En effet, les émissions de gaz à effet de serre de l'Union européenne ont diminué de 33,3 % entre 1990 et 2020, représentant une réduction totale de 1 595 millions de tonnes d'équivalent CO₂. Cette tendance s'est poursuivie avec une baisse continue de 2,8 % en 2021 et une nouvelle diminution de 2,9 % au premier trimestre 2023 par rapport à 2022 [11]. Ces chiffres témoignent de l'engagement résolu de l'Union européenne à adopter des énergies plus propres et à réduire son empreinte carbone de manière significative.

Le département de la Vienne s'est engagé dans le développement significatif de sources d'énergie renouvelable, en tirant parti des ressources naturelles abondantes de la région, notamment le soleil et le vent. Cette initiative a permis d'augmenter progressivement la capacité de production d'énergie propre, en mettant en place des projets solaires photovoltaïques et en installant des parcs éoliens dans des zones appropriées. L'objectif principal de ces efforts est de réduire les émissions de gaz à effet de serre, de favoriser l'autonomie énergétique locale, et de contribuer aux objectifs environnementaux à la fois au niveau national et régional.

Dans le contexte du développement des énergies renouvelables, la SRD initie et soutient activement une gamme de projets d'énergies renouvelables, notamment des installations solaires photovoltaïques et des parcs éoliens. Ces projets permettent de convertir les ressources naturelles, telles que la lumière solaire et le vent, en énergie propre et abordable. En collaborant avec des partenaires locaux, des entreprises et des acteurs de l'industrie, la SRD vise à accélérer la transition énergétique de la Vienne en augmentant sa capacité de production d'énergie renouvelable.

La première phase de cette transition consiste à instrumenter le réseau électrique avec des systèmes de communication et de collecte de données, présentés sous forme de séries temporelles univariées. En installant des capteurs et des dispositifs de surveillance à différents points du réseau, la SRD vise à obtenir une compréhension approfondie de la dynamique du réseau en temps réel.

Ces systèmes permettent de recueillir des informations sur les fluctuations de la demande et de la production d'énergie, ainsi que sur les conditions météorologiques, qui sont particulièrement importantes pour les sources d'énergie renouvelable telles que l'énergie solaire et éolienne.

Dans la deuxième phase, la SRD s'attache à élaborer des outils d'aide à la décision en exploitant et en surveillant ces données d'état instantané du réseau. Grâce à l'analyse de ces informations en temps réel, la SRD peut optimiser la gestion de l'énergie en ajustant la production, la distribution et la consommation en fonction des besoins actuels. Cette approche permet de mieux équilibrer l'offre et la demande, de minimiser les pertes d'énergie et d'améliorer l'efficacité globale du système électrique.

L'objectif principal de ce projet consiste à anticiper la production éolienne, ce qui permettra d'optimiser l'efficacité du réseau de distribution en garantissant une précision suffisante pour s'approcher au plus près d'un schéma d'exploitation réaliste. Cette combinaison de l'optimisation du réseau et des techniques de prévision vise à améliorer la gestion des variations temporelles et spatiales des producteurs et des consommateurs d'énergie renouvelable.

Par la suite, il s'agit de développer des approches novatrices pour prédire la production des énergies renouvelables et la consommation d'électricité, en tirant parti des diverses données internes disponibles. Ce processus intégré permettra ainsi de mieux appréhender et gérer les complexités liées à la variabilité des sources d'énergie et des besoins en électricité. L'apprentissage profond, en tant qu'élément clé, contribue de manière significative à la réalisation de cet objectif ambitieux en permettant une analyse approfondie des modèles et des tendances cachées au sein des données.

Ce rapport plonge profondément dans le domaine de la prévision de séries temporelles en trois chapitres distincts. Dans le premier chapitre, nous exposons le contexte général, définissant ce qu'est une série temporelle, explorant ses propriétés statistiques et examinant ses applications dans le monde réel. Le deuxième chapitre se penche sur les modèles de prévision classiques et les modèles basés sur les réseaux de neurones, analysant leurs avantages et leurs limites. Notre objectif central est révélé dans le troisième chapitre, où nous décrivons les données utilisées, exposons nos méthodologies de prévision et présentons les implémentations des approches explorées. Après cette présentation approfondie, nous consacrons une section à la discussion des résultats obtenus. Enfin, nous concluons ce rapport par une synthèse générale des conclusions de notre étude et explorons les perspectives futures.

CONTEXTE GÉNÉRAL



SOMMAIRE

1.1	INTRODUCTION	4
1.2	LES ÉNERGIES ÉOLIENNE ET PHOTOVOLTAÏQUE : SOURCES D'INNOVATION À POITIERS, FRANCE	4
1.3	FONDEMENTS THÉORIQUES DES SÉRIES TEMPORELLES	5
1.3.1	Définition	5
1.3.2	Propriétés statistiques des séries temporelles	6
1.3.3	Modèle de décomposition	6
1.3.4	Cas d'utilisation des séries temporelles	9
1.4	PROBLÉMATIQUE ET OBJECTIF DU PROJET	9
1.5	CONCLUSION	10

1.1 INTRODUCTION

La prédiction des séries temporelles consiste à estimer les valeurs futures d'une série temporelle en se basant sur les observations passées. C'est un domaine d'étude essentiel dans de nombreux domaines tels que la finance, l'économie, la météorologie, la santé, les sciences sociales et bien d'autres. La capacité de prévoir avec précision l'évolution d'une série temporelle offre de nombreux avantages, tels que la planification efficace, la prise de décisions éclairées et l'optimisation des ressources.

1.2 LES ÉNERGIES ÉOLIENNE ET PHOTOVOLTAÏQUE : SOURCES D'INNOVATION À POITIERS, FRANCE

Les énergies éolienne et photovoltaïque représentent deux facettes cruciales de la transition énergétique, contribuant à réduire notre dépendance aux combustibles fossiles tout en limitant les émissions de gaz à effet de serre. L'énergie éolienne est générée par la rotation des pales d'éoliennes grâce à la force du vent. Les éoliennes convertissent cette énergie cinétique en électricité par l'intermédiaire d'un générateur. L'énergie photovoltaïque, quant à elle, exploite la capacité des cellules solaires à convertir la lumière du soleil en électricité à travers le phénomène photovoltaïque. Les panneaux solaires sont constitués de nombreuses cellules solaires reliées en série pour produire de l'électricité à partir du rayonnement solaire [1].

À Poitiers, en France, la société SRD joue un rôle crucial dans le domaine des énergies renouvelables. Cette entreprise est impliquée dans la fabrication et la maintenance d'équipements liés à l'énergie éolienne et photovoltaïque. Elle contribue ainsi à la croissance de ces secteurs en France en développant des technologies plus efficaces et durables, et en fournissant des services de maintenance pour garantir la fiabilité des installations.

Le laboratoire LIAS à Poitiers a également un rôle essentiel dans la recherche et le développement des énergies renouvelables. Il se concentre sur l'optimisation des systèmes de production d'énergie, y compris ceux basés sur l'énergie éolienne et photovoltaïque, en utilisant des approches informatiques et automatisées avancées. Le LIAS contribue ainsi à améliorer l'efficacité, la gestion et la fiabilité des systèmes d'énergie renouvelable, ce qui est essentiel pour maximiser leur contribution à la transition énergétique.



FIGURE 1.1 – Logo du laboratoire LIAS
Source : [21]



FIGURE 1.2 – Logo du service SRD
Source : [34]

1.3 FONDEMENTS THÉORIQUES DES SÉRIES TEMPORELLES

1.3.1 Définition

Une série temporelle est un type spécial de données où les observations sont enregistrées dans un ordre chronologique afin de représenter l'évolution d'un phénomène observé au fil du temps. Contrairement aux données classiques, les séries temporelles sont caractérisées par la dépendance entre les observations successives, ce qui signifie que chaque observation est influencée par les valeurs précédentes et peut également avoir un impact sur les valeurs futures.

Dans le domaine de la météorologie, les séries temporelles jouent un rôle essentiel pour la compréhension et la prédiction des conditions météorologiques. Les observations météorologiques, telles que la température, la pression atmosphérique, l'humidité, la vitesse et la direction du vent, sont enregistrées à intervalles réguliers pour former des séries temporelles [36].

1.3.1.1 Série temporelle univariée

Une série temporelle univariée est une séquence ordonnée de points de données qui sont enregistrés ou mesurés à des intervalles réguliers dans le temps. Chaque observation dans une série temporelle univariée est associée à un instant spécifique dans le temps, ce qui permet d'analyser et de comprendre les motifs et les tendances qui se produisent au fil du temps. Dans une série temporelle univariée, il n'y a qu'une seule variable d'intérêt mesurée, telle que la température quotidienne, le chiffre d'affaires mensuel d'une entreprise, le nombre de ventes trimestrielles d'un produit, etc. Cette variable est enregistrée à intervalles de temps réguliers, par exemple, toutes les heures, tous les jours, toutes les semaines ou tous les mois [25].

1.3.2 Propriétés statistiques des séries temporelles

Les séries temporelles présentent plusieurs propriétés statistiques qui sont importantes à prendre en compte lors de leur analyse et de leur modélisation [8] :

- **Tendance** : la tendance se réfère à la direction générale dans laquelle la série temporelle évolue sur une période prolongée. Une tendance peut être ascendante (croissante), descendante (décroissante) ou horizontale (constante). La présence d'une tendance indique un comportement systématique dans les données au fil du temps.
- **Saisonnalité** : la saisonnalité fait référence à des variations régulières et récurrentes dans la série temporelle qui se produisent à des intervalles de temps fixes. Ces variations sont souvent associées à des facteurs saisonniers tels que les cycles annuels, mensuels, hebdomadaires ou diurnes. Par exemple, les ventes de crème glacée peuvent présenter une saisonnalité avec des pics pendant les mois d'été.
- **Cyclicité** : la cyclicité se réfère à des fluctuations de long terme qui ne sont pas nécessairement périodiques. Contrairement à la saisonnalité, la cyclicité n'a pas de durée fixe et peut varier en termes de longueur et d'amplitude. Les cycles peuvent être influencés par des facteurs économiques, politiques ou sociaux et peuvent se répéter sur plusieurs années.
- **Autocorrélation** : l'autocorrélation est la corrélation entre les valeurs d'une série temporelle et ses propres valeurs retardées dans le temps. Elle mesure la dépendance entre les observations successives. Une forte autocorrélation indique une tendance à la persistance ou à la répétition des motifs dans la série temporelle.
- **Stationnarité** : la stationnarité est une propriété importante des séries temporelles. Une série est considérée comme stationnaire si ses propriétés statistiques ne varient pas dans le temps, notamment en termes de moyenne, de variance et de covariance. La stationnarité facilite l'analyse et la modélisation des séries temporelles.
- **Fluctuations aléatoires** : les séries temporelles peuvent contenir des variations aléatoires, également appelées bruit, qui ne peuvent pas être expliquées par les autres composantes.

1.3.3 Modèle de décomposition

L'analyse des séries temporelles permet de détecter des tendances, des motifs saisonniers ou des résidus (appelé également bruit). Pour comprendre comment décomposer une série temporelle, on va voir un exemple en Python, en utilisant le jeu de données AirPassengers de la bibliothèque pandas qui représente une série temporelle univariée contenant des informations sur le nombre mensuel de passagers aériens internationaux entre 1949 et 1960.

1.3.3.1 Exemple de code

```

1 import pandas as pd
2 from statsmodels.tsa.seasonal import seasonal_decompose
3
4 # Charger le jeu de données
5 data = pd.read_csv('https://raw.githubusercontent.com/jbrownlee/
6   Datasets/master/airline-passengers.csv')
7
8 # Afficher les premières lignes du DataFrame
9 print(data.head())
10
11 # Vérifier les valeurs manquantes
12 missing_values = data.isna().sum()
13
14 # Imprimer le nombre de valeurs manquantes pour chaque colonne
15 print(missing_values)
16
17 # Appliquer la décomposition saisonnière à la série temporelle
18 decomposition = seasonal_decompose(data)
19 decomposition.plot()

```

1.3.3.2 Affichage des résultats

```

1   Month  Passengers
2 0 1949-01      112
3 1 1949-02      118
4 2 1949-03      132
5 3 1949-04      129
6 4 1949-05      121

```

```

1 Month      0
2 Passengers 0
3 dtype: int64

```

1.3.3.3 L'analyse des résultats

1. Le DataFrame ne contient pas de valeurs manquantes, comme indiqué par le résultat de la vérification des valeurs manquantes.
2. Un motif saisonnier sur une période de 6 mois et 12 mois est clairement visible (Voir la figure 1.3) : l'utilisation de la méthode 'seasonal-decompose' de la bibliothèque 'statsmodels.tsa.seasonal' permet d'appliquer une décomposition saisonnière à la série temporelle des données des passagers des compagnies aériennes. La décomposition divise la série en plusieurs composantes, notamment une composante saisonnière. Lorsque la méthode plot() est utilisée pour afficher la décomposition, il devient évident qu'il existe un motif saisonnier qui se répète tous les 6 mois et tous les 12 mois.

