

N° Ordre...../FHC/UMBB/2023

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE  
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE  
**UNIVERSITE M'HAMED BOUGARA-BOUMERDES**



**Faculté des Hydrocarbures et de la Chimie**  
**Mémoire de Fin d'Etudes**  
**En vue de l'obtention du diplôme :**

## **MASTER**

Présenté par

**MOUSSAOUI DOUAA**  
**TERCHA WASSILA**

Filière : Hydrocarbures  
Option : Génie Electrique : Electricité Industrielle

### **Thème**

---

**Optimisation des performances des systèmes photovoltaïques par l'utilisation de l'apprentissage automatique et du l'apprentissage profond pour la prédiction des irradiances et des températures.**

---

**Devant le jury :**

HAMADACHE Mohammed	MCB	UMBB	Président
KAHOUL Fadhila	MCB	UMBB	Examinatrice
KHELIFI Fateh	MAA	UMBB	Examineur
TADJER Sid Ahmed	MCA	UMBB	Encadreur
ZAHRAOUI Younes	Senior-chercheur	Univ d'Estonie	Co-encadrant

Année universitaire 2022/2023

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE  
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE  
UNIVERSITE M'HAMED BOUGARA-BOUMERDES



**Faculté des Hydrocarbures et de la Chimie**  
Département : Automatisation et Electrification des procédés  
Filière : Hydrocarbures  
Option : Génie Electrique : Electricité Industrielle

**Mémoire de Fin d'Etudes**  
**En vue de l'obtention du diplôme :**

**MASTER**

***Thème***

---

**Optimisation des performances des systèmes photovoltaïques par l'utilisation de l'apprentissage automatique et du l'apprentissage profond pour la prédiction des irradiances et des températures.**

---

**Présenté par :**  
MOUSSAOUI DOUAA  
TERCHA WASSILA

**Avis favorable de l'encadreur :**

**Signature**

**TADJER S A**

**Avis favorable du Président du jury**  
**Nom Prénom** **Signature**

**Cachet et signature**

## **REMERCIEMENT**

Nos remerciements s'adressent premièrement et avant tout à ALLAH, le Tout-Puissant, qui nous a donné la force, le courage et volonté d'aller jusqu'au bout et de mener ce présent travail à son point ultime.

Tout d'abord, on tient à exprimer notre profonde reconnaissance et notre vif remerciement à Monsieur TADJER Sid Ahmed, qui nous a fait l'honneur d'encadrer ce travail avec une grande disponibilité. Sa confiance, ses encouragements, son support et la passion de son travail ont été très stimulants.

Nous tenons également à remercier Monsieur ZAHRAOUI Younes pour son précieux soutien tout au long de notre parcours.

Nous tenons à remercier particulièrement les membres du jury pour avoir accepté d'examiner ce travail et de prendre part du jury. Nos vifs remerciements s'adressent à nos enseignants du département Automatisation et électrification des procédés Industriels de la Faculté Des Hydrocarbures et de la Chimie pour leur soutien, et pour la qualité de formation qu'ils nous ont offerte durant toutes ces années d'études.

Enfin toute personne qui a participé de près ou de loin à l'accomplissement de ce mémoire soit sincèrement remerciée.

## **Dédicace**

Chers Parents ;

C'est avec une immense gratitude que je dédie ce mémoire à vous deux un projet qui représente le fruit de mes efforts, de ma passion et de ma détermination tout au long de mon parcours.

Votre influence, votre compréhension et vos douaa ont joué un rôle essentiel dans la réalisation de ce travail, et je tiens à vous exprimer toute ma gratitude.

Je dédie également ce mémoire :

À mes frères OUSSAMA, ABDELHAK, WALID et DJILLALI

À toute ma famille paternelle qu'elle soit ou maternelle.

À tous mes cousines surtout : MAROUA, LINA, MERIEM, SARA et ASMA, merci d'avoir fait tout ce chemin avec moi.

À tous mes camarades de la promotion H18, et à mon groupe MAEI18

À tous ce qui ont contribué de prêt et de loin à la réalisation de ce mémoire.

Enfin, à tous ceux qui m'aiment.

**MOUSSAOUI DOUAA**

## **Dédicace**

Je dédie ce modeste travail :

Aux êtres les plus chers de ma vie, mon père Mr ABDELKADER TERCHA et ma mère Mme YAMINA ALLIOUI qui ont été les bougies allumant mon chemin vers la réussite ; qui m'ont tout donné, leurs amours, leurs sacrifices pour que je puisse suivre mes études dans des bonnes conditions et qui ne cessent pas de m'encourager et de veiller pour mon bien.

À tous ce qui ont contribué de prêt et de loin à la réalisation de ce mémoire.

Enfin, à tous ceux qui m'aiment.

**TERCHA Wassila**

---

Remerciements	
Dédicace	
Liste des figures	
Liste des tableaux	
Nomenclature et abréviations	
<b>Introduction générale.....</b>	<b>1</b>
<b>Chapitre I : L'apprentissage automatique et l'apprentissage profond</b>	
I.1 Introduction.....	<b>4</b>
I.2 L'apprentissage automatique .....	<b>4</b>
I.2.1 Définition .....	4
I.3 Les différents types de machine Learning .....	<b>5</b>
I.3.1 L'apprentissage supervisé.....	6
I.3.1.1 Classification.....	6
I.3.1.2 Régression.....	6
I.3.2 L'apprentissage non supervisé .....	7
I.3.2.1 Regroupement (Clustering).....	7
I.3.2.2 Réduction de la dimension.....	7
I.3.3 L'apprentissage semi-supervisé .....	9
I.3.4 L'apprentissage par renforcement.....	9
I.4 Les différents types d'algorithme .....	<b>10</b>
I.5 Arbre de décision .....	<b>12</b>
I.5.1 Définition .....	12
I.5.2 Le fonctionnement de l'arbre de décision.....	13
I.6 La forêt aléatoire .....	<b>14</b>
I.6.1 Définition .....	14
I.6.2 Le fonctionnement de la forêt aléatoire .....	14

---

I.7 L'apprentissage profond.....	15
I.7.1 Définition .....	15
I.8 Architectures de L'apprentissage profond.....	15
I.8.1 L'apprentissage supervisé.....	16
I.8.2 L'apprentissage semi-supervisé .....	16
I.8.3 L'apprentissage non supervisé.....	16
I.9 Modèles du Deep Learning .....	17
I.9.1 Long Short-Term Memory (LSTM) .....	17
I.9.2 Gated Recurrent Units (réseau de neurones récurrents à portes).....	19
I.10 Conclusion .....	21