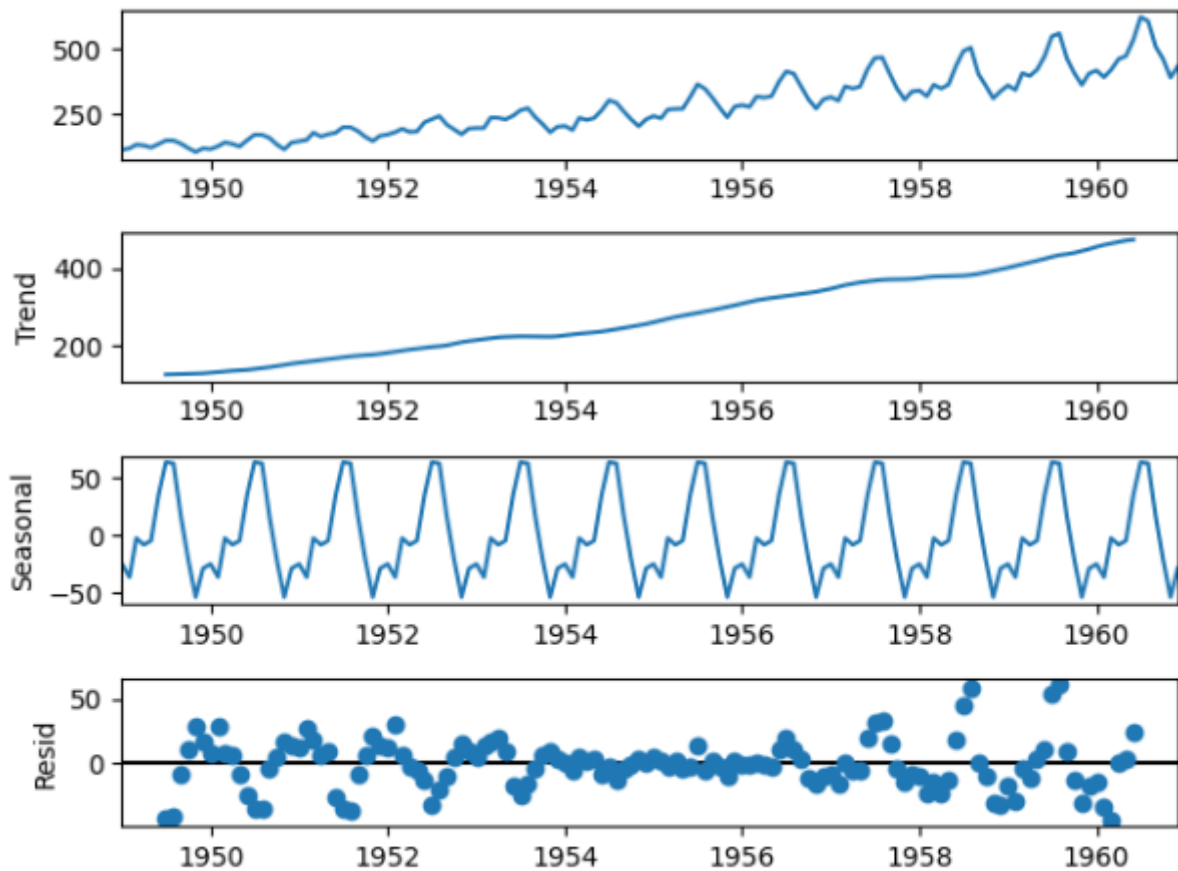


FIGURE 1.3 – Décomposition de la série temporelle

Cette information est précieuse pour comprendre les fluctuations saisonnières dans les données et peut être utilisée pour effectuer des analyses et des prévisions plus précises.

3. Une tendance à la hausse et à la baisse est évidente (Voir la figure 1.3) : en examinant la décomposition saisonnière affichée par la méthode plot(), on peut observer une tendance générale à la hausse et à la baisse dans les données. Cela suggère qu'il y a une variation à long terme dans le nombre de passagers des compagnies aériennes au fil du temps. Cette tendance peut être due à divers facteurs économiques, sociaux ou saisonniers, et elle peut être utilisée pour analyser les tendances à long terme et prendre des décisions éclairées en matière de planification et de prévision.

1.3.4 Cas d'utilisation des séries temporelles

Les séries temporelles, en qualité d'instrument pour analyser des données dans un contexte chronologique, ouvrent la porte à une variété d'applications cruciales [14] :

- **Prévisions** : les séries temporelles permettent de prévoir les valeurs futures, par exemple, prédire les ventes d'un produit, la demande énergétique ou les prix des actions.
- **Analyse des tendances** : les séries temporelles peuvent être utilisées pour analyser les tendances à long terme, détecter les cycles économiques ou étudier l'impact des politiques publiques sur les données.
- **Détection d'anomalies** : les séries temporelles peuvent aider à détecter les événements anormaux ou les comportements inhabituels, tels que des pannes de système, des fraudes financières ou des fluctuations soudaines des données.
- **Modélisation de processus** : les séries temporelles permettent de modéliser et de comprendre les mécanismes sous-jacents à un processus, comme la modélisation de la croissance d'une population, l'évolution des maladies, etc.

1.4 PROBLÉMATIQUE ET OBJECTIF DU PROJET

SRD est une entreprise publique française qui gère le réseau électrique du département de la Vienne. Elle fournit des services de distribution d'électricité à environ 200 000 clients dans la Vienne. SRD est une entreprise importante pour l'économie de la Vienne et joue un rôle essentiel dans la fourniture d'électricité aux habitants de la région[33].

Cette entreprise se trouve face à un défi majeur dans le domaine de l'énergie éolienne en France. Bien que l'énergie éolienne offre d'énormes avantages en tant que source d'énergie renouvelable, sa production est fortement influencée par des conditions météorologiques instables et imprévisibles.

Ces fluctuations rendent la planification précise de la production éolienne particulièrement difficile, ce qui peut entraîner des inefficacités opérationnelles pour les producteurs d'énergie éolienne. En l'absence de prévisions fiables, les gestionnaires de ces installations éprouvent des difficultés à ajuster leur production en temps réel, entraînant ainsi des surproductions inutiles ou, à l'inverse, des périodes de sous-production qui perturbent l'équilibre du réseau électrique national.

De plus, les incertitudes liées à la production éolienne peuvent affecter la stabilité globale du réseau électrique, puisque les gestionnaires doivent constamment s'adapter aux variations imprévues de la production. Cela peut entraîner

des surcharges, des pics de demande difficilement gérables, voire des coupures d'électricité dans les situations les plus critiques.

Pour répondre à ces défis, l'objectif premier de l'entreprise SRD est de concevoir et de mettre en œuvre des solutions technologiques avancées, fondées sur l'apprentissage automatique et l'intelligence artificielle, en utilisant des données historiques sur la production éolienne. Ces modèles de prédictions ont pour but d'améliorer l'efficacité, la rentabilité et la fiabilité des installations éoliennes, tout en facilitant leur intégration harmonieuse dans le réseau électrique national. En utilisant ces avancées technologiques, SRD aspire à rendre l'énergie éolienne plus prévisible et plus stable, contribuant ainsi à accroître son potentiel en tant que source d'énergie renouvelable fiable et durable.

Les avantages de ces modèles de prédictions sont multiples pour les producteurs d'énergie éolienne et pour le secteur de l'énergie en général :

- **Optimisation de la production** : grâce à ces modèles, les producteurs pourront planifier et optimiser leur production en fonction des prévisions météorologiques concernant le vent. Cela permettra de maximiser l'efficacité de leurs installations éoliennes.
- **Gestion du réseau électrique** : les prédictions précises contribueront à une meilleure intégration de l'énergie éolienne dans le réseau électrique. En anticipant les variations de la production éolienne, les gestionnaires de réseau pourront équilibrer plus efficacement l'offre et la demande, réduisant ainsi les risques de surcharge ou de sous-capacité.
- **Prévention des pannes** : les prédictions précises permettent d'identifier les variations potentielles dans la production éolienne, aidant ainsi les entreprises à anticiper les pannes éventuelles et à prendre des mesures préventives pour minimiser les temps d'arrêt.
- **Rentabilité accrue** : une meilleure prévision de la production éolienne permet aux producteurs d'optimiser leur planification opérationnelle, ce qui peut se traduire par une augmentation de la rentabilité globale de leurs installations.
- **Réduction des coûts** : en optimisant la production et en prévoyant les besoins énergétiques, les entreprises peuvent également réduire les coûts liés à la gestion de la production d'énergie éolienne.

1.5 CONCLUSION

En conclusion de ce chapitre, nous avons exploré en détail ce qu'est une série, en mettant l'accent sur ses propriétés statistiques essentielles ainsi que sur ses différentes applications. Cette compréhension fondamentale des séries temporelles servira de base solide pour aborder notre prochain chapitre, où nous

plongerons dans le monde passionnant des modèles de prévision. Dans le chapitre à venir, nous examinerons les prévisions classiques ainsi que les réseaux de neurones.

PRÉSENTATION DES MODÈLES DE PRÉVISION

2

SOMMAIRE

2.1	INTRODUCTION	13
2.2	MODÈLES CLASSIQUES	13
2.2.1	BOX-Jenkins	13
2.2.2	ARIMA	14
2.2.3	SARIMA	15
2.3	MODÈLES BASÉS SUR LES RÉSEAUX DE NEURONES	17
2.3.1	Modèle RNN	17
2.3.2	Modèle LSTM	20
2.3.3	Modèle GRU	22
2.3.4	Comparaison entre les réseaux de neurones récurrents LSTM, RNN et GRU	23
2.4	COMPARAISON ENTRE LES MODÈLES CLASSIQUES DE PRÉVISION ET LES MODÈLES DE RÉSEAUX DE NEURONES	23
2.5	CONCLUSION	25

2.1 INTRODUCTION

Les modèles de prévision exploitent les techniques de l'apprentissage automatique pour analyser les données et identifier les schémas ou les relations qui permettent de faire des prédictions. Ces modèles peuvent être supervisés ou non supervisés. Dans les modèles supervisés, les données historiques sont étiquetées avec des valeurs cibles connues, tandis que dans les modèles non supervisés, les données ne sont pas étiquetées et le modèle cherche à identifier des structures ou des clusters significatifs.

L'une des approches les plus couramment utilisées pour construire des modèles de prévision est l'apprentissage automatique supervisé, où les modèles sont formés à partir de paires de données d'entrée-sortie. Ces données d'entrée, également appelées caractéristiques ou variables indépendantes, sont utilisées pour prédire les valeurs de sortie, également appelées variables dépendantes ou cibles. Les modèles de prévision peuvent utiliser une variété d'algorithmes d'apprentissage automatique, tels que les réseaux de neurones, les arbres de décision, les machines à vecteurs de support (SVM) et les méthodes ensemblistes comme les forêts aléatoires ou le gradient boosting [2].

Une autre approche populaire pour les modèles de prévision est l'apprentissage automatique non supervisé, qui vise à découvrir des structures et des modèles dans les données sans utiliser des valeurs de sortie pré-étiquetées. Ces modèles peuvent être utilisés pour effectuer des tâches telles que la segmentation de clients, la détection d'anomalies, la réduction de dimensionnalité et la visualisation des données [30].

2.2 MODÈLES CLASSIQUES

2.2.1 BOX-Jenkins

Le modèle Box-Jenkins, proposé par George Box et Gwilym Jenkins, est une approche statistique puissante pour l'analyse et la prévision des séries temporelles. Il vise à comprendre la structure sous-jacente des données séquentielles et à développer des modèles appropriés pour les décrire [5].

La méthodologie Box-Jenkins se base sur trois étapes principales : l'identification du modèle, l'estimation des paramètres et la vérification du modèle.

1. **Identification du modèle** : cette étape consiste à analyser les données de la série temporelle pour déterminer les composantes et les motifs présents. On cherche à identifier les composantes d'autorégression (AR) et de moyenne mobile (MA) qui peuvent être incluses dans le modèle. Cela se fait en examinant les autocorrélations et les autocorrélations partielles des données.
2. **Estimation des paramètres** : une fois que le modèle a été identifié, les paramètres du modèle sont estimés à partir des données. Différentes méthodes d'estimation statistique peuvent être utilisées, telles que la méthode des

moindres carrés ordinaires ou la méthode de maximisation de la vraisemblance. L'estimation des paramètres permet de quantifier la relation entre les observations passées et futures de la série temporelle.

3. **Vérification du modèle** : cette étape vise à évaluer la qualité du modèle ajusté. On examine les résidus, qui sont les différences entre les valeurs réelles et les valeurs prédites par le modèle. On vérifie si les résidus sont stationnaires et s'ils présentent une structure systématique. Des tests statistiques sont effectués pour s'assurer que le modèle est adéquat et qu'il ne reste pas de structure dans les résidus[4].

2.2.2 ARIMA

Modèle de moyenne mobile intégré autorégressif est une forme de modèle de BOX-Jenkins qui combine les composantes d'autorégression (AR), de moyenne mobile (MA) et de différenciation intégrée (I). L'ARIMA est une extension du modèle ARMA qui inclut la possibilité de traiter des séries temporelles non stationnaires [26].

1. **Autorégression (AR)** : l'autorégression fait référence à la dépendance linéaire entre une observation et un certain nombre d'observations passées, appelées "retards". Dans un modèle AR, les valeurs précédentes de la série temporelle sont utilisées pour prédire les valeurs futures. L'ordre de l'autorégression, généralement noté AR(p), indique le nombre de retards utilisés pour la prédiction.
2. **Moyenne mobile (MA)** : la moyenne mobile fait référence à la dépendance linéaire entre une observation et les erreurs de prédiction passées. Dans un modèle MA, les erreurs de prédiction passées sont utilisées pour prédire les valeurs futures. L'ordre de la moyenne mobile, généralement noté MA(q), indique le nombre d'erreurs de prédiction utilisées dans le modèle.
3. **Différenciation intégrée (I)** : la différenciation intégrée est utilisée pour rendre une série temporelle stationnaire. La stationnarité est une propriété souhaitée dans de nombreux modèles statistiques. La différenciation intégrée implique de prendre des différences successives entre les observations pour éliminer les tendances ou les variations non stationnaires. L'ordre de différenciation, généralement noté d, indique le nombre de fois que la différenciation est appliquée à la série temporelle.

Un modèle ARIMA est généralement noté $ARIMA(p, d, q)$, où p correspond à l'ordre de l'autorégression, d correspond à l'ordre de différenciation intégrée et q correspond à l'ordre de la moyenne mobile. Les valeurs optimales des paramètres p , d et q sont déterminées en utilisant des techniques telles que l'estimation de la fonction d'autocorrélation partielle (PACF) et l'estimation de la fonction d'autocorrélation (ACF) [32].

2.2.3 SARIMA

SARIMA est une extension du modèle ARIMA qui permet de modéliser et de prévoir les séries temporelles saisonnières. Il intègre les composantes d'autorégression (AR), de moyenne mobile (MA), de différenciation intégrée (I) ainsi que des termes saisonniers [10].


Le modèle SARIMA est utilisé lorsque la série temporelle présente une saisonnalité, c'est-à-dire des variations régulières qui se répètent à intervalles de temps fixes, tels que les variations saisonnières trimestrielles dans les ventes de produits de Noël ou les variations saisonnières mensuelles dans les températures.