## **Chapitre II : Généralités sur les systèmes photovoltaïques**

II.1 Introduction .....	23
II.2 Energie solaire .....	23
II.2.1 L'énergie solaire thermique .....	23
II.2.2 L'énergie solaire photovoltaïque.....	23
II.3 Rayonnement solaire .....	24
II.3.1 Le rayonnement direct .....	24
II.3.2 Le rayonnement diffus.....	24
II.3.3 Le rayonnement réfléchi (ou albédo).....	24
II.4 Conversion photovoltaïque.....	26
II.5 Le solaire photovoltaïque .....	26
II.6 L'effet photovoltaïque .....	27
II.7 Cellules photovoltaïques.....	28
II.7.1 Définition.....	28
II.7.2 Type des cellules photovoltaïque .....	28
II.7.2.1 Silicium monocristallin.....	28
II.7.2.2 Silicium polycristallin.....	29

---

II.7.2.3 Les cellules polycristallins.....	29
II.7.2.4 Silicium amorphe (a-si) .....	30
II.8 Modélisation d'un système photovoltaïque.....	<b>32</b>
II.8.1 Modèle de la cellule solaire .....	32
II.8.2 Les modèles mathématiques.....	34
a) Cellule idéale.....	34
b) Modèle à une diode .....	35
c) Modèle à deux diodes.....	37
II.9 Influence de la température et de l'éclairement.....	<b>37</b>
II.9.1 L'éclairement.....	37
II.9.2 La température.....	38
II.10 Avantages et inconvénients de l'énergie photovoltaïque .....	<b>39</b>
II.10.1 Les avantages.....	39
II.10.2 Les inconvénients .....	39
II.11 Conclusion.....	<b>40</b>

### **Chapitre III : La prévision météorologiques le langage de développement**

III.1 Introduction .....	<b>42</b>
III.2 Définition de la prévision météorologiques .....	<b>42</b>
III.3 L'impact des conditions météorologiques sur l'efficacité des systèmes PV.....	<b>42</b>
III.3.1 L'irradiance solaire .....	42
III.3.2 La température.....	43
III.4 Les méthodes de prévision météorologique .....	<b>43</b>
III.5 Horizons de prévision.....	<b>44</b>
III.5.1 Les prévisions à court terme.....	44
III.5.2 Les prévisions à moyen terme .....	44
III.5.3 Les prévisions à long terme.....	44
III.6 Avantages de la prédiction météorologique .....	<b>44</b>

---

III.7 Défis dans les prévisions météorologique pour la gestion des systèmes PV .....	45
III.8 Le langage de développement (Python) .....	46
III.8.1 Définition .....	46
III.8.2 Les avantages de Python .....	46
III.8.3 Les bibliothèques de python.....	48
III.8.3.1 Pandas.....	48
III.8.3.2 Numpy .....	49
III.8.3.3 Matplotlib .....	50
III.8.3.4 Scikit learn.....	51
III.8.3.5 Keras.....	52
III.8.3.6 Tensorflow .....	52
III.8.4 L'environnement de développement.....	53
III.8.4.1 Le navigateur Anaconda.....	53
III.8.4.1.1 Jupyter .....	54
III.8.4.1.2 Definition .....	54
III.8.4.1.3 Description des jeux de données .....	55
III.8.4.1.4 Prétraitement des données (normalisation, nettoyage).....	55
III.8.4.1.5 Séparation des ensembles de données (entraînement, test).....	56
III.8.5 Métriques de comparaison .....	57
III.8.5.1 MAE.....	57
III.8.5.2 MSE.....	58
III.8.5.3 RMSE.....	59
Conclusion.....	59

## **Chapitre IV : Implémentation des résultats**

IV.1. Choix des méthodes de machines learning.....	61
IV.1.1. La régression .....	61
IV.1.2. Pourquoi la régression.....	61

---

IV.1.3. Les méthodes choisies.....	61
IV.2. Choix des méthodes de Deep Learning.....	62
IV.2.1. La méthode des réseaux de neurones récurrents (RNN).....	62
IV.2.2. Pourquoi la méthode de réseaux de neurones récurrents .....	62
IV.2.3. Les méthodes choisies.....	62
IV.3. Méthodologie expérimentale.....	63
IV.3.1. L'arbre de décision.....	63
IV.3.2. Le foret aléatoire .....	74
IV.3.3. LSTM.....	78
IV.3.4. GRU .....	85
IV.4. Comparaison entre les modèles.....	91
IV.5. Examen des performances des modèles .....	92
IV.6. Conclusion .....	93
<b>Conclusion générale .....</b>	<b>93</b>

## **Bibliography**

**Chapitre I : L'apprentissage automatique et l'apprentissage profond**

## Liste de figures

I.1 Figure : Un modèle de ML qui essaye d'obtenir la meilleure approximation possible de F...	5
I.2 Figure : Type d'apprentissage en ML.....	6
I.3 Figure : Clustering.....	7
I.4 Figure : Réduction de la dimension.....	8
I.5 Figure: Supervised vs Unsupervised Learning.....	9
I.6 Figure : Mécanisme d'apprentissage par renforcement, interactions entre l'environnement et l'agent.....	10
I.7 Figure : Description des différents composants d'un arbre.....	12
I.8 Figure : Schéma de la forêt aléatoire.....	14
I.9 Figure : Architecture d'un bloc LSTM.....	17
I.10 Figure : La structure d'une cellule mémoire GRU.....	20

**Chapitre II : Généralités sur les systèmes photovoltaïques**

## Liste de figures

II.1 Figure : Rayonnement solaire mesuré au sommet de l'atmosphère au niveau de mer.....	25
II.2 Figure : Normes de mesures du spectre d'énergie émis par le soleil, notion de la convention AM.....	26
II.3 Figure : Présente le silicium de type N et le silicium de type P.....	28
II.4 Figure : Cellule silicium monocristallin.....	29
II.5 Figure : Cellule silicium Polycristallin.....	29
II.6 Figure : Cellule silicium amorphe.....	30
II.7 Figure : Schéma électrique équivalent d'une cellule photovoltaïque idéale.....	34
II.8 Figure : Schéma électrique équivalent du modèle à une diode.....	35
II.9 Figure : Schéma électrique équivalent du modèle deux diodes.....	37
II.10 Figure : Evolution de la caractéristique I(V) et P(V) en fonction de l'irradiation.....	38
II.11 Figure : Evolution de la caractéristique I(V) et P(V) pour différentes températures.....	38

**Chapitre III : La prévision météorologiques le langage de développement**

## Liste de figures

III.1 Figure : Langage de programmation python.....	46
III.2 Figure : Bibliothèque pandas.....	48
III.3 Figure : Bibliothèque Numpy.....	49
III.4 Figure : Les formes de matrices en numpy.....	49
III.5 Figure : Bibliothèque matplotlib.....	50
III.6 Figure : Les types des graphes en Bibliothèque Matplotlib.....	51
III.7 Figure : Bibliothèque scikit learn.....	51
III.8 Figure : Bibliothèque Keras.....	52
III.9 Figure : Bibliothèque TensorFlow.....	52
III.10 Figure : Le navigateur Anaconda.....	53
III.11 Figure : Le notebook Jupyter.....	54
III.12 Figure : La page de travail en Jupyter.....	54