Le modèle SARIMA est spécifié en utilisant les paramètres suivants :

1. **Composantes non saisonnières** : les composantes AR, MA et I sont similaires à celles du modèle ARIMA standard. Elles capturent les dépendances autoregressives, les moyennes mobiles et les différences intégrées nécessaires pour rendre la série temporelle stationnaire. Les paramètres p , d et q spécifient l'ordre des termes AR, I et MA respectivement.
2. **Composantes saisonnières AR, MA et I** : ces composantes prennent en compte les variations saisonnières de la série temporelle. Elles capturent les dépendances autoregressives saisonnières, les moyennes mobiles saisonnières et les différences intégrées saisonnières. Les paramètres P , D et Q spécifient l'ordre des termes saisonniers AR, I et MA respectivement.
3. **Période saisonnière (s)** : c'est la longueur du cycle saisonnier dans la série temporelle. Par exemple, si la série est mensuelle et la saisonnalité est annuelle, s serait égal à 12 [19].

Le modèle SARIMA est généralement noté :

$$\text{SARIMA } (p, d, q)(P, D, Q)_s$$



non-saisonnier
saisonnier

Comparaison des modèles d'apprentissage BOX-Jenkins, ARIMA et SARIMA

Les modèles BOX-Jenkins, ARIMA et SARIMA sont largement utilisés dans le domaine de l'analyse des séries temporelles pour capturer les tendances, les autocorrélations et les composantes saisonnières des données.

Chaque modèle possède ses propres caractéristiques et ses avantages, ce qui en fait des outils importants pour modéliser et prévoir les séries temporelles dans divers domaines tels que l'économie, la finance, la météorologie et bien d'autres. Examinons de plus près les caractéristiques et les limitations de chaque modèle dans le contexte de l'analyse des séries temporelles (Voir la table 2.1).

Modèle	Utilisation	Avantages	Inconvénients
Box-Jenkins	Utilisé pour modéliser des séries temporelles stationnaires avec des tendances et des saisons.	- Permet de modéliser des structures complexes de séries temporelles.	- Le choix approprié du modèle peut être complexe à mettre en œuvre pour les débutants en raison de la nécessité d'une expertise en statistiques avancées.
ARIMA	convient aux séries temporelles qui présentent une tendance, à condition qu'elles soient rendues stationnaires grâce à la différenciation intégrée.	- Offre une bonne flexibilité pour modéliser différents types de séries temporelles.	-L'absence de prise en compte des saisons dans les données peut rendre l'estimation peu précise pour les séries très volatiles.
SARIMA	Adapté aux séries temporelles avec des tendances, de l'autocorrélation et des saisons.	- Permet de modéliser efficacement les données saisonnières.	- Nécessite une estimation précise des paramètres, ce qui peut être difficile avec de petites séries temporelles. - Peut-être computationnellement plus coûteux que les modèles ARIMA simples.

TABLE 2.1 – Comparaison des modèles d'apprentissage Box-Jenkins, ARIMA et SARIMA

2.3 MODÈLES BASÉS SUR LES RÉSEAUX DE NEURONES

2.3.1 Modèle RNN

2.3.1.1 Définition

Les Réseaux de Neurones Récurrents (RNN) sont un type de réseau de neurones artificiels conçu pour traiter des données temporelles et modéliser des séquences de longueur variable. Sa structure est constituée de cellules récurrentes qui ont la capacité de stocker les Informations calculées en mémoire pour être réutilisées lors d'un prochain calcul, cette caractéristique de mémorisation s'appelle l'état caché du RNN qui utilise les mêmes paramètres pour chaque élément d'une séquence, car il effectue la même tâche sur tous les éléments ou les couches cachées pour produire la sortie. Pour cette raison, les RNN sont appelés récurrents [37].

Contrairement aux réseaux de neurones classiques, qui traitent chaque entrée indépendamment d'autres, c.-à-d. que la donnée d'entrée passe une fois dans chaque couche, ce processus s'appelle le **feedforward** (nourrit vers l'avant).

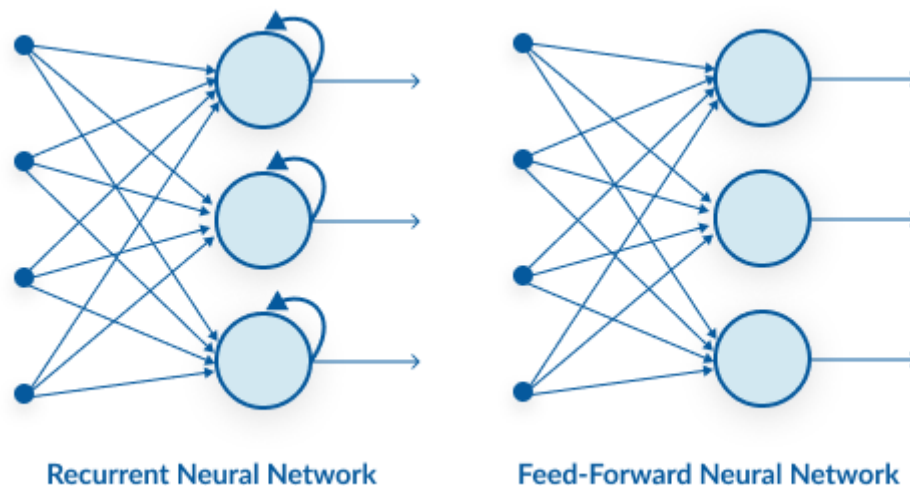


FIGURE 2.1 – Comparaison : Réseau de neurones récurrents vs Réseau de neurones nourri vers l'avant

Source : [27]

2.3.1.2 Architecture d'un réseau neuronal récurrent

L'architecture d'un RNN est basée sur des unités récurrentes appelées cellules, qui se répètent à chaque instant de temps et sont connectées entre elles pour former une séquence.

Chaque cellule prend en entrée une donnée à un instant de temps donné, ainsi que l'état interne calculé lors de l'instant de temps précédent. L'état interne de la cellule est mis à jour à chaque instant de temps en fonction de l'entrée actuelle et de l'état précédent (Voir la figure 2.2). Cette mise à jour permet au RNN de conserver et de mettre à jour l'information séquentielle à mesure qu'elle progresse dans la séquence[15].

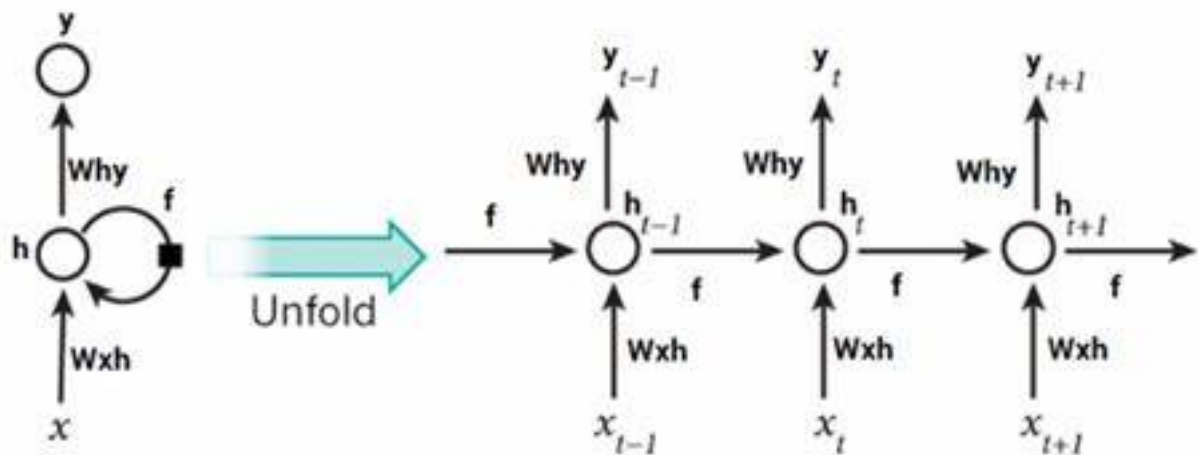


FIGURE 2.2 – L'architecture d'un réseau de neurones récurrents.
Source : [17].

Symbole	Description
h_t	état courant
h_{t-1}	état précédent
X_t	entrée au temps t
W	poids au niveau du neurone récurrent
U	poids au neurone d'entrée
V	poids à la couche de sortie
Y_t	sortie à pas de temps t

TABLE 2.2 – Tableau des éléments d'un réseau de neurones récurrents

2.3.1.3 Types de RNN

Selon la tâche à accomplir, différentes configurations de données peuvent être utilisées pour exploiter pleinement les capacités des RNN. Quatre relations courantes entre les données d'entrée et de sortie sont[6] :

1. **Tête-à-tête (One-to-one) :**

chaque donnée d'entrée est associée à une seule donnée de sortie. Cette configuration est couramment utilisée dans les problèmes de classification, où chaque instance d'entrée doit être classée dans une seule catégorie. Par exemple, dans la classification d'images, chaque image est traitée individuellement et associée à une étiquette de classe unique.

2. **Un à plusieurs (One-to-many) :**

une seule donnée d'entrée est associée à plusieurs données de sortie. Cette configuration est souvent utilisée dans la génération de texte, où un modèle RNN prend une seule donnée d'entrée, telle qu'un mot ou un caractère, et génère une séquence de mots ou de caractères en sortie. Par exemple, dans la génération de légendes d'images, une seule image peut être utilisée en entrée pour générer une description détaillée associée à cette image.

3. **Plusieurs à un (Many-to-one) :**

plusieurs données d'entrée sont traitées simultanément, mais le modèle RNN génère une seule sortie agrégée pour toutes les entrées. Cette configuration est utile lorsque l'on souhaite obtenir une prédiction globale ou un résumé des données d'entrée. Par exemple, dans l'analyse de sentiments, plusieurs phrases peuvent être regroupées pour déterminer le sentiment global exprimé dans un texte. Les RNN sont capables de capturer les relations contextuelles entre les données et d'émettre une prédiction globale basée sur l'ensemble des entrées.

4. **Plusieurs à plusieurs (Many-to-many) :** plusieurs données d'entrée sont

associées à plusieurs données de sortie, où chaque donnée d'entrée est liée à une donnée de sortie spécifique. Cette configuration est couramment utilisée dans la traduction automatique, où une séquence de mots en entrée est associée à une séquence de mots en sortie dans une autre langue. Les RNN peuvent être utilisés pour capturer les dépendances à long terme dans les séquences et générer des prédictions précises correspondant à chaque élément de la séquence d'entrée.

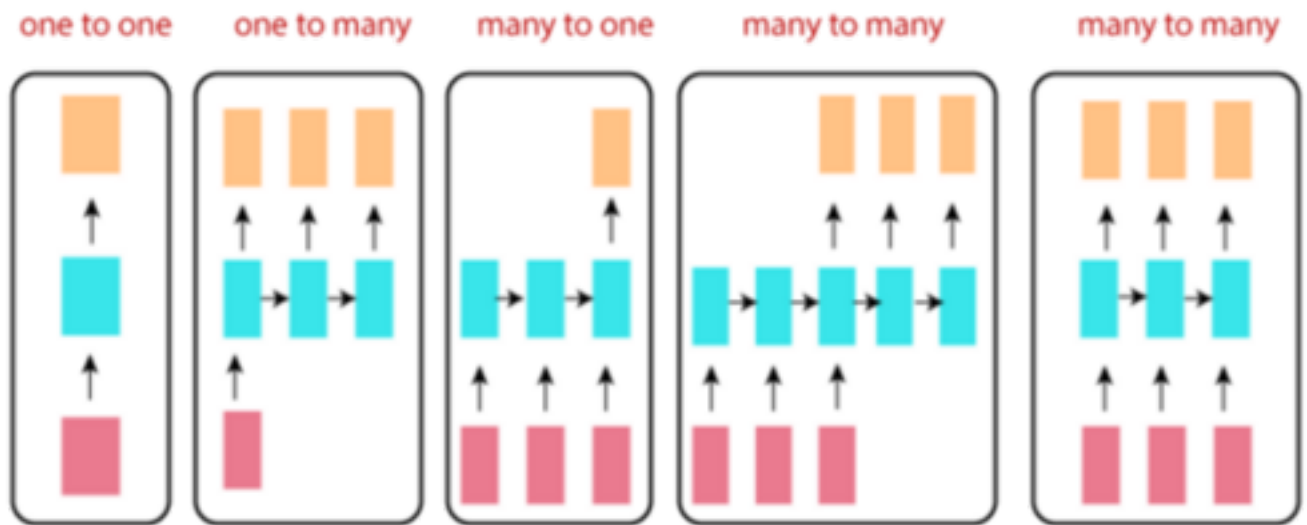


FIGURE 2.3 – Les différents types du modèle RNN.
Source : [35].

2.3.2 Modèle LSTM

2.3.2.1 Définition

Un modèle *Long Short-Term Memory* (LSTM) est un type de modèle de réseau de neurones récurrents (RNN) largement utilisé dans le domaine de l'apprentissage automatique. Il est conçu pour traiter des séquences de données, telles que des séquences de mots dans du texte, des séries temporelles ou des séquences d'observations.

La particularité des modèles LSTM réside dans leur capacité à capturer et à mémoriser les dépendances à long terme dans les séquences [22].

2.3.2.2 Architecture d'un modèle LSTM

Contrairement aux RNN traditionnels, les LSTM utilisent des unités de mémoire appelées "cellules LSTM" qui sont capables de retenir des informations pendant de longues périodes de temps.

Chaque cellule LSTM est composée de plusieurs portes (Voir la figure 2.4), telles que la porte d'oubli (forget gate), la porte d'entrée (input gate) et la porte de sortie (output gate). Ces portes contrôlent le flux d'informations à l'intérieur de la cellule LSTM et permettent de décider quelles informations doivent être conservées, oubliées ou produites en sortie :

1. **Porte d'oubli (Forget Gate)** :elle permet au réseau de supprimer les informations qui ne sont plus pertinentes de la cellule de mémoire à l'étape actuelle.
2. **Porte d'entrée (Input Gate)** :elle est contrôlée par une fonction d'activation. Elle détermine aussi la quantité de nouvelles informations à stocker dans la cellule de mémoire à l'étape actuelle.
3. **Porte de sortie (Output Gate)** :elle détermine la quantité d'informations à transmettre à la sortie du réseau à l'étape actuelle.

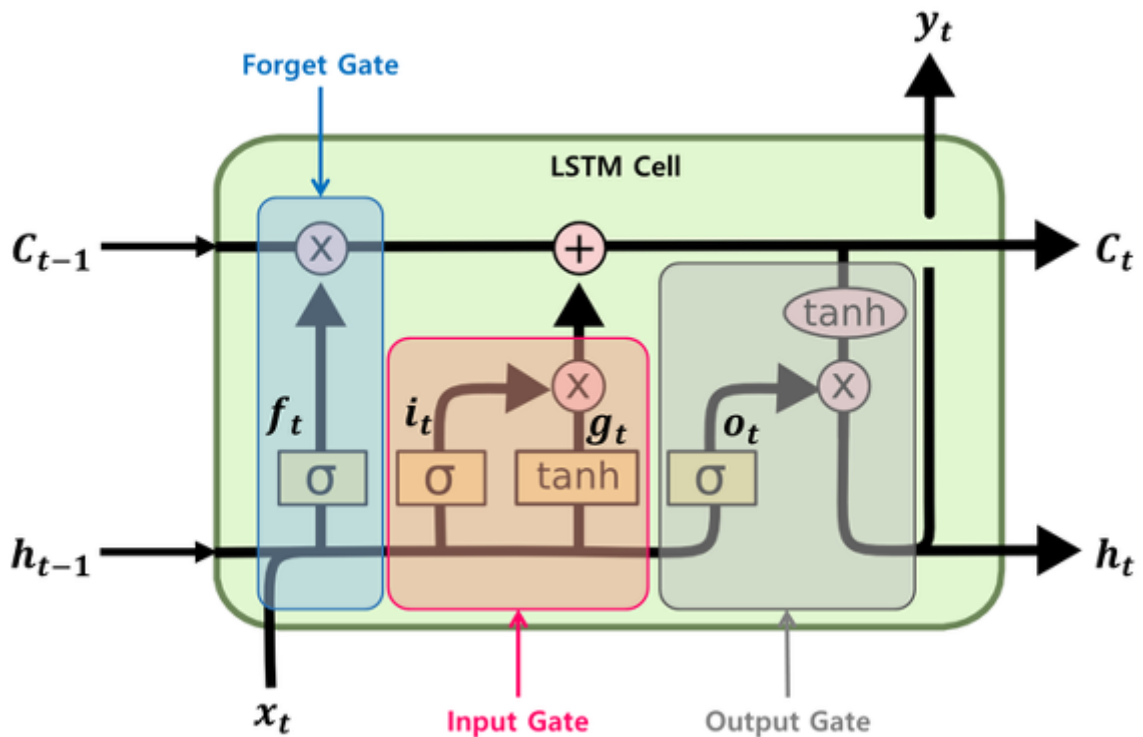


FIGURE 2.4 – L'architecture d'un modèle LSTM.
Source : [23].

Symbole	Description
C_t	état de la cellule au temps t
h_t	sortie d'état masqué au temps t
C_{t-1}	état de la cellule au temps t-1
h_{t-1}	entrée d'état masqué au temps t-1
Y_t	donnée de sortie au temps t
X_t	donnée d'entrée au temps t

TABLE 2.3 – Tableau des éléments d'une cellule LSTM

On peut distinguer deux concepts qui jouent différents rôles :

1. **L'état de la cellule (C :cell state)** : également appelé état de la mémoire à long terme, est une composante clé d'une cellule LSTM. Il est utilisé pour stocker et transporter des informations sur une longue période. L'état de la cellule est mis à jour à chaque pas de temps en fonction des informations entrantes et sortantes de la cellule. Il est conçu pour permettre aux informations importantes d'être conservées sur une longue durée, tandis que les informations moins importantes peuvent être supprimées au fil du temps.
2. **L'état masqué (h :hidden state)** : l'état masqué est la sortie de l'unité LSTM à chaque pas de temps. Il représente l'information utile extraite de l'état de cellule et est utilisé pour prédire ou influencer d'autres parties du réseau. L'état masqué est calculé à partir de l'état de cellule en passant à travers une fonction d'activation et en le filtrant avec la porte de sortie [16].

2.3.3 Modèle GRU

2.3.3.1 Définition

Le modèle GRU est un type de réseau de neurones récurrents (RNN) qui a été introduit par Kyunghyun Cho et al. en 2014 [7]. Il utilise des mécanismes de portes pour réguler l'information et capturer efficacement les dépendances séquentielles à long terme dans les données. Il a été conçu pour résoudre les problèmes de "vanishing gradient" (gradient qui s'annule) souvent rencontrés dans les RNN traditionnels, ce qui en fait un choix populaire pour la prédiction de séries temporelles et d'autres tâches qui impliquent des données séquentielles[31].

2.3.3.2 Architecture du modèle GRU

Le modèle GRU est composé de plusieurs unités GRU empilées les unes sur les autres pour former une architecture en couches. Chaque unité GRU dispose de mécanismes de portes qui contrôlent le flux d'informations entre les étapes de temps successives (Voir la figure 2.5). Voici les principaux éléments de l'architecture d'un modèle GRU :

1. **Entrée (Input)** : la première étape de l'architecture GRU consiste à fournir des séquences d'entrée. Dans le contexte de la prédiction de séries temporelles, chaque séquence est constituée d'observations mesurées à des intervalles de temps réguliers. Les séquences d'entrée sont représentées sous forme de vecteurs, où chaque composante du vecteur représente une caractéristique de l'observation à un instant de temps donné.
2. **État caché (Hidden State)** : le modèle GRU utilise un état caché (ou état interne) pour capturer les informations séquentielles au fil du temps. Cet

état caché est également représenté sous forme de vecteur. Il est ainsi mis à jour à chaque pas de temps en fonction des entrées actuelles et de l'état caché précédent.

3. **Porte de réinitialisation (Reset gate)** : cette porte détermine quelle partie de l'information précédente doit être oubliée. Elle prend en entrée l'observation actuelle et l'état caché précédent (sortie de la couche GRU à l'instant de temps précédent) et génère un vecteur de réinitialisation. Ce vecteur indique quelles informations de l'état caché précédent doivent être oubliées et quels nouveaux états candidats doivent être pris en compte.
4. **Porte de mise à jour (Update gate)** : Elle détermine quelles parties de la mémoire à court terme doivent être mises à jour avec les informations actuelles et quelles parties doivent rester inchangées.

2.3.4 Comparaison entre les réseaux de neurones récurrents LSTM, RNN et GRU

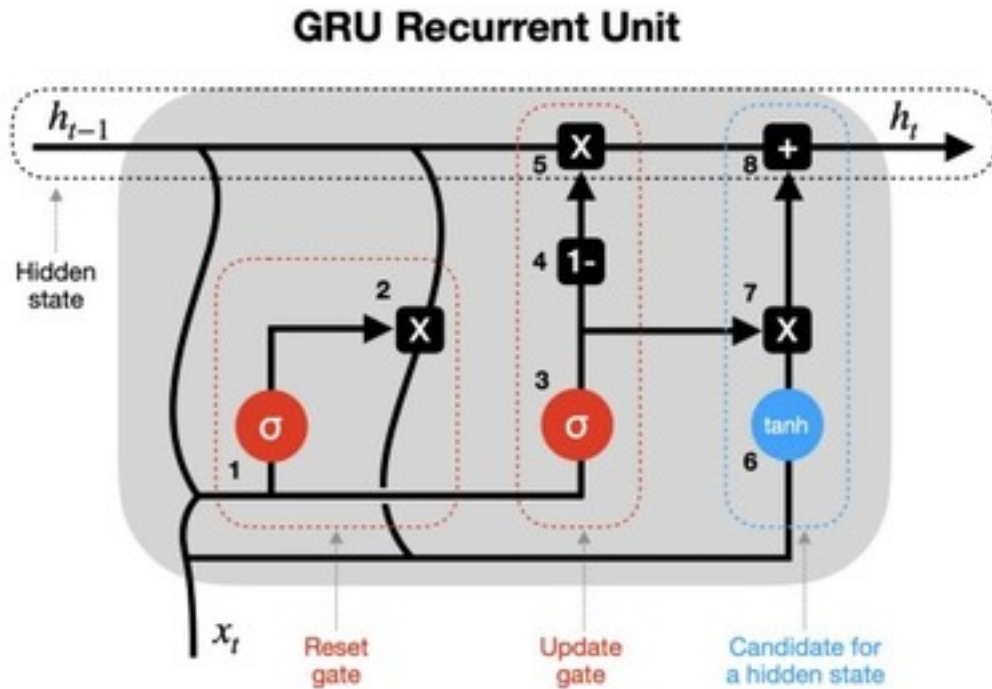
Les LSTM et les GRU sont des architectures avancées du RNN classique, spécifiquement conçues pour résoudre le problème de disparition ou d'explosion du gradient. Cette amélioration leur permet d'être plus efficaces et de capturer des dépendances à plus long terme dans les données séquentielles.

Le choix de l'architecture dépend du contexte spécifique de la tâche, des ressources disponibles et des contraintes de temps de calcul. Dans de nombreux cas, les LSTM et les GRU démontrent des performances similaires, mais le GRU peut être préféré en raison de sa complexité réduite et de son efficacité accrue (Voir la table 2.4).

2.4 COMPARAISON ENTRE LES MODÈLES CLASSIQUES DE PRÉVISION ET LES MODÈLES DE RÉSEAUX DE NEURONES

Les modèles classiques de prévision tels que ARIMA (AutoRegressive Integrated Moving Average) et SARIMA (Seasonal ARIMA) ont longtemps été utilisés pour analyser les séries temporelles et effectuer des prévisions. Ces modèles se basent sur des relations linéaires entre les observations passées et l'observation future à prédire. ARIMA modélise les relations auto-régressives et les moyennes mobiles, tandis que SARIMA étend ces concepts pour prendre en compte les composantes saisonnières.

En revanche, les modèles de réseaux de neurones tels que LSTM (Long Short-Term Memory) et GRU (Gated Recurrent Unit) appartiennent à la famille des réseaux de neurones récurrents (RNN). Ces modèles ont la capacité d'apprendre des dépendances à long terme dans les séries temporelles, ce que les modèles



h_{t-1} - hidden state at previous timestep t-1 (memory)

x_t - input vector at current timestep t

h_t - hidden state at current timestep t

X - vector pointwise multiplication **+** - vector pointwise addition

tanh - tanh activation function

σ - sigmoid activation function

Y - concatenation of vectors

○ - states

○ - gates

○ - updates

FIGURE 2.5 – L'architecture d'un modèle GRU.

Source : [9].

classiques ont souvent du mal à capturer. Les LSTM et les GRU sont équipés de mécanismes spéciaux pour mémoriser l'information sur de longues séquences, évitant ainsi les problèmes de disparition ou d'explosion des gradients.

Une différence clé entre les deux réside dans leur flexibilité. Les modèles classiques comme ARIMA nécessitent une expertise domaniale pour sélectionner les ordres appropriés (p, d, q) et (P, D, Q, S) pour ARIMA et SARIMA respectivement. En revanche, les réseaux de neurones sont plus adaptatifs et peuvent apprendre automatiquement les relations temporelles complexes à partir des

Aspect	RNN	LSTM	GRU
Architecture	Simple cellule cyclique	Cellule avec 3 portes : Porte d'oubli, Porte d'entrée, Porte de sortie	Cellule avec 2 portes : Porte de réinitialisation, Porte de mise à jour
Capacité d'apprentissage	Dépendances séquentielles courtes	Dépendances à long terme grâce à la porte d'oubli	Dépendances à long terme grâce aux portes de réinitialisation et de mise à jour
Performances sur différentes tâches	Moins efficace que LSTM et GRU	Meilleure performance que RNN pour les dépendances à long terme	Performances similaires à celles des LSTM pour les dépendances à long terme
Complexité	Plus simple	Plus complexe	Intermédiaire
Efficacité	Moins efficace en raison du problème de disparition du gradient	Plus efficace que le RNN	Légèrement plus efficace que le LSTM en termes de temps de calcul

TABLE 2.4 – Comparaison des réseaux de neurones récurrents LSTM, RNN et GRU

données, ce qui peut être particulièrement avantageux lorsque les modèles sous-jacents sont non linéaires et difficiles à définir de manière explicite.

Cependant, les modèles de réseaux de neurones requièrent généralement de grandes quantités de données pour fonctionner de manière optimale et sont plus sensibles à l'overfitting. De plus, leur entraînement peut être plus complexe et chronophage que celui des modèles classiques [18].

2.5 CONCLUSION

En conclusion de ce chapitre, nous avons exploré deux approches fondamentales pour la prévision : les modèles classiques et ceux basés sur les réseaux de neurones. Nous avons mis en lumière leurs forces et leurs faiblesses, ouvrant ainsi la porte à une compréhension plus approfondie des mécanismes de prévision.

Le chapitre à venir nous emmènera plus loin dans notre exploration en utilisant ces modèles pour des prévisions concrètes. Nous plongerons dans le cœur de notre travail, en mettant en application les connaissances acquises jusqu'à présent pour anticiper avec précision les événements à venir.

CONTRIBUTIONS & APPLICATION AUX DONNÉES SRD

3

SOMMAIRE

3.1	INTRODUCTION	27
3.2	OBJECTIF DÉTAILLÉ	27
3.3	GESTION DES DONNÉES	28
3.3.1	Collection de données	28
3.3.2	Description de données	28
3.3.3	Qualité de données	29
3.3.4	Prétraitement de données	29
3.4	MÉTHODOLOGIE DE PRÉVISION	30
3.4.1	Méthodologie d'évaluation	30
3.4.2	Métriques de performance	31
3.4.3	Sélection de l'optimiseur pour l'entraînement de modèles	32
3.4.4	Décomposition des series temporelles	34
3.5	IMPLÉMENTATION DES APPROCHES	35
3.5.1	Prévision court terme	36
3.5.2	Prévision moyen terme	37
3.6	RÉSULTATS	37
3.6.1	Prévision court terme	37
3.6.2	Prévision moyen terme	38
3.7	DISCUSSION	38
3.8	SYNTHÈSE DES RÉSULTATS	41
3.9	CONCLUSION	41

3.1 INTRODUCTION

Diverses données sont récoltées sur l'environnement des parcs éoliens tels que la vitesse du vent, sa direction et la production électrique générée, données essentielles pour optimiser la performance des éoliennes, anticiper les fluctuations de production, maintenir la stabilité du réseau électrique et guider la planification ainsi que la maintenance.

Le service SRD facilite cette collecte de données via les capteurs sur les éoliennes, qui sont ensuite analysées en profondeur dans les centres de contrôle. Ce chapitre vise à clarifier notre objectif principal, centré sur la méthodologie de prévision pour anticiper les événements futurs, tout en explorant diverses méthodes d'évaluation de la performance du modèle afin d'assurer son bon fonctionnement.