**Chapitre IV : Implémentation des résultats**

## Liste de figures

IV.1 Figure : Visualisation des données de température.....	64
IV.2 Figure : les valeurs actuelles de température avec les valeurs prédites.....	65
IV.3 Figure : Les valeurs de température prédit par l'arbre de décision.....	66
IV.4 Figure : Les valeurs actuelles de température.....	66
IV.5 Figure : Combinaison des valeurs actuelles avec les valeurs prédites par DT.....	67
IV.6 Figure : Visualisation des données de l'irradiance.....	70
IV.7 Figure : Les valeurs actuelles de l'irradiance avec les valeurs prédites par DT.....	70
IV.8 Figure : Les valeurs actuelles de l'irradiance.....	71
IV.9 Figure : Les valeurs prédites de l'irradiance.....	71
IV.10 Figure : Les valeurs actuelles de température et les valeurs prédits en forêt aléatoire....	73
IV.11 Figure : Un intervalle des valeurs prédits et les valeurs actuelles de la température.....	74

IV.12 Figure : les valeurs actuelles de l'irradiance et les valeurs prédits en forêt aléatoire.....75

IV.13 Figure : Un intervalle des valeurs prédits et les valeurs actuelles de l'irradiance.....75

IV.14 Figure : Les pertes par époques pour la prédiction de température en utilisant LSTM....79

IV.15 Figure : Les valeurs actuelles de température et les valeurs prédits par LSTM.....80

IV.16 Figure : Un intervalle des valeurs actuelles de la température des valeurs prédits par LSTM.....80

IV.17 Figure : Les valeurs actuelles de test de l'irradiance avec les valeurs prédites par LSTM.....81

IV.18 Figure : Un intervalle les valeurs actuelles de test de l'irradiance avec les valeurs prédites par LSTM.....82

IV.19 Figure : Les pertes par époques pour la prédiction de la température en utilisant GRU...82

IV.20 Figure : Les valeurs actuelles de température et les valeurs prédits par GRU.....85

IV.21 Figure : Un intervalle les valeurs actuelles de test de température avec les valeurs prédites par GRU.....86

IV.22 Figure : Les pertes par époques pour la prédiction de l'irradiance en utilisant GRU.....86

IV.23 Figure : Les valeurs actuelles de l'irradiance et les valeurs prédits par GRU.....87

IV.24 Figure : Un intervalle les valeurs actuelles de test de l'irradiance avec les valeurs prédites par GRU.....88

IV.25 Figure : Les valeurs prédites pour examiner les performances des modèles.....88

<b>Tableau I.1:</b> L'apprentissage supervisé et non supervisé.....	8
<b>Tableau II.1:</b> Tableau comparatif des différentes technologies les plus utiliser d'une cellule photovoltaïque.....	32
<b>Tableau III.1 :</b> Description des données.....	55
<b>Tableau IV.1 :</b> Les résultats des métriques pour l'irradiance (Arbre de décision).....	73
<b>Tableau IV.2 :</b> Les résultats des métriques pour la température et l'irradiance (Forêts aléatoires) .....	77
<b>Tableau IV.3 :</b> Les résultats des métriques pour la température et l'irradiance (LSTM).....	84
<b>Tableau IV.4 :</b> Les résultats des métriques pour la température et l'irradiance (GRU).....	90
<b>Tableau IV.5 :</b> Comparaison entre les valeurs des métriques des déférents modèles.....	91
<b>Tableau IV.6 :</b> les valeurs prédites pour examiner les performances des modèles.....	92

SVM: Support Vector Machines

ML: Machine Learning

DL: Deep Learning

DNN: Deep Neural Networks

CNN: Convolutional Neural Networks

RNN: Recurrent Neural Networks

LSTM: Long Short Terme Memory

AE : encodeurs automatiques

DT : Decision tree

RBM : Machines Boltzmann restreintes

$X_t$  : l'entrée courante à l'instant t

$h_{t-1}$  : la sortie a l'instant t-1

$b_f$  : un vecteur de biais

$W_f$  : une matrice de poids

$\sigma$  : fonction sigmoïde utilisée

$N_t$ : valeur candidate

$C_t$  : état interne

Tanh : la fonction d'activation tangente hyperbolique

$h_t$  : sortie obtenue

PV : photovoltaïque

AM : La masse d'air

$\theta$  : Est l'angle du soleil depuis le zénith

A-si : silicium amorphe

$I_{cc}$  : Courant de court circuit de la cellule.

$I_s$  : Courant de saturation de la diode

$V_{th}$ : Tension thermique.

A : Coefficient d'idéalité de la diode.

q : Charge d'un électron  $1.6 \cdot 10^{-19}$  .

$T_c$  : Température de la jonction.

K : Constante de Boltzmann  $1.38 \cdot 10^{-23}$  (J / K)

$V_m$  : Tension de point de puissance maximale (V).

$I_m$  : Courant de point de puissance maximale (A).

FF : Le facteur de forme

E : L'ensoleillement qui représente la puissance lumineuse reçue par unité de surface ( $W/m^2$ ).

S : La surface effective des cellules.

NOCT : La température nominale de fonctionnement de la cellule solaire

$T_a$  : La température ambiante.

$\eta$  : Le rendement de conversion d'énergie

$P_{in}$  : La puissance du flux lumineux incident

$P_m$  : Point de puissance maximale

$V_{oc}$  : La tension en circuit ouvert

I : Courant disponible

$I_{ph}$  : Courant produit par la photopile

V : Tension aux bornes de la cellule

$I_s$  : Courant de polarisation de la diode ou courant de saturation

$I_{pv}$  : Le courant délivré par le module.

$I_d$  : Courant de diode.

$I_{sh}$  : Le courant shunt.

$E_{ref}$ : Ensoleillement de référence

$T_{cref}$ : Température de référence

$R_s$  : Résistance série

MAE : Mean Absolute Error (Erreur Moyenne Absolue)

MSE : Mean Squared Error (Erreur quadratique moyenne)

RMSE : La mesure de la répartition de ces résidus

### **Introduction générale**

L'énergie solaire est devenue une composante essentielle de la transition énergétique vers des sources renouvelables, offrant une alternative prometteuse aux combustibles fossiles. Les systèmes photovoltaïques (PV) jouent un rôle crucial dans cette transition en permettant la conversion directe de l'énergie solaire en électricité. Ces systèmes sont largement déployés à travers le monde, aussi bien pour les installations résidentielles que pour les centrales solaires à grande échelle. L'énergie solaire présente de nombreux avantages, tels que des émissions de carbone réduites, une disponibilité abondante et une indépendance par rapport aux ressources fossiles limitées.

Cependant, l'efficacité des systèmes PV est étroitement liée aux conditions météorologiques, en particulier à l'irradiance solaire. L'irradiance solaire représente la quantité de rayonnement solaire atteignant les panneaux solaires et varie en fonction des conditions météorologiques telles que l'ensoleillement, la couverture nuageuse et les variations saisonnières. Afin d'optimiser l'efficacité des systèmes PV et de maximiser la production d'énergie solaire, il est crucial de prendre en compte ces variations et de disposer de prévisions météorologiques précises et fiables.