3.2 OBJECTIF DÉTAILLÉ

L'énergie éolienne est sujette à des fluctuations en raison de sa dépendance aux conditions météorologiques. Les variations dans la vitesse du vent, la direction du vent, et d'autres facteurs climatiques ont un impact direct sur la puissance générée par les éoliennes, rendant la prévision de production un défi complexe. L'intégration efficace de cette énergie intermittente dans le réseau électrique présente des défis de gestion, pouvant engendrer des perturbations de tension et altérer la stabilité de l'alimentation électrique.

Pour répondre à ces enjeux, des prévisions précises de production éolienne sont essentielles pour assurer une planification optimale du réseau sur différentes échéances temporelles. Ces prévisions permettent d'anticiper les variations de production et de prendre des mesures pour maintenir la qualité de la fourniture d'électricité tout en optimisant l'exploitation du réseau.

Dans ce projet, nous cherchons à développer des modèles de prédiction avancés pour anticiper de manière précise la production éolienne de l'entreprise SRD, en se concentrant sur deux échelles temporelles distinctes : le court terme et le moyen terme. Pour la prédiction à court terme, l'objectif est de fournir des estimations de la production éolienne avec une granularité fine de 10 minutes, en anticipant les variations sur une fenêtre de prévision de 2 heures à 3 heures. Cette approche permettra d'ajuster les ressources en temps réel et d'optimiser l'utilisation de l'énergie éolienne. D'autre part, pour la prédiction à moyen terme, l'objectif est d'élaborer des prévisions de production éolienne pour une période de 2 heures, avec une fenêtre de prévision étendue jusqu'à un mois, tout en conservant un intervalle d'échantillonnage de 10 minutes.

Cette prévision à moyen terme offrira une perspective stratégique pour la gestion de la production éolienne, en permettant une planification plus efficace des ressources et une meilleure intégration des énergies renouvelables dans le réseau

électrique. En somme, ce projet vise à fournir des outils de prédiction sophistiqués pour répondre aux défis opérationnels à court terme et aux besoins de planification à moyen terme de l'entreprise SRD, contribuant ainsi à une exploitation plus efficiente et durable de son parc éolien.

3.3 GESTION DES DONNÉES

3.3.1 Collection de données

Les données utilisées dans ce projet ont été fournies par le Service SRD de Poitiers, en France. SRD est une entité spécialisée dans la recherche et le développement dans le domaine de l'énergie renouvelable. Il joue un rôle essentiel dans l'avancement des connaissances et des technologies liées à l'énergie éolienne, en mettant l'accent sur l'analyse des données et la modélisation prédictive.

Le SRD de Poitiers est réputé pour son expertise dans l'acquisition et la gestion de vastes ensembles de données liées à la production éolienne. Il collabore étroitement avec des parcs éoliens, des fournisseurs d'électricité et d'autres acteurs de l'industrie pour recueillir des informations détaillées sur les performances des éoliennes, les conditions météorologiques locales et d'autres variables pertinentes.

En plus de collecter et de gérer des données, le SRD de Poitiers se consacre à l'analyse approfondie de ces informations. Il utilise des techniques statistiques avancées, des méthodes de modélisation prédictive et des outils basés sur l'intelligence artificielle pour extraire des connaissances précieuses et fournir des recommandations pour l'optimisation de la production éolienne. En collaborant avec des chercheurs, des universités et des partenaires industriels, le SRD de Poitiers contribue activement à l'avancement des connaissances dans le domaine de l'énergie éolienne. Il participe également à des projets de recherche nationaux et internationaux visant à améliorer l'intégration des énergies renouvelables dans les systèmes énergétiques et à promouvoir une transition énergétique durable.

3.3.2 Description de données

Le SRD a mis à notre disposition une base de données comprenant un total de 157 823 observations collectées sur une période de trois ans, de 2020 à 2022. Chaque observation est associée à une date au format jour/mois/année, ainsi qu'à l'heure de la journée où la mesure a été prise, avec un intervalle de collecte de données de 10 minutes.

Plus précisément, ces données représentent les mesures de la puissance de production éolienne, exprimée en kilowatts (kW). Elles constituent une source précieuse d'informations réelles sur la production éolienne. Cette base de séries temporelles permet de suivre les variations et les tendances de la production éolienne sur plusieurs années, offrant ainsi un aperçu détaillé de l'évolution de la production d'énergie éolienne au fil du temps.

3.3.3 Qualité de données

La qualité des données se réfère à l'exactitude et à l'intégrité des données dans leur représentation des phénomènes réels sous étude. Cette notion englobe l'identification des incohérences et des problèmes présents dans les données [12] (Fan et Geerts, 2012). Elle englobe un processus en plusieurs étapes pour détecter ces incohérences :

1. **Précision des données** : cette phase concerne la fiabilité et la précision des données par rapport au phénomène observé.
2. **Repérage des données manquantes** : il s'agit d'identifier les informations absentes ainsi que leur étendue, en expliquant les raisons de leur absence.
3. **Identification des doublons** : cette étape vise à repérer les duplications au sein des différentes données.
4. **Détection des anomalies** : cette étape cherche à identifier les données aberrantes et les anomalies distinctes par rapport aux autres observations dans la série.

3.3.4 Prétraitement de données

Le prétraitement des données de séries temporelles joue un rôle crucial dans la préparation des données brutes pour une analyse et une modélisation ultérieures. Son objectif principal est de rendre les données exploitables en éliminant les obstacles potentiels et en mettant en forme les informations temporelles de manière adéquate.

Un premier aspect consiste à fusionner les colonnes Date et Time, qui sont souvent présentes séparément dans les données de séries temporelles. Cette fusion permet de faciliter la manipulation et l'indexation des données en fonction du temps.

La vérification de la stationnarité s'avère fondamentale pour plusieurs analyses. Une série temporelle est considérée comme stationnaire si ses propriétés statistiques ne fluctuent pas avec le temps. L'utilisation d'un test statistique tel que le test d'**Augmented Dickey-Fuller (ADF)** permet d'établir la stationnarité. Si la série ne satisfait pas ce critère, des méthodes de différenciation peuvent être employées pour la rendre stationnaire.

La séparation des ensembles d'entraînement et de test se révèle cruciale pour évaluer la performance des modèles d'analyse et de prédiction. Les données les plus anciennes sont généralement consacrées à la formation des modèles, tandis que les données plus récentes sont réservées aux tests. Cette approche offre l'opportunité d'apprécier la capacité des modèles à généraliser à de nouvelles données.

En ce qui concerne la détection et le filtrage des valeurs aberrantes, des étapes capitales pour préserver la qualité des données, nous avons employé le **score Z (Z-score)** comme méthode. Ceci permet de repérer les valeurs aberrantes en les comparant à leurs distances avec la moyenne.

Finalement, la normalisation des données, un volet essentiel pour garantir la cohérence et faciliter la convergence des algorithmes, a été effectuée en utilisant la mise à l'échelle **Min-Max (Min-Max Scaler)**. Cette technique réduit les données à une plage commune tout en préservant les relations entre les valeurs.

3.4 MÉTHODOLOGIE DE PRÉVISION

Dans un premier temps, une définition claire du problème a été établie, en mettant en lumière les enjeux liés à la prévision dans le contexte de l'entreprise. La compréhension approfondie de ces enjeux a permis de définir précisément les besoins en matière de prévision, en l'occurrence, la prédiction de la production éolienne pour plusieurs producteurs.

La seconde étape a consisté en l'importation des données nécessaires pour réaliser les prévisions. Les données importées comprennent principalement la production éolienne historique des différents producteurs. Ces données brutes constituent la base sur laquelle repose l'ensemble du processus de prévision.

Le prétraitement et la transformation des données ont été réalisés dans la troisième phase. Ce processus a impliqué la manipulation des données pour les rendre aptes à être utilisées dans la construction des modèles de prévision. En plus de la variable principale qu'est la production éolienne, d'autres caractéristiques pertinentes ont été ajoutées aux données, telles que le nombre de jours et le mois. Ces valeurs supplémentaires ont été soustraites de la date de chaque observation. Cette étape permet de prendre en compte des facteurs temporels additionnels et d'améliorer l'analyse des données en considérant les variations saisonnières et les tendances temporelles.

Enfin, la quatrième étape a porté sur le développement des modèles de prévision. Deux types de modèles de réseaux de neurones récurrents ont été choisis, à savoir les LSTM et les GRU. Ces modèles ont été sélectionnés en raison de leur capacité à capturer les dépendances temporelles complexes dans les séries chronologiques, comme c'est le cas avec les données de production éolienne. Les modèles ont été construits, entraînés et ajustés en utilisant les données préalablement préparées.

3.4.1 Méthodologie d'évaluation

L'étape initiale de la méthodologie d'évaluation consiste en la division des données temporelles en trois ensembles distincts : l'ensemble d'entraînement, de validation et l'ensemble de tests. L'ensemble d'entraînement est utilisé pour former notre modèle de prévision, permettant ainsi au modèle d'apprendre les schémas et les tendances présents dans les données passées. L'ensemble de tests, quant à lui, est réservé pour l'évaluation finale de la performance du modèle. Il contient des données non vues par le modèle pendant la phase d'entraînement, simulant ainsi des situations de prévision dans des conditions réelles.

Dans l'étape d'entraînement, notre modèle de prévision est exposé à l'en-

semble d'entraînement (training set). Le modèle explore les données passées et cherche à identifier les motifs récurrents, les saisons, les tendances et les éventuelles irrégularités présentes dans la série temporelle. Différents algorithmes de prévision peuvent être appliqués, tels que les méthodes ARIMA ou les réseaux de neurones récurrents (RNN).

Après la formation du modèle sur l'ensemble d'entraînement, il est évalué sur l'ensemble de validation pour déterminer sa capacité à généraliser avec de nouvelles données. Dans cette phase d'évaluation, le modèle génère des prédictions en utilisant les données d'entrée de l'ensemble de validation et compare ces prédictions aux cibles réelles correspondantes. Les métriques d'évaluation telles que l'erreur moyenne absolue (MAE) et l'erreur quadratique moyenne (RMSE) sont calculées pour quantifier à quel point les prédictions du modèle correspondent aux valeurs réelles. Cette étape permet de valider l'efficacité du modèle dans des situations de prévision réelles et d'assurer sa fiabilité pour des applications futures.

Une fois que le modèle est entraîné et que ses hyperparamètres sont optimisés, il est temps de l'évaluer sur l'ensemble de tests. Ce jeu de données non vu auparavant simule les conditions réelles de prévision. Les prédictions générées par le modèle sont comparées aux valeurs réelles correspondantes dans l'ensemble de tests. Les métriques d'évaluation telles que l'erreur moyenne absolue (MAE) et l'erreur quadratique moyenne (RMSE) sont calculées pour quantifier la précision du modèle.

3.4.2 Métriques de performance

Pour évaluer la performance des modèles de prédiction des séries temporelles, il existe plusieurs mesures d'évaluation clés. Parmi celles-ci, nous allons examiner en détail deux des plus couramment utilisées : l'erreur absolue moyenne (MAE) et l'erreur quadratique moyenne (RMSE), ainsi que d'autres métriques importantes[13].

3.4.2.1 Erreur Absolue Moyenne (MAE)

Le MAE est une mesure de la différence moyenne entre les valeurs prédites et les valeurs réelles. Cette métrique est calculée en prenant la moyenne des valeurs absolues des écarts entre les prédictions et les observations réelles. Mathématiquement, le MAE est donné par :

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|$$

où :

- n est le nombre d'observations.
- y_i est la valeur réelle de l'observation i .
- \hat{y}_i est la valeur prédite pour l'observation i .

Le MAE est simple à interpréter car il représente directement l'erreur moyenne entre les prédictions et les observations réelles.

3.4.2.2 Erreur Quadratique Moyenne (MSE) :

Le MSE est une mesure de l'erreur moyenne entre les valeurs prédites et les valeurs réelles dans un modèle de régression. Il est calculé en prenant la moyenne des carrés des écarts entre les valeurs prédites et les valeurs réelles. La formule mathématique est la suivante :

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

L'objectif du MSE est de mesurer la dispersion des erreurs entre les valeurs prédites et les valeurs réelles. Plus le MSE est bas, plus les prédictions du modèle sont proches des valeurs réelles.

3.4.2.3 Racine de l'Erreur Quadratique Moyenne (RMSE) :

Le RMSE est une mesure qui pénalise davantage les grandes erreurs par rapport aux petites erreurs, en élevant les écarts au carré avant de calculer la moyenne. Mathématiquement, le RMSE est donné par :

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}$$

Le RMSE a tendance à être plus sensible aux erreurs importantes car les écarts sont élevés au carré. Cela signifie qu'une grande erreur aura un impact plus important sur le RMSE que sur le MAE. Cependant, comme il inclut une opération de racine carrée, le RMSE peut également être sensible aux valeurs aberrantes.

3.4.3 Sélection de l'optimiseur pour l'entraînement de modèles

L'optimiseur est un élément essentiel de la phase d'entraînement de notre modèle de prévision de séries temporelles. Il intervient dans le processus d'ajustement des poids et des biais du modèle afin de minimiser l'erreur entre les prédictions générées et les valeurs réelles de la série temporelle. Le choix de

l'optimiseur approprié a un impact significatif sur la vitesse de convergence de l'apprentissage et la qualité des prédictions ultérieures.

Le choix de l'optimiseur dépend souvent de la nature du modèle et des caractéristiques des données. Pour les modèles de séries temporelles, des optimiseurs couramment utilisés incluent l'optimiseur stochastique par descente de gradient (SGD) et l'optimiseur adaptatif momentané (ADAM). Chaque optimiseur possède des propriétés uniques qui influent sur la vitesse de convergence, la gestion des gradients et l'adaptabilité aux variations de la fonction de coût.

Les optimiseurs comportent également des hyperparamètres qui nécessitent une configuration appropriée. Par exemple, le taux d'apprentissage (learning rate) détermine la taille des pas effectués dans la direction du gradient. Un taux d'apprentissage trop élevé peut entraîner des oscillations et un apprentissage instable, tandis qu'un taux d'apprentissage trop faible peut ralentir la convergence.

Le **Gradient Descent** est l'un des optimiseurs les plus fondamentaux et largement utilisés dans l'apprentissage automatique. Il s'agit d'une méthode itérative qui ajuste les paramètres du modèle dans la direction opposée du gradient de la fonction de perte. Bien que le Gradient Descent puisse être efficace pour la convergence, il peut lentement converger ou se retrouver piégé dans des minimums locaux.

Dans **SGD**, au lieu de mettre à jour les paramètres du modèle en utilisant l'ensemble complet des données d'entraînement, les mises à jour sont effectuées sur un seul exemple d'entraînement à la fois. L'avantage principal de cette méthode dans le contexte des séries temporelles est sa capacité à gérer des ensembles de données massifs en effectuant des mises à jour basées sur un seul exemple à la fois, ce qui peut économiser des ressources computationnelles [3].

ADAM [24] adapte le taux d'apprentissage pour chaque paramètre en utilisant à la fois le moment (première moyenne mobile des gradients) et le moment du gradient (second moment mobile des gradients). Il ajuste automatiquement le taux d'apprentissage en fonction de l'observation des gradients passés, ce qui le rend efficace pour différentes tâches. Des études antérieures ont montré que l'optimiseur ADAM [24] est plus efficace que les méthodes d'optimisation standard, telles que la descente de gradient stochastique (SGD), en termes de convergence et de temps d'entraînement.

Dans le cadre de notre approche novatrice, nous avons opté pour l'utilisation de l'optimiseur ADAM pour l'entraînement de notre modèle.

L'évaluation expérimentale a été effectuée dans des conditions réelles d'application en utilisant les ressources du laboratoire. Initialement, nous avons utilisé les machines mises à disposition par le laboratoire, notamment une machine virtuelle hébergée sur leur serveur, équipée d'un GPU dédié. Cette ressource nous a permis d'effectuer des calculs intensifs nécessaires à notre entraînement initial.

Par la suite, nous avons poursuivi notre travail sur nos propres machines personnelles, en utilisant les données et les modèles préalablement préparés sur la

machine du laboratoire. Les spécifications de notre propre matériel sont également mentionnées ci-dessous pour une transparence totale dans la description de notre méthodologie expérimentale.

Paramètres	Spécifications
Processeur	Intel Core i3-7020U
Fréquence du processeur	2,30 GHz
Nombre de cœurs	2
Mémoire RAM	12 Go
Système d'exploitation	Windows 10

TABLE 3.1 – Spécifications de l'ordinateur utilisé pour l'évaluation expérimentale.

3.4.4 Décomposition des séries temporelles

Les séries temporelles possèdent une caractéristique spéciale qui les distingue des autres types de données : elles sont composées d'observations qui sont enregistrées dans un ordre chronologique précis. Cette chronologie joue un rôle crucial dans l'analyse et la prévision des séries temporelles. Contrairement aux ensembles de données classiques, où les caractéristiques (features) et les cibles sont souvent distinctes et indépendantes, les séries temporelles sont dépendantes du temps, ce qui signifie que chaque observation peut influencer la suivante. Cette dépendance temporelle rend nécessaire le respect de l'ordre chronologique lors de la prévision, car les motifs et les tendances historiques sont essentiels pour anticiper les futurs mouvements [29].

Lorsque nous traitons des séries temporelles, il est souvent utile de découper la série en sous-séquences afin de capturer les motifs locaux et de mieux saisir les tendances à différentes échelles. Par exemple, prenons une série temporelle de 14 observations (Voir la figure 3.1). Si nous découpons cette série en sous-séquences de taille 4, nous obtiendrions 7 ensembles et chaque ensemble génère une seule sortie (Voir la figure 3.2).

Temps	t1	t2	t3	t4	t5	t6	t7	t8	t9	t10	t11	t12	t13	t14
Observations	54	102	89	36	77	12	3	290	125	854	623	425	320	8

FIGURE 3.1 – Exemple d'une série temporelle

Pour exploiter efficacement les séries temporelles dans notre processus d'entraînement et de test des modèles, nous avons adopté une approche de décomposition bien définie. Notre stratégie reposait sur l'utilisation de lots (ou "batches") pour organiser les données temporelles. Chaque lot contenait 32 séquences distinctes, et chaque séquence était constituée de 16 observations historiques. Cette division en lots nous a permis d'exploiter au mieux la capacité de calcul de nos modèles, en traitant simultanément plusieurs séquences tout en préservant la structure temporelle des données.

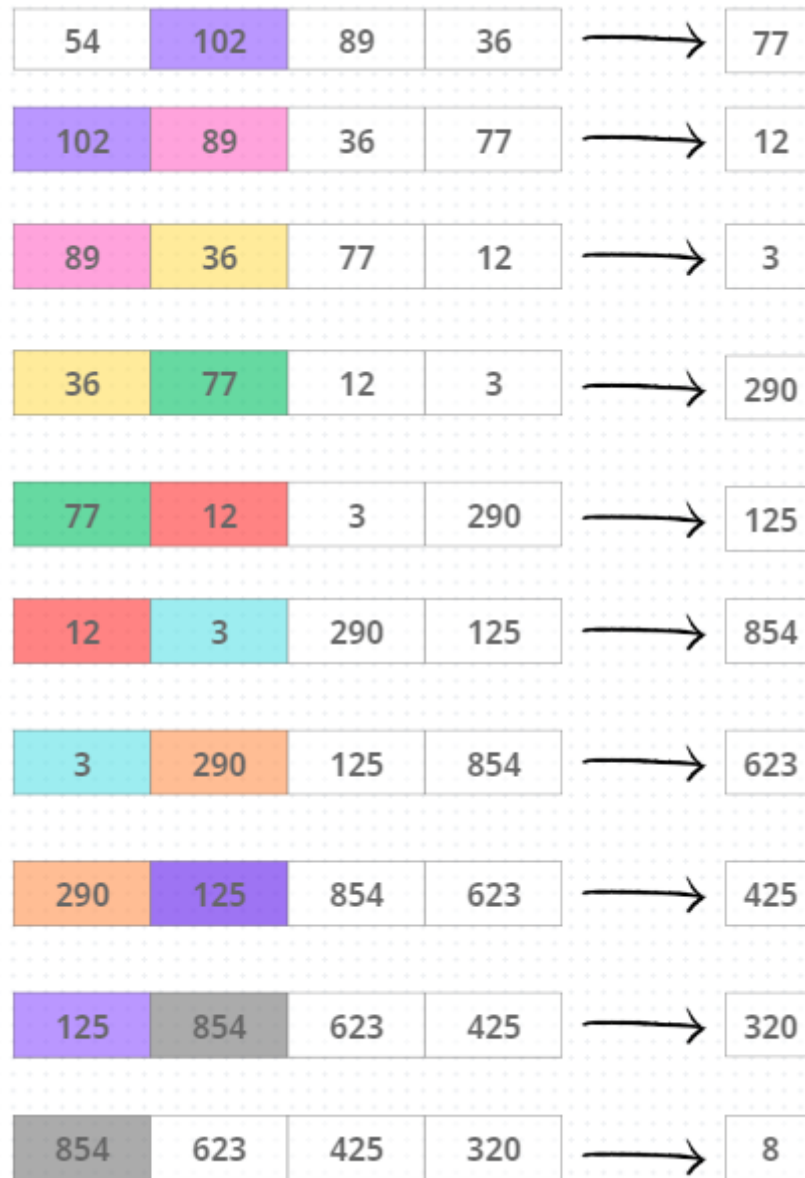


FIGURE 3.2 – Les sous-séquences d'une série après décomposition.

3.5 IMPLÉMENTATION DES APPROCHES

Pour exploiter au mieux les données, il est crucial de recourir à des outils et des technologies efficaces. C'est ici que Python et PyTorch entrent en jeu.

Python, un langage de programmation versatile et largement adopté, offre un environnement propice à l'exploration, à la manipulation et à l'analyse des séries temporelles. Doté d'une syntaxe claire et intuitive, Python facilite le traitement des données et la visualisation des tendances temporelles, tout en offrant un vaste écosystème de bibliothèques et de frameworks adaptés à ces tâches spécifiques. Les bibliothèques telles que NumPy et Pandas permettent de manipuler les don-

nées temporelles avec aisance, tandis que Matplotlib et Seaborn permettent de générer des visualisations informatives[28].

Lorsque la modélisation des séries temporelles exige des techniques d'apprentissage automatique plus avancées, PyTorch se révèle être un atout majeur. PyTorch est une bibliothèque d'apprentissage profond qui propose une approche basée sur le calcul différentiable. Cela signifie que la création et l'entraînement de modèles de prédictions deviennent plus flexibles, car PyTorch automatise le calcul des gradients nécessaires pour optimiser les paramètres du modèle[20].

3.5.1 Prévision court terme

Dans notre approche d'implémentation pour la prévision à court terme, nous avons adopté une approche comparative en mettant en œuvre trois modèles différents : LSTM, GRU et ARIMA. Notre objectif était de déterminer lequel de ces modèles serait le plus performant pour la tâche de prévision. Pour ajuster le modèle ARIMA de manière optimale, nous avons mis en place une méthodologie rigoureuse consistant à parcourir systématiquement une grille de recherche. Cette grille de recherche a été conçue pour explorer différentes combinaisons de valeurs des paramètres d'ordre (p , d , q) afin d'identifier les paramètres les plus appropriés pour notre modèle.

Après avoir traité le modèle ARIMA et ses paramètres, nous nous sommes tournés vers la mise en place du modèle LSTM pour la prévision à court terme. Pour ce faire, nous avons d'abord implémenté une classe LSTMModel conçue en tant que sous-classe de nn.Module. Cette base est fournie par PyTorch et permet la création de modèles personnalisés. Le modèle LSTM a été spécifiquement conçu pour traiter des séquences en entrée et générer des prédictions en fonction de ces séquences.

Ensuite, nous avons choisi de personnaliser l'architecture du modèle LSTM en définissant divers hyperparamètres. Ces hyperparamètres incluent la taille d'entrée (input-size), la taille cachée (hidden-size), le nombre de couches (num-layers) et la taille de sortie (output-size). En ajustant ces paramètres, nous avons pu configurer le modèle de manière optimale en tenant compte des caractéristiques spécifiques de nos données et des performances que nous recherchons.

Parallèlement, nous avons suivi une méthodologie similaire lors de la mise en place du modèle GRU. En personnalisant également les hyperparamètres spécifiques au modèle GRU, nous avons été en mesure de développer une architecture spécialisée (Voir la figure 3.3). Cette approche a grandement contribué à l'obtention de performances favorables dans nos tâches de prévision à court terme, consolidant ainsi notre démarche méthodique et notre expertise dans l'utilisation de réseaux neuronaux pour de telles applications qui comporte trois couches : une couche d'entrée (input), une couche cachée (hidden) et une couche de sortie (output). Dans ce réseau, les entrées (x) sont pondérées par des poids (w), puis soumises à une fonction d'activation (σ) pour produire une sortie (a).

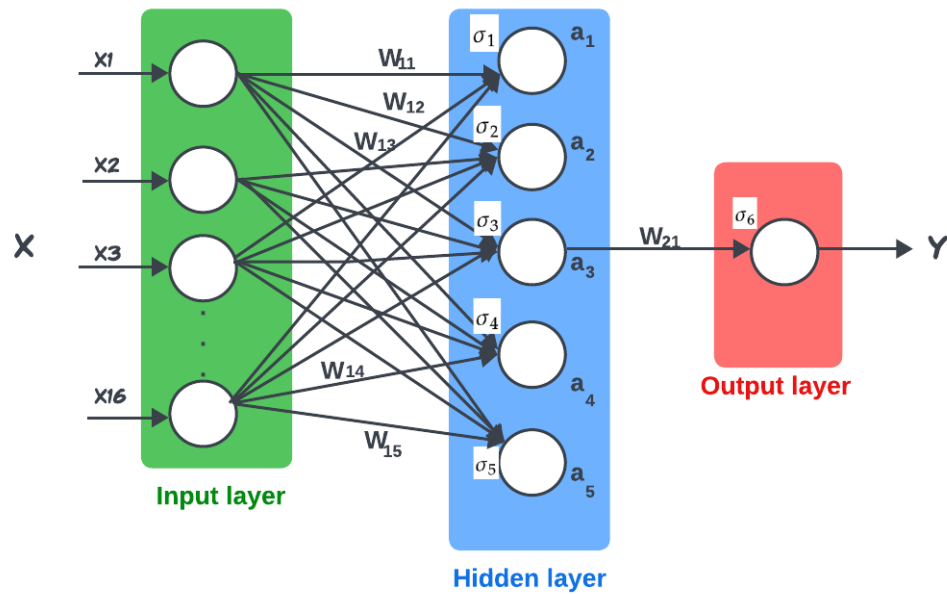


FIGURE 3.3 – Schéma représentant l'architecture des modèles LSTM et GRU.

3.5.2 Prévision moyen terme

Nous avons élargi notre objectif pour anticiper les événements futurs avec une fenêtre de prévision de deux heures. Pour atteindre cette anticipation, nous avons mis en place la classe `WindowGenerator` qui agit comme un outil de prétraitement des données et de gestion des fenêtres temporelles pour l'entraînement du modèle.

La classe `WindowGenerator` est conçue pour manipuler et organiser les données chronologiques nécessaires à l'entraînement du modèle LSTM. En substance, elle prépare les séquences d'entrée et de sortie en fonction des paramètres tels que la largeur d'entrée (`input_width`), la largeur de sortie (`label_width`) et le décalage (`shift`). Ces paramètres définissent la taille des fenêtres temporelles que le modèle examinera pour la prédiction.

Ce processus de prétraitement est fondamental pour garantir que le modèle LSTM puisse apprendre les motifs temporels et effectuer des prévisions précises à moyen terme.

3.6 RÉSULTATS

3.6.1 Prévision court terme

Dans cette partie, nous exposons les divers résultats générés en mettant en œuvre la méthode de prédiction et d'évaluation décrite précédemment, en utilisant les modèles LSTM et GRU, qui ont donné des résultats significativement meilleurs par rapport au modèle ARIMA.

Les résultats de notre modèle de prédiction des séries temporelles avec les modèles LSTM et GRU se révèlent prometteurs. Lors de l'évaluation sur l'ensemble de test, ces modèles ont démontré une capacité significative à capturer les tendances et les motifs sous-jacents des données temporelles. Les valeurs de l'erreur quadratique moyenne (RMSE) obtenues indiquent une précision satisfaisante par rapport à l'amplitude des valeurs cibles.

Les courbes de prédiction des modèles LSTM et GRU se superposent étroitement aux séries temporelles réelles, illustrant leur capacité à suivre les fluctuations à court terme. En revanche, le modèle ARIMA n'a pas produit des résultats favorables en comparaison (Voir la figure 3.5).