La prévision météorologique traditionnelle repose sur des modèles numériques basés sur des équations physiques simplifiées. Cependant, ces modèles peuvent présenter des limites en termes de résolution temporelle et spatiale, ainsi que de modélisation précise des processus météorologiques complexes. Afin de surmonter ces limites et d'améliorer la précision des prévisions météorologiques pour les systèmes PV, l'utilisation de techniques avancées d'apprentissage automatique, notamment l'apprentissage profond, suscite un intérêt croissant.

L'apprentissage profond est une branche de l'intelligence artificielle qui utilise des réseaux de neurones profonds pour analyser de grandes quantités de données et extraire des modèles complexes. L'application de l'apprentissage profond à la prévision météorologique offre de nouvelles perspectives pour améliorer la précision et la fiabilité des prévisions, en capturant les relations non linéaires entre les variables météorologiques et l'irradiance solaire. Les modèles d'apprentissage profond peuvent apprendre à partir de données historiques pour effectuer des prédictions plus précises, en prenant en compte les interactions complexes entre les variables météorologiques.

L'objectif de ce mémoire est d'explorer l'utilisation de l'apprentissage automatique et de l'apprentissage profond pour améliorer l'efficacité de système photovoltaïque grâce à la prévision météorologique. En utilisant des techniques avancées, telles que les réseaux de neurones artificiels ...

Le présent travail est subdivisé en quatre chapitres essentiels :

Le premier chapitre se concentre sur l'étude approfondie de l'apprentissage automatique et de l'apprentissage profond en explorant leurs définitions, leurs principes fondamentaux et les différents types d'algorithmes et d'architectures qui les caractérisent.

Le deuxième chapitre est consacré à l'étude de l'énergie solaire et les systèmes photovoltaïque ou en présente les différents types et la technologie utilisé dans les systèmes PV ainsi que son principe de fonctionnement.

Le troisième chapitre représente une initiative pour le langage de développement (Python) et l'environnement jupyter.

Le quatrième chapitre résume les résultats obtenus par nos modèles et des interprétations des graphes de la prédiction.

# **Chapitre I**

## **L'apprentissage automatique et l'apprentissage profond**

## I.1 Introduction

L'intelligence artificielle (IA) est un domaine en constante évolution qui vise à développer des systèmes capables de réaliser des tâches complexes, normalement associées à l'intelligence humaine. L'un des aspects clés de l'IA est l'apprentissage automatique (Machine Learning) et l'apprentissage profond (Deep Learning). Ces méthodes permettent aux machines d'apprendre à partir de données et de prendre des décisions autonomes en utilisant des algorithmes sophistiqués et des modèles mathématiques. Grâce à l'exploitation de la puissance de calcul des ordinateurs modernes l'apprentissage automatique et profond ont connu des avancées significatives ces dernières années. Ils sont utilisés dans de nombreux domaines, tels que la reconnaissance vocale, la vision par ordinateur, la prédiction des comportements ou encore l'analyse de données massives. Ces techniques offrent de nouvelles perspectives pour résoudre des problèmes complexes et améliorer les performances des systèmes intelligents. Dans ce chapitre, nous explorerons les concepts fondamentaux de l'apprentissage automatique et profond ainsi que leur modélisation.

## I.2 L'apprentissage automatique

### I.2.1 Définition

La définition de l'apprentissage automatique a connu une progression durant plusieurs années, cela est dû qu'à chaque fois il y avait de nouvelles découvertes dans ce domaine :

- Le terme « Apprentissage automatique » a été inventé par l'informaticien Américain « Arthur Samuel », dans le début de l'année 1959. Il a créé le premier programme qui permet aux ordinateurs de jouer et d'apprendre le jeu de dames sans être explicitement programmé [1]
- En 1997, l'informaticien américain « Tom Michael Mitchell » introduit une nouvelle définition de l'apprentissage automatique. Il a considéré qu'un programme apprend d'une expérience  $E$ , par rapport à une classe de tâches  $T$ , et avec une mesure de performance  $P$ . [1]
- Avec le temps, la définition de l'apprentissage automatique a commencé à prendre une dimension mathématique et statistique. Selon les auteurs [2], l'apprentissage automatique est essentiellement une forme de statistiques appliquées, mettant davantage l'accent sur

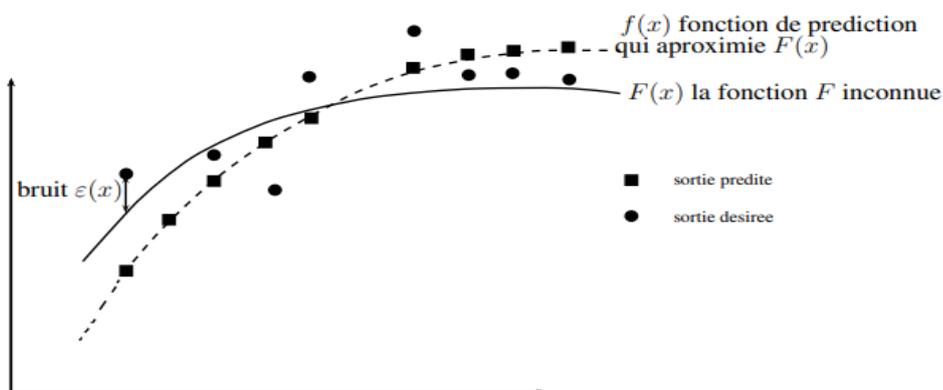
l'utilisation d'ordinateurs pour estimer statistiquement les fonctions compliquées et un accent moindre sur la démonstration des intervalles de confiance autour de ces fonctions.

Ces définitions peuvent varier en fonction de l'angle étudié, mais elles sont toutes orientées vers une seule direction, qui est définie comme suit : l'apprentissage automatique est la science ou l'art de la programmation des ordinateurs afin qu'ils puissent apprendre des données [1]

La valeur de la variable  $y$  dépend de :

- Une fonction  $F(x)$  déterminée par les variables prédictives.
- Un bruit  $\varepsilon(x)$  qui est le résultat d'un nombre de paramètres dont on ne peut pas tenir compte.