3.6.2 Prévision moyen terme

Dans le cas de la prévision long terme, nous avons testé un algorithme LSTM implémenté en Python avec la bibliothèque PyTorch, en utilisant également la librairie Statsmodels pour la prédiction de séries temporelles. L'objectif principal de cette analyse était de réaliser des prédictions à long terme sur la production éolienne en exploitant la classe Windowgenerator, un composant clé de la méthodologie utilisée.

Les résultats obtenus ont été évalués à l'aide de métriques appropriées telles que l'erreur quadratique moyenne (RMSE) et l'erreur absolue moyenne (MAE). Ces métriques ont permis d'évaluer les performances du modèle en termes de précision de la prédiction et de capacité à capturer les tendances générales de la production éolienne.

3.7 DISCUSSION

Au cours de cette étude, nous avons élaboré une approche pour anticiper la quantité d'énergie éolienne produite par le réseau électrique de SRD. Pour ce faire, nous avons employé diverses techniques comprenant des méthodes simples, des approches statistiques et des méthodes d'apprentissage automatique. Nous avons exploité les données disponibles provenant des courbes de production émanant des producteurs de taille moyenne et grande. Ces méthodes ont été évaluées pour prédire la production à des intervalles de 10 minutes.

Afin de faciliter une discussion approfondie des résultats, nous avons regroupé les producteurs en familles en fonction de leurs puissances respectives (Voir la table 3.2 et la figure 3.4).

Dans le cas de prévisions court terme, les valeurs de l'erreur quadratique moyenne (RMSE) fournissent une mesure quantitative de l'exactitude des prévisions par rapport aux données réelles. Après la phase de test, nous avons observé que les résultats obtenus par le modèle GRU présentent des valeurs de RMSE plus faibles que celles du modèle LSTM et ARIMA (Voir la table 3.3). Cette diminution de l'erreur suggère que le modèle GRU a réussi à capturer plus

Famille de producteurs	Producteurs de chaque famille	Puissance de producteur(en Kw)
F1 (Faible puissance)	P0005	318
	P0013	390
	P0018	400
F2 (Puissance moyenne)	P0010	630
	P0012	630
	P0016	630
	P0017	630
	P0015	800
F3 (Haute puissance)	P0049	1250
	P0060	2000
	P0054	2106
	P0014	3000
	P0058	4000
	P0052	5490
	P0061	6950
	P0063	80007

TABLE 3.2 – Répartition des producteurs par puissance

efficacement les relations temporelles dans les données et à fournir des prévisions plus précises.

En examinant les graphes comparant les valeurs réelles aux valeurs prédites par chaque modèle, nous pouvons observer que les courbes générées par les modèles LSTM et GRU sont en général conformes aux tendances et variations des données réelles (Voir la figure 3.5). Cela suggère que les deux modèles ont réussi à saisir les motifs sous-jacents dans les données de manière efficace. En revanche, il est important de noter que le modèle ARIMA ne semble pas produire des résultats satisfaisants (Voir la figure 3.5), car les courbes de valeurs réelles et prédites présentent une dispersion significative, ce qui indique une performance moins précise sur la modélisation des données.

Outre les performances prédictives, le temps d'apprentissage et le temps d'exécution pour effectuer les prédictions sont des facteurs importants à prendre en considération. Nous avons noté que le modèle GRU a nécessité davantage de temps d'exécution par rapport au modèle LSTM et ARIMA (Voir la table 3.4). Cette augmentation du temps peut être due aux structures de calcul plus complexes dans le modèle GRU, notamment en raison de ses portes de mise à jour et de réinitialisation. Bien que le temps d'exécution accru puisse être un inconvé-

Famille de producteurs	Modèle LSTM	Modèle GRU	Modèle ARIMA
F1	0.26 ± 0.006	0.0688 ± 0.0016	80.54 ± 9.996
F2	0.28 ± 0.0143	0.0717 ± 0.0074	159.164 ± 45.223
F3	0.27 ± 0.0142	0.0706 ± 0.002	1380.53 ± 883.424
Classement moyen	2	1	3

TABLE 3.3 – Comparaison des performances moyennes (RMSE) des modèles LSTM, GRU et ARIMA pour différentes familles de producteurs

nient, il est important de considérer ce compromis entre la précision et l'efficacité temporelle en fonction des besoins spécifiques de l'application.

Famille de producteurs	Modèle LSTM	Modèle GRU	Modèle ARIMA
F1	29.66	35.66	14.33
F2	30.8	36	16.2
F3	30.875	36.25	16.375
Classement moyen	2	3	1

TABLE 3.4 – Comparaison des Temps d'Exécution Moyens en minutes entre les Modèles LSTM, GRU et Modèle ARIMA pour Différentes Familles de Producteurs

Pour effectuer des prévisions à moyen terme sur une période de 2 heures, nous avons utilisé un modèle LSTM. Lors de l'évaluation des performances de notre modèle, nous avons observé des valeurs significativement élevées pour le RMSE et le MAE indiquant que les prévisions présentent des écarts importants par rapport aux valeurs réelles. Ces résultats soulignent la difficulté de notre modèle à capturer avec précision les variations temporelles des données à moyen terme.

En examinant les graphiques de comparaison entre les valeurs réelles et prédites, il est clair que les deux séries de données présentent un écart substantiel. Les valeurs réelles et prédites divergent considérablement, ce qui suggère que le modèle LSTM ne parvient pas à saisir les nuances subtiles des changements dans les données. Les pics et les creux des données réelles ne sont pas correctement anticipés par le modèle, ce qui se traduit par des prévisions moins précises.

3.8 SYNTHÈSE DES RÉSULTATS

En résumé, notre analyse comparative des modèles LSTM et GRU pour les prévisions à court terme révèle que le modèle GRU présente des performances plus favorables en termes de valeurs de RMSE par rapport au modèle LSTM. Les graphiques de valeurs réelles et prédites, bien que présentant des similitudes visuelles, n'ont pas suffi à compenser l'écart de précision entre les deux modèles.

Toutefois, il est crucial de noter que cette amélioration de la précision du modèle GRU a été accompagnée d'un temps d'exécution plus long en raison de sa complexité accrue. Ainsi, le choix entre les modèles LSTM et GRU devra être basé sur un équilibre entre la précision des prévisions et l'efficacité du temps d'exécution, en fonction des besoins spécifiques de l'application envisagée.

3.9 CONCLUSION

Dans ce dernier chapitre, nous avons parcouru un voyage passionnant vers la réalisation de notre objectif. Tout d'abord, nous avons minutieusement détaillé notre objectif, établissant ainsi les bases solides nécessaires pour la suite de notre travail. Ensuite, nous avons exposé les différentes étapes cruciales que nous avons entreprises pour prétraiter nos données, garantissant ainsi leur qualité et leur pertinence pour nos expériences.

Une fois nos données prêtes, nous avons consacré du temps à expliquer comment nous avons évalué nos modèles. Pour mesurer leur performance, nous avons utilisé des métriques éprouvées, fournissant une évaluation objective de leur efficacité. De plus, nous avons choisi avec soin un optimiseur adapté pour améliorer les performances de nos modèles.

Après nous avons mis en œuvre différentes approches pour la prévision à court et moyen terme, en fournissant une description détaillée de chaque méthode. Les résultats obtenus ont été présentés de manière transparente pour démontrer la rigueur de notre méthodologie. Ensuite, nous avons engagé une discussion approfondie pour examiner ces résultats en détail, ce qui nous a permis de mieux comprendre les implications de nos découvertes et les enseignements à en tirer. Dans l'ensemble, ce chapitre marque une étape cruciale de notre parcours, nous rapprochant davantage de la réalisation de notre objectif, tout en clarifiant les choix stratégiques que nous avons effectués tout au long de cette aventure.

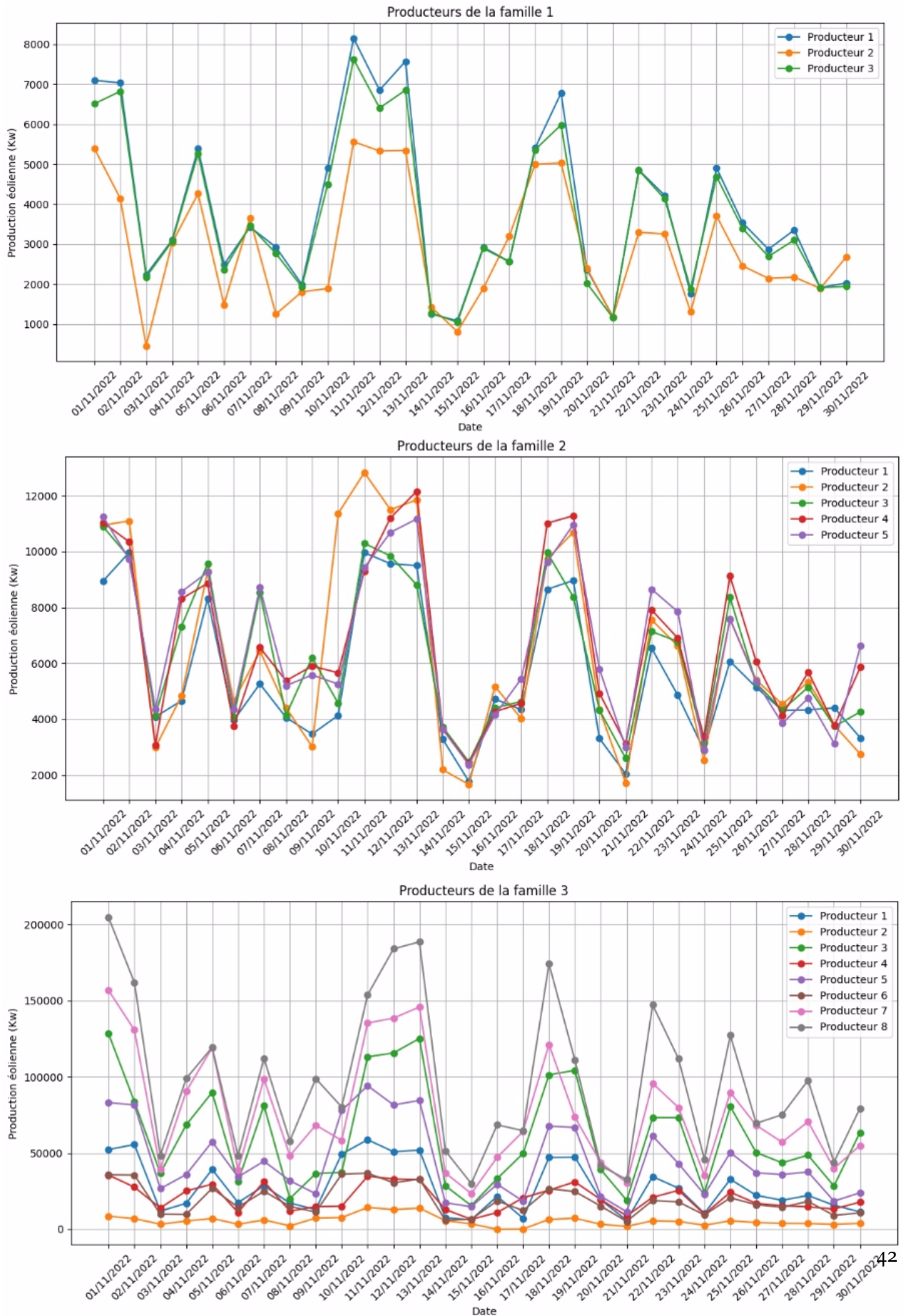


FIGURE 3.4 – Courbes qui représentent la production éolienne de chaque famille de producteurs.

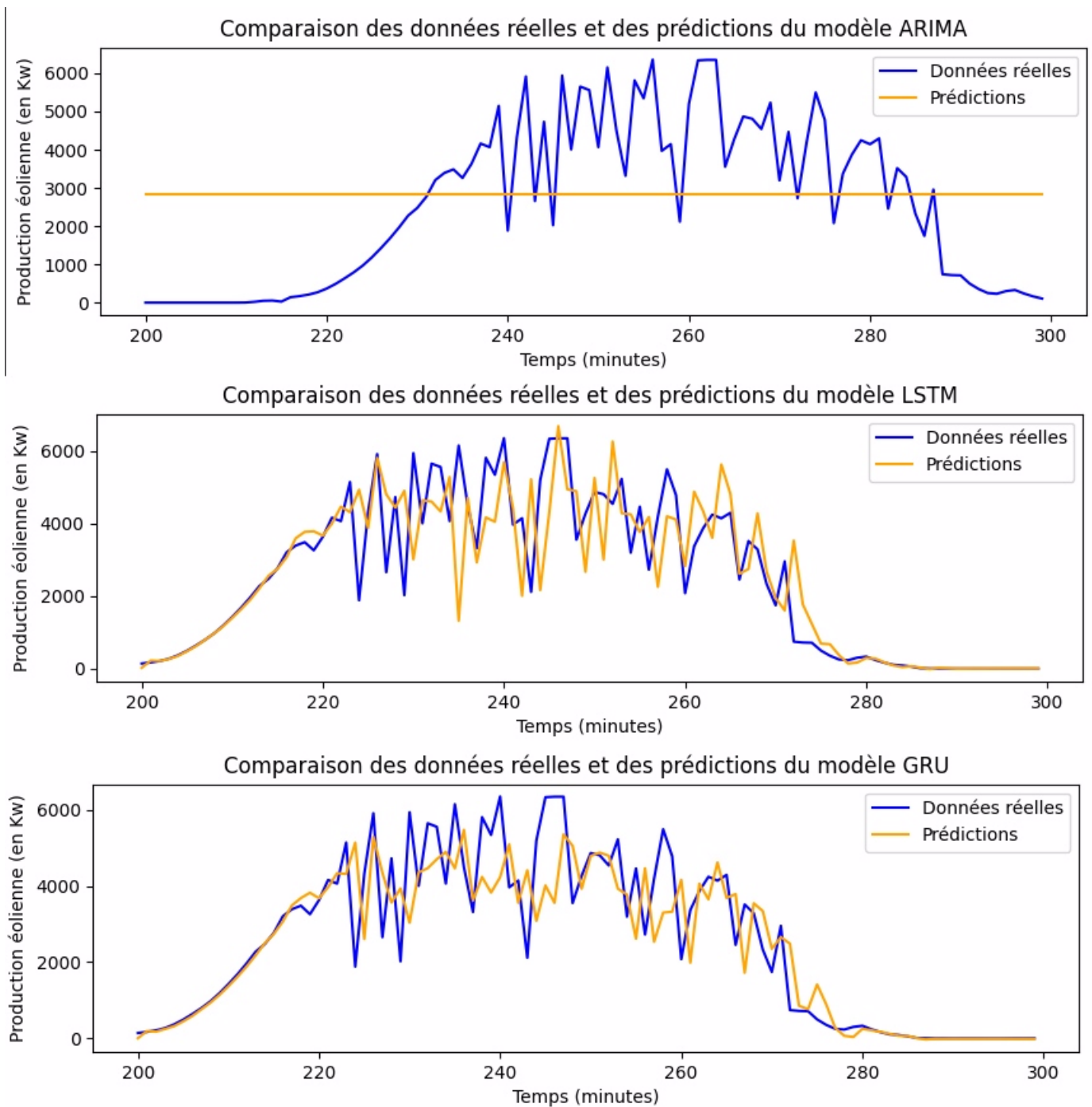


FIGURE 3.5 – Comparaison des données réelles et des prédictions des modèles ARIMA, LSTM et GRU : Un Aperçu de la Performance sur 1% des Données.

CONCLUSION & PERSPECTIVES

«Il est facile de manquer le but et difficile de l'atteindre »

Aristote

Ce mémoire clôture un parcours passionnant qui s'inscrit au cœur d'un partenariat collaboratif entre le gestionnaire de réseau de distribution de l'électricité SRD et le laboratoire LIAS. Notre quête au sein de ce projet ambitieux a été de repousser les frontières de la prévision en exploitant pleinement les potentiels de l'intelligence artificielle.

L'objectif pivot de cette initiative résidait dans la mise au point de techniques novatrices dédiées à l'anticipation de la production éolienne au sein du réseau SRD. En disposant d'un accès privilégié aux données historiques englobant la période triennale de 2020 à 2022, généreusement mises à notre disposition par SRD, nous avons été en mesure de concevoir des modèles de prédiction conçus pour esquisser les contours de la production à venir. Les avancées significatives dans le domaine de l'apprentissage automatique ont grandement renforcé la pertinence et la précision des prédictions obtenues.

Le cheminement parcouru a été jalonné d'étapes cruciales, chacune contribuant à consolider notre démarche méthodologique. Le prétraitement initial des données a émergé tel un pilier fondamental. Cette phase préliminaire a nécessité une rigueur particulière, incluant l'élimination des valeurs aberrantes, la normalisation des données et la détection d'éventuelles tendances saisonnières sous-jacentes. Cette étape a établi les bases essentielles pour la qualité et la fiabilité des étapes ultérieures.

Toutefois, ce mémoire représente davantage un commencement qu'une conclusion. L'horizon des prévisions à long terme, s'étirant jusqu'à deux mois et au-delà, se dresse encore comme un défi éminemment stimulant. Néanmoins, ce projet a établi les bases nécessaires pour aborder ces défis avec assurance. De plus, il convient de noter que l'ajout de données exogènes, telles que les données météorologiques, offre une perspective prometteuse pour l'amélioration des prévisions à moyen terme. L'intégration de ces données externes permet de capturer plus précisément l'impact des conditions météorologiques sur les variables prévues, renforçant ainsi la fiabilité et la précision des résultats obtenus. Cela ouvre la voie à des opportunités passionnantes pour affiner les modèles existants et améliorer considérablement les performances des prévisions à moyen terme.

BIBLIOGRAPHIE

- [1] Axel BARTHELMIES. *Les énergies renouvelables : Éolienne, photovoltaïque, biomasse, géothermie, hydroélectricité*. EDP Sciences, 2023.
- [2] Christopher M BISHOP. *Pattern recognition and machine learning*. Springer Science & Business Media, 2006.
- [3] Léon BOTTOU. “Stochastic gradient descent tricks”. In : *Neural networks : Tricks of the trade*. Springer, 2012, p. 431-448.
- [4] G. E. P. BOX, G. M. JENKINS et G. C. REINSEL. *Time series analysis : forecasting and control*. 5th. Wiley, 2016.
- [5] Jason BROWNLEE. *A gentle introduction to the Box-Jenkins method for time series forecasting*. en-US. URL : <https://machinelearningmastery.com/gentle-introduction-box-jenkins-method-time-series-forecasting/> (visité le 20/06/2023).
- [6] Nagesh Singh CHAUHAN. *Introduction to RNN and LSTM*. en-US. URL : <https://www.theaidream.com/post/introduction-to-rnn-and-lstm> (visité le 04/07/2023).
- [7] Kyunghyun CHO et al. *GRU : An Efficient Neural Network for Language Modeling*. arXiv preprint arXiv :1406.1078, 2014.
- [8] Didier DELIGNIÈRES. *Analyse statistique des séries temporelles*. Dunod, 2010.
- [9] Saul DOBILAS. *GRU Recurrent Neural Networks — A Smart Way to Predict Sequences in Python*. Medium. 24 fév. 2022. URL : <https://towardsdatascience.com/gru-recurrent-neural-networks-a-smart-way-to-predict-sequences-in-python-80864e4fe9f6> (visité le 21/08/2023).
- [10] Ette Harrison ETUK et Nathaniel OJEKUDO. “Subset SARIMA modelling : An alternative definition and a case study”. In : *British Journal of Mathematics & Computer Science* 5.4 (2015), p. 538.
- [11] EUROSTAT. *Emissions de gaz à effet de serre par secteur et par pays de l’UE, 1990-2022*. 2023. URL : <https://www.toutteleurope.eu/environnement/union-europeenne-chine-etats-unis-qui-emet-le-plus-de-gaz-a-effet-de-serre/#:~:text=L'Union%20europ%C3%A9enne%20s'%C3%A9tait,UE%20de%20respecter%20ses%20engagements>.
- [12] Wenfei FAN et Floris GEERTS. *Foundations of Data Quality Management*. T. 4. Synthesis Lectures on Data Management. Morgan & Claypool Publishers, 2012, p. 1-217.
- [13] Aurélien GERON. *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow : Concepts, tools, and techniques to build intelligent systems*. O’Reilly Media, Inc., 2017.

- [14] Pierre Y. GLOAGUEN et Alain GALLIÉ. *Prévision des séries temporelles*. Dunod, 2022, p. 352.
- [15] Ian GOODFELLOW et al. "Deep learning". In : *The MIT Press 2.2* (2016), p. 800.
- [16] *Guide illustré des LSTM et GRU : une explication étape par étape*. fr. URL : <https://ichi.pro/fr/guide-illustre-des-lstm-et-gru-une-explication-etape-par-etape-75771469479713> (visité le 12/07/2023).
- [17] Dishashree26 GUPTA. *Fundamentals of Deep Learning – Introduction to Recurrent Neural Networks*. Analytics Vidhya. 7 déc. 2017. URL : <https://www.analyticsvidhya.com/blog/2017/12/introduction-to-recurrent-neural-networks/> (visité le 21/08/2023).
- [18] Rob J. HYNDMAN et George ATHANASOPOULOS. *Forecasting : Principles and Practice*. O'Reilly Media, 2018.
- [19] Rob J. HYNDMAN et George ATHANASOPOULOS. *Forecasting : principles and practice*. 3rd. OTexts, 2018. URL : <https://otexts.com/fpp3/>.
- [20] Sagar IMAMBI, Kolla Bhanu PRAKASH et GR KANAGACHIDAMBARESAN. "Py-Torch". In : *Programming with TensorFlow : Solution for Edge Computing Applications* (2021), p. 87-104.
- [21] *index - Laboratoire d'Informatique et d'Automatique pour les Systèmes*. URL : <https://univ-poitiers.hal.science/LIAS> (visité le 24/09/2023).
- [22] Fazle KARIM et al. "LSTM fully convolutional networks for time series classification". In : *IEEE access 6* (2017), p. 1662-1669.
- [23] Eun-Ji KIM, Taek-Kie LEE et Kyu-Ho KIM. "A Study on Prediction of Wind Power Based on Deep-Learning Using Weather Data". In : *The transactions of The Korean Institute of Electrical Engineers 70.5* (31 mai 2021), p. 735-741. ISSN : 1975-8359, 2287-4364. DOI : [10.5370/KIEE.2021.70.5.735](https://doi.org/10.5370/KIEE.2021.70.5.735). URL : <http://www.dbpia.co.kr/Journal/ArticleDetail/NODE10553666> (visité le 21/08/2023).
- [24] Diederik P. KINGMA et Jimmy BA. "Adam : A Method for Stochastic Optimization". In : *arXiv preprint arXiv :1412.6980* (2014).
- [25] Mostafa MAJIDPOUR et al. "Fast univariate time series prediction of solar power for real-time control of energy storage system". In : *Forecasting 1.1* (2018), p. 107-120.
- [26] Paul NEWBOLD. "ARIMA model building and the time series analysis approach to forecasting". In : *Journal of forecasting 2.1* (1983), p. 23-35.
- [27] *Recurrent Neural Network (RNN)*. URL : <https://machine-learning.paperspace.com/wiki/recurrent-neural-network-rnn> (visité le 21/08/2023).
- [28] Koki SAITOH. *Deep learning from the basics : Python and deep learning : Theory and implementation*. Packt Publishing Ltd, 2021.
- [29] Thomas J. SARGENT et John STACHURSKI. *Time Series Analysis : Forecasting and Control*. MIT Press, 2013.
- [30] Ramadass SATHYA, Annamma ABRAHAM et al. "Comparison of supervised and unsupervised learning algorithms for pattern classification". In : *International Journal of Advanced Research in Artificial Intelligence 2.2* (2013), p. 34-38.

-
- [31] Shipra SAXENA. *Introduction to Gated Recurrent Unit (GRU)*. en. URL : <https://www.analyticsvidhya.com/blog/2021/03/introduction-to-gated-recurrent-unit-gru/> (visité le 18/07/2023).
- [32] Andrea L SCHAFFER, Timothy A DOBBINS et Sallie-Anne PEARSON. "Interrupted time series analysis using autoregressive integrated moving average (ARIMA) models : a guide for evaluating large-scale health interventions". In : *BMC medical research methodology* 21.1 (2021), p. 1-12.
- [33] SRD. URL : <https://fr.kompass.com/c/srd/fr5289367/> (visité le 17/08/2023).
- [34] SRD (Energies Vienne) | LinkedIn. URL : <https://fr.linkedin.com/company/srd-groupe-energies-vienne> (visité le 24/09/2023).
- [35] TensorFlow | Types of RNN - Javatpoint. URL : <https://www.javatpoint.com/tensorflow-types-of-rnn> (visité le 21/08/2023).
- [36] José F TORRES et al. "Deep learning for time series forecasting : a survey". In : *Big Data* 9.1 (2021), p. 3-21.
- [37] *What are recurrent neural networks*. en-US. URL : <https://www.ibm.com/topics/recurrent-neural-networks> (visité le 04/07/2023).

Résumé : La société SRD se démarque par son réseau de distribution d'électricité hautement technologique, axé sur l'efficacité, l'intégration des énergies renouvelables et la fiabilité de l'énergie fournie. En partenariat avec le laboratoire LIAS, SRD accède aux dernières avancées en modélisation éolienne et en apprentissage automatique, renforçant ainsi la crédibilité de ses projets et favorisant un échange de connaissances fructueux entre l'industrie et le milieu académique. Notre travail consiste à développer des modèles de prévision avancée pour la production éolienne en utilisant des techniques de traitement des données temporelles, notamment les séries temporelles. Ces modèles sont conçus pour anticiper avec précision la production éolienne sur deux échelles temporelles spécifiques : à court terme (2 heures) pour la gestion en temps réel de l'équilibre production-demande d'électricité, et à moyen terme (1 mois) pour l'ajustement de la planification opérationnelle en fonction des saisons et des conditions météorologiques. Pour atteindre ces objectifs, nous avons mis en œuvre divers modèles de réseaux de neurones, tels que les réseaux LSTM et GRU, spécialement adaptés pour capturer les dépendances temporelles complexes présentes dans les données éoliennes.

Mots clés : Apprentissage automatique, Modèles de prévision, Traitement des données temporelles, Séries temporelles.

Abstract : The SRD company stands out for its highly technological electricity distribution network, focused on efficiency, the integration of renewable energies and the reliability of the energy supplied. In partnership with the LIAS laboratory, SRD accesses the latest advances in wind modeling and machine learning, thus strengthening the credibility of its projects and promoting a fruitful exchange of knowledge between industry and academia. Our work consists of developing advanced forecasting models for wind production using temporal data processing techniques, notably time series. These models are designed to accurately anticipate wind production on two specific time scales : short term (2 hours) for real-time management of the electricity production-demand balance, and medium term (1 month) for adjusting operational planning according to seasons and weather conditions. To achieve these goals, we implemented various neural network models, such as LSTM and GRU networks, specifically tailored to capture the complex temporal dependencies present in wind data.

Keywords : Machine learning, Forecasting models, Temporal data processing, Time series.

المخلص :

تميز شركة SRD بشبكها لتوزيع الكهرباء ذات التقنية العالية، والتي تركز على الكفاءة وتكامل الطاقات المتجددة وموثوقية الطاقة الموردة. بالشراكة مع مختبر LIAS ، تصل SRD إلى أحدث التطورات في نمذجة الرياح والتعلم الآلي، وبالتالي تعزيز مصداقية مشاريعها وتعزيز التبادل المثمر للمعرفة بين الصناعة والأوساط الأكاديمية. يتكون عملنا من تطوير نماذج تنبؤية متقدمة لإنتاج الرياح باستخدام تقنيات معالجة البيانات الزمنية، ولا سيما السلاسل الزمنية. تم تصميم هذه النماذج لتوقع إنتاج الرياح بدقة على نطاقين زمنيين محددين : المدى القصير (ساعتان) للإدارة في الوقت الحقيقي لتوازن إنتاج الكهرباء والطلب، والمدى المتوسط (شهر واحد) لضبط التخطيط التشغيلي وفقاً للمواسم والطقس. شروط. ولتحقيق هذه الأهداف، قمنا بتنفيذ نماذج مختلفة للشبكات العصبية، مثل شبكات LSTM و GRU ، المصممة خصيصاً لالتقاط التبعيات الزمنية المعقدة الموجودة في بيانات الرياح.

الكلمات المفتاحية : التعلم الآلي، نماذج التنبؤ، معالجة البيانات الزمنية، السلاسل الزمنية.