Aussi bien  $F$  que  $\varepsilon$  ne seront jamais connues mais l'objectif d'un modèle de ML est d'obtenir la meilleure approximation possible de  $F$  à partir des observations disponibles. Cette approximation sera notée  $f$ , on l'appelle la fonction de prédiction. [3]

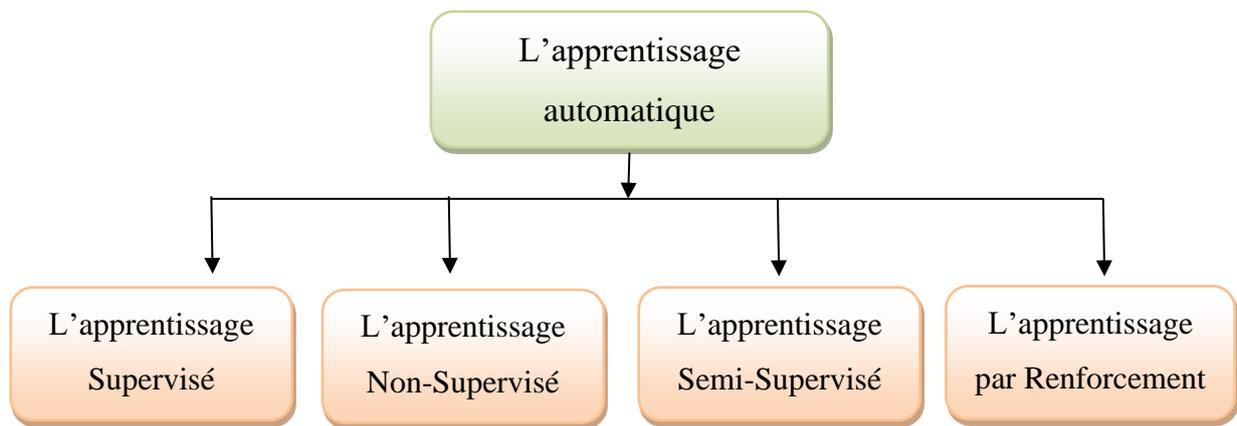


**Figure I.1** : un modèle de ML qui essaye d'obtenir la meilleure approximation possible de  $F$

### I.3 Les différents types de machine Learning

Pour comprendre comment une machine peut apprendre à partir de données, il faut se pencher sur les 4 paradigmes d'apprentissage du Machine Learning :

- **L'Apprentissage Supervisé** (Supervised Learning)
- **L'Apprentissage Non-Supervisé** (Unsupervised Learning)
- **L'Apprentissage par Renforcement** (Reinforcement Learning)
- **L'Apprentissage semi-supervisé** (Semi-supervised learning)



**Figure I.2 :** Type d'apprentissage en ML

### I.3.1 L'apprentissage supervisé

L'apprentissage supervisé est une tâche d'apprentissage automatique consistant à apprendre une fonction de prédiction à partir d'exemples annotés et bien expliqués. L'objectif principale est de définir les règles qui permettent de classer les organismes dans les classes des variables qualitatives ou quantitatives qui les caractérisent, tel que des exemples qui peuvent être liés par un expert comme base de l'apprentissage (échantillons d'apprentissage) et cette base sera utilisée dans l'apprentissage des règles de classification qui permettent de décrire un objet (par exemple: un chat ou un chien) et d'identifier une caractéristique spécifique (taille, couleur et type). [4]

Les problèmes dans l'apprentissage supervisé sont 2 types : Classification, Régression.

#### I.3.1.1 Classification

C'est une tâche d'apprentissage supervisé dont la sortie a des étiquettes définies (valeur discrète) ce qu'on appelle classes ou catégories.

On peut avoir des classifications Binaire ou alors Multiclasse.

La classification binaire prédit pour un modèle (0 ou 1) vrai ou faux, dans le cas de multiclassés le modèle va prédire plus d'une classe.

Le but ici reste le même quel que soit le nombre de classes, prédire si **les valeurs discrètes** appartiennent à une classe particulière. [5]

#### I.3.1.2 Régression

Il s'agit d'une tâche d'apprentissage supervisé dont la sortie a **une valeur continue**.

Le but ici est de prédire une valeur aussi proche de la valeur de sortie réelle que notre modèle le permet, puis l'évaluation est effectuée en calculant la valeur d'erreur. Plus l'erreur est petite, plus la précision de notre modèle de régression est grande. [5]

### I.3.2 L'apprentissage non supervisé

L'apprentissage non supervisé se caractérise par une autorégulation (l'absence totale d'enseignant), c'est-à-dire qu'il ne contient pas de signal d'erreur, comme c'est le cas dans l'apprentissage supervisé. Il apprend des observations ou des découvertes, qui consistent à définir une classification « raisonnable » d'un groupe de certaines choses ou situations où il copie les performances du cerveau humain pour trouver des informations par corrélation (assimilation de stimuli), c'est-à-dire montrer la relation entre l'ensemble des éléments. Par exemple : déterminer une mesure de similitude entre deux objets, puis trouver quel groupe d'objets est similaire l'un à l'autre, puis généraliser à des groupes beaucoup plus grands. [6]

#### I.3.2.1 Regroupement (Clustering)

Le clustering permet de séparer les données entrées en un ensemble ou groupe de données qui ont des traits similaires et de les affecter à un cluster. Contrairement à la classification dans l'apprentissage supervisé ces différents clusters ou groupe ne sont pas connus à l'avance, c'est l'algorithme lui-même qui va séparer les données aux nombres de clusters qu'il faut. [5]

Comme l'on peut le voir pouvez le voir dans l'exemple Figure II.3, les points de jeu de données donnés ont été divisés en groupes identifiables par les couleurs rouge, vert et bleu. [7]

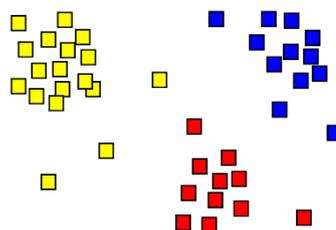
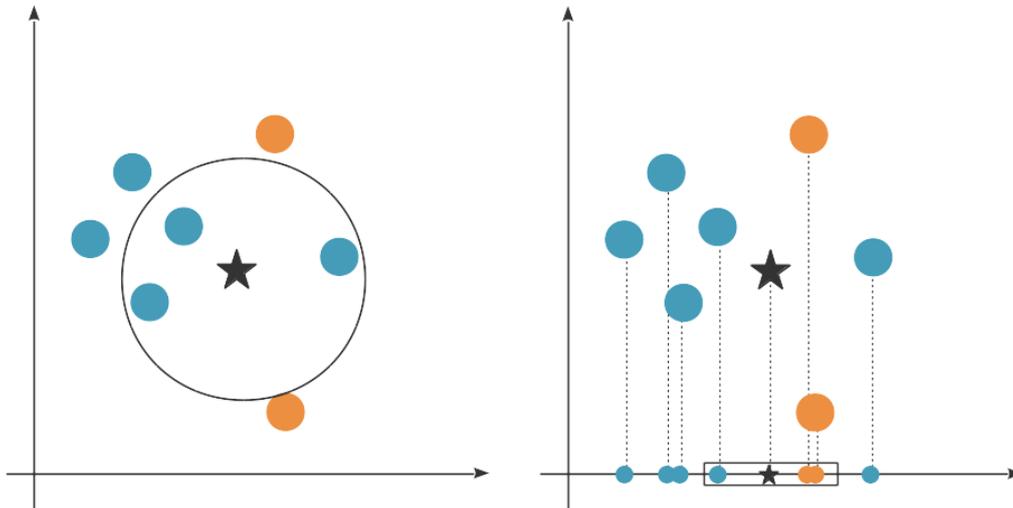


Figure I.3 : Clustering

#### I.3.2.2 Réduction de la dimension

L'objectif est de simplifier les données sans perdre trop d'informations, à titre d'exemple, fusionner plusieurs caractéristiques en un seul caractère.

En utilisant les deux dimensions, les 3 plus proches voisins de l'étoile sont majoritairement bleus. En utilisant seulement la variable en abscisse, les 3 plus proches voisins sont majoritairement orange. Si la variable en ordonnée n'est pas pertinente, elle fausse l'algorithme. [8]



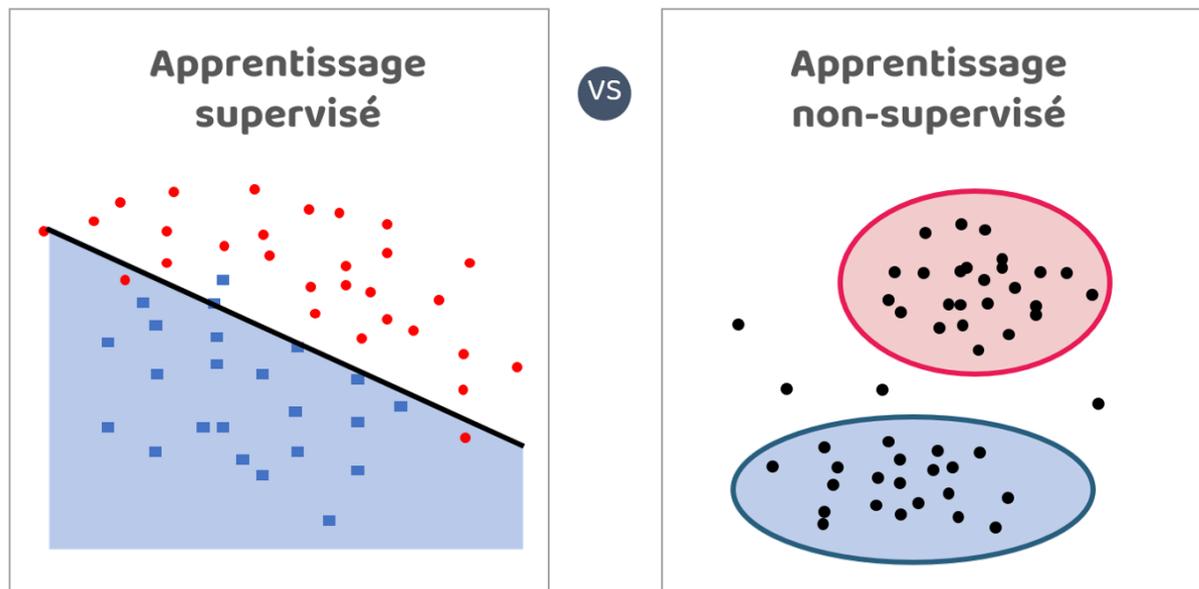
**Figure I.4 :** Réduction de la dimension

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

Ce tableau résume la différence entre apprentissage supervisé et non supervisé : [5]

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

**Tableau I.1:** L'apprentissage supervisé et non supervisé.



**Figure I.5:** Supervised vs Unsupervised Learning

### I.3.3 L'apprentissage semi-supervisé

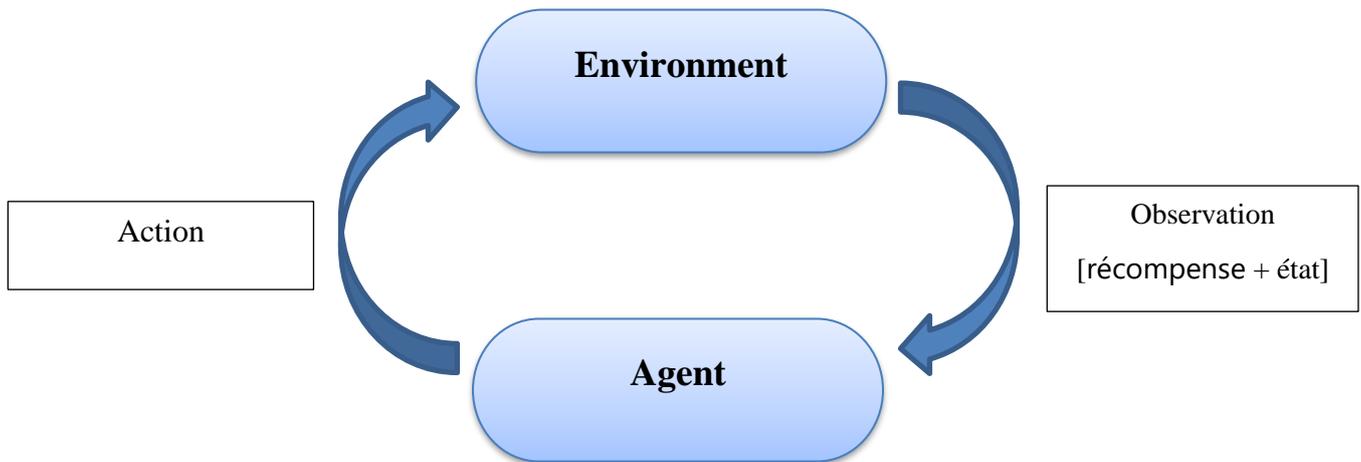
Il existe d'autres types de classification qui s'appuient sur d'autres types de méthodes d'apprentissages comme « l'apprentissage semi-supervisé ». En effet, l'apprentissage semi-supervisé est un bon compromis entre les deux types d'apprentissage « supervisé » et « non-supervisé », car il permet de traiter un grand nombre de données sans avoir besoin de toutes les étiqueter, et il profite des avantages des deux types mentionnés.

Le contexte semi-supervisé qui se situe à l'intersection entre le contexte supervisé et le contexte non supervisé, est alors une solution alternative. Il se caractérise par la présence de quelques informations disponibles sur l'ensemble des données. Ces informations sont représentées soit sous la forme de quelques données labellisées, soit sous la forme de ressemblance ou dissemblance au sein de couples de données. [9]

### I.3.4 L'apprentissage par renforcement

Avec l'apprentissage par renforcement la machine n'a pas besoin de l'aide de l'être humain, ni en termes de supervision, ni en termes de fourniture de données. L'apprentissage par renforcement est une branche très différente. Le système d'apprentissage, appelé un agent dans ce contexte (Voir Figure I.6), peut observer l'environnement, sélectionner et effectuer des actions, et enfin obtenir des récompenses ou des pénalités (des récompenses négatives). La machine peut apprendre toute seule la meilleure stratégie à suivre, appelée une politique, pour

obtenir plusieurs récompenses au fil du temps. Une politique définit l'action que l'agent devrait choisir lorsqu'il est dans une situation donnée. [1]



**Figure I.6 :** Mécanisme d'apprentissage par renforcement, interactions entre l'environnement et l'agent

L'interaction entre l'environnement et l'agent se déroule selon un cycle continu. À chaque étape, l'agent perçoit l'état de l'environnement, utilise une politique (une stratégie ou une règle) pour décider de l'action à prendre, exécute cette action dans l'environnement et reçoit une récompense ou une pénalité. La récompense fournit une indication à l'agent sur la qualité de l'action effectuée, l'incitant ainsi à apprendre et à améliorer ses performances futures.

#### I.4 Les différents types d'algorithme

Suivant notre travail on va basée sur les algorithmes de supervised learning (apprentissage supervisé) qui sont utilisés pour résoudre des problèmes de classification et de régression, où un modèle est entraîné à prédire une valeur cible ou une classe à partir d'un ensemble d'exemples d'entraînement étiquetés. Voici quelques-uns des algorithmes de supervised learning les plus couramment utilisés :

a) Régression linéaire :

Ce modèle est utilisé pour la régression, où la variable cible est une valeur continue. Il cherche à établir une relation linéaire entre les caractéristiques d'entrée et la variable cible.

b) Régression logistique :

Ce modèle est utilisé pour la classification binaire, où la variable cible est soit 0 soit 1. Il est basé sur la fonction logistique pour estimer la probabilité de classe pour chaque exemple.

c) Arbres de décision :

Les arbres de décision sont des modèles qui utilisent une structure d'arbre pour prendre des décisions en fonction des caractéristiques d'entrée. Ils peuvent être utilisés pour la classification et la régression.

d) Random Forest :

Le Random Forest est une technique qui combine plusieurs arbres de décision pour améliorer les performances et réduire le surapprentissage. Il agrège les prédictions de multiples arbres pour obtenir une prédiction finale.

e) Support Vector Machines (SVM) :

Les SVM sont utilisés pour la classification binaire et peuvent être étendus à la classification Multiclasse. Ils recherchent un hyperplan optimal pour séparer les exemples de différentes classes dans un espace de grande dimension.

f) Naïve Bayes :

Les classificateurs Naïve Bayes sont basés sur le théorème de Bayes et supposent une indépendance conditionnelle entre les caractéristiques. Ils sont rapides à entraîner et bien adaptés aux problèmes de classification textuelle.

g) Gradient Boosting :

Les algorithmes de gradient Boosting, tels que XGBoost et LightGBM, construisent un modèle prédictif en ajoutant itérativement des arbres de décision à un modèle existant, en mettant l'accent sur les exemples mal prédits.

Ces algorithmes de supervised learning offrent différentes approches pour résoudre des problèmes de classification et de régression en utilisant des données étiquetées. Le choix de l'algorithme dépend du contexte du problème, de la taille et de la nature des données, et des performances souhaitées. Il est souvent recommandé d'expérimenter plusieurs algorithmes et d'évaluer leurs performances pour trouver celui qui convient le mieux à votre tâche spécifique.

Concentrons-nous sur les algorithmes utilisés dans notre travail qui sont très largement utilisés. Donc nous allons étudier deux algorithmes de machine Learning : les arbres de décision et le Random Forest.

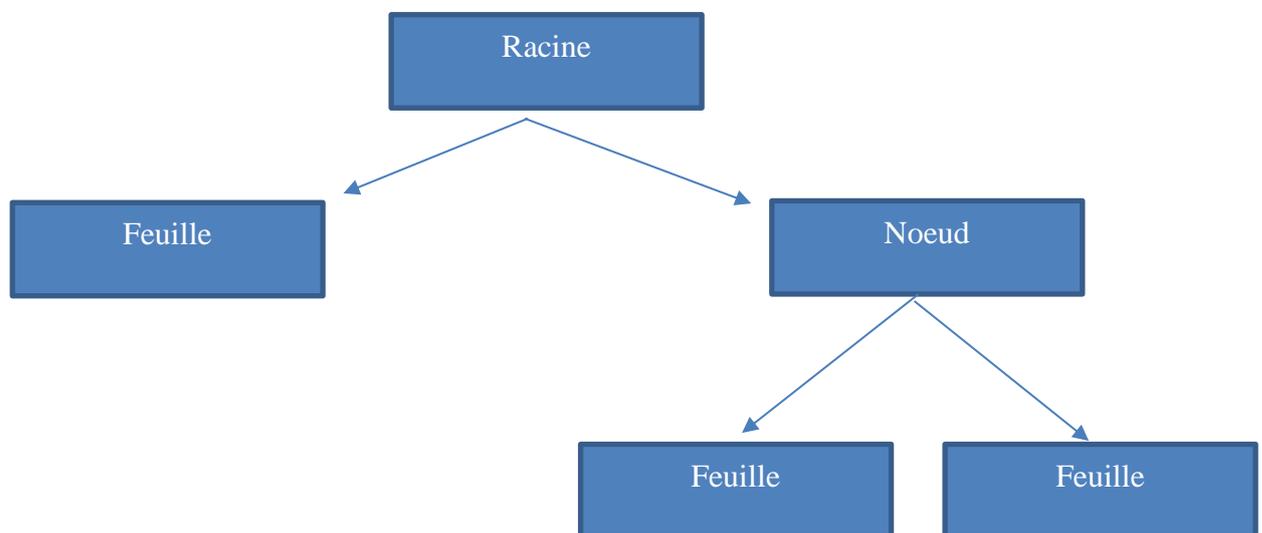
## I.5 Arbre de décision

### I.5.1 Définition

Un arbre de décision est une méthode de classification et de prédiction la plus populaire en apprentissage supervisé. Il est largement répandu dans les domaines de statistiques, de l'ingénierie, de la théorie de la décision, ou encore de l'apprentissage automatique. [10]

L'utilisation croissante des arbres de décision est liée à leur simplicité et à leur interprétabilité, à leur faible coût de calcul et à la possibilité d'être représentés graphiquement tout en gardant des performances très acceptables. Un arbre de décision représente un ensemble de restrictions ou de conditions qui sont organisées de manière hiérarchique, et qui sont appliquées successivement d'une racine à un nœud terminal ou à une feuille de l'arbre. [11]

Les arbres sont souvent représentés de haut en bas en commençant par le nœud racine, comme sur le schéma récapitulatif suivant :



**Figure I.7** : Description des différents composants d'un arbre

### I.5.2 Le fonctionnement de l'arbre de décision

Le modèle de l'arbre de décision est un algorithme d'apprentissage supervisé utilisé pour la classification et la régression. Il est représenté sous forme d'un arbre où chaque nœud interne représente une caractéristique ou un attribut, chaque branche représente une règle de décision basée sur cette caractéristique, et chaque feuille représente une classe ou une valeur prédite.

Pour illustrer cela avec une équation, supposons que nous ayons un ensemble de données avec une seule caractéristique ( $x$ ) et une variable de classe ( $y$ ). Nous voulons construire un arbre de décision pour prédire la valeur de  $y$  en fonction de  $x$ .

L'équation générale d'un arbre de décision peut être représentée comme suit :

```
if (condition1): I.1  
    if (condition2):  
        y = value1  
    else:  
        y = value2  
else:  
    y = value3
```

Chaque condition représente un test sur la caractéristique  $x$  à un nœud donné de l'arbre. Ainsi, l'arbre de décision se ramifie en fonction des différentes conditions et attribue une valeur de classe ( $y$ ) en fonction des résultats de ces tests conditionnels.

## I.6 La forêt aléatoire

### I.6.1 Définition

La forêt aléatoire ou Random Forrest est un algorithme d'apprentissage supervisé. La "forêt" qu'il construit est un ensemble d'arbres de décision, généralement formés avec la méthode "bagging". L'idée générale de cette méthode est qu'une combinaison de modèles d'apprentissage augmente le résultat global. [12]

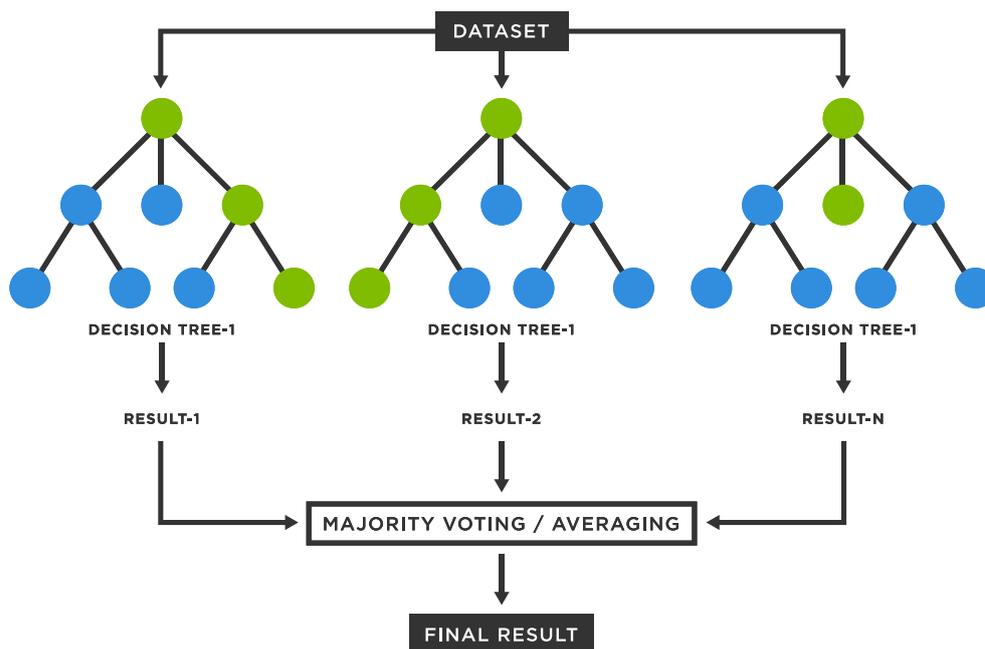


Figure I.8 : schéma de la forêt aléatoire

### I.6.2 Le fonctionnement de la forêt aléatoire

La prédiction d'une forêt aléatoire est obtenue en agrégeant les prédictions de tous les arbres individuels. La méthode d'agrégation varie en fonction du problème :

Pour les tâches de classification, la prédiction finale est généralement basée sur un vote majoritaire des classes prédites par chaque arbre. La classe qui obtient le plus grand nombre de votes est choisie comme prédiction finale.

Pour les tâches de régression, la prédiction finale est souvent basée sur la moyenne des valeurs prédites par chaque arbre.

La forêt aléatoire peut être représentée mathématiquement comme suit :

Pour une tâche de classification :

$$y = f(x) = \text{majority\_vote}(y_1, y_2, \dots, y_n) \quad \text{I.2}$$

Pour une tâche de régression :

$$y = f(x) = \text{average}(y_1, y_2, \dots, y_n) \quad \text{I.3}$$

Dans ces équations,  $y$  représente la prédiction finale de la forêt aléatoire pour une entrée  $x$ .  $y_1, y_2, \dots, y_n$  représentent les prédictions individuelles des arbres dans la forêt. La fonction `majority_vote()` effectue un vote majoritaire pour les tâches de classification, tandis que la fonction `average()` calcule la moyenne pour les tâches de régression.

Il convient de noter que la forêt aléatoire offre plusieurs avantages, notamment la réduction du surapprentissage (overfitting) et une meilleure robustesse aux données bruitées. Elle est souvent utilisée pour des problèmes de classification et de régression complexes.

## I.7 L'apprentissage profond

### I.7.1 Définition

L'apprentissage profond ou "Deep Learning" est un nouveau domaine de recherche de la machine Learning (ML), qui a été introduit dans le but de rapprocher le ML de son objectif principal à savoir : l'intelligence artificielle. Il concerne les algorithmes inspirés par la structure et du fonctionnement du cerveau. Ils peuvent apprendre plusieurs niveaux de représentation dans le but de modéliser des relations complexes entre les données. [13]

L'apprentissage profond s'appuie sur un réseau de neurones artificiels qui est composé de milliers d'unités (les neurones) dont chacune recevant et interprétant les informations de la couche précédant par exemple : reconnaissance visuelle, les premières couches d'unités identifient des lignes, des courbes, des angles, etc., des couches supérieures identifient des formes, des combinaisons de formes, des objets, des contextes, etc... [14]

### I.8 Architectures de L'apprentissage profond

Les approches d'apprentissage en profondeur tout comme l'apprentissage automatique peuvent être regroupées en différentes catégories : supervisées, semi-supervisées et non

supervisées. Une autre catégorie d'apprentissage appelée apprentissage par renforcement (RL) ou Deep RL (DRL), est également souvent abordée dans le contexte des approches d'apprentissage semi-supervisé ou, parfois, non supervisé.

### **I.8.1 L'apprentissage supervisé**

C'est une technique d'apprentissage qui utilise des données étiquetées. Dans le cas des approches DL supervisées, l'environnement dispose d'un ensemble d'entrées et de sorties correspondantes. Il modifiera alors itérativement les paramètres du réseau pour une meilleure approximation des sorties souhaitées. Il existe différentes approches d'apprentissage supervisé pour le Deep Learning, notamment les réseaux de neurones profonds (Deep Neural Networks-DNN), les réseaux de neurones convolutifs (Convolutionnel Neural Networks- CNN), les réseaux de neurones récurrents (Recurrent Neural Networks-RNN), y compris la mémoire à long terme (Long Short Terme Memory-LSTM) et les unités récurrentes fermées (GRU). [15]

### **I.8.2 L'apprentissage semi-supervisé**

C'est un apprentissage basé sur des ensembles de données partiellement étiquetés (également appelés apprentissage par renforcement). Dans certains cas, les DRL et les Réseaux Adversaires Génératifs (GAN) sont utilisés comme techniques d'apprentissage semi-supervisé. De plus, RNN, y compris LSTM et GRU, sont également utilisées pour l'apprentissage semi-supervisé. [15]

### **I.8.3 L'apprentissage non supervisé**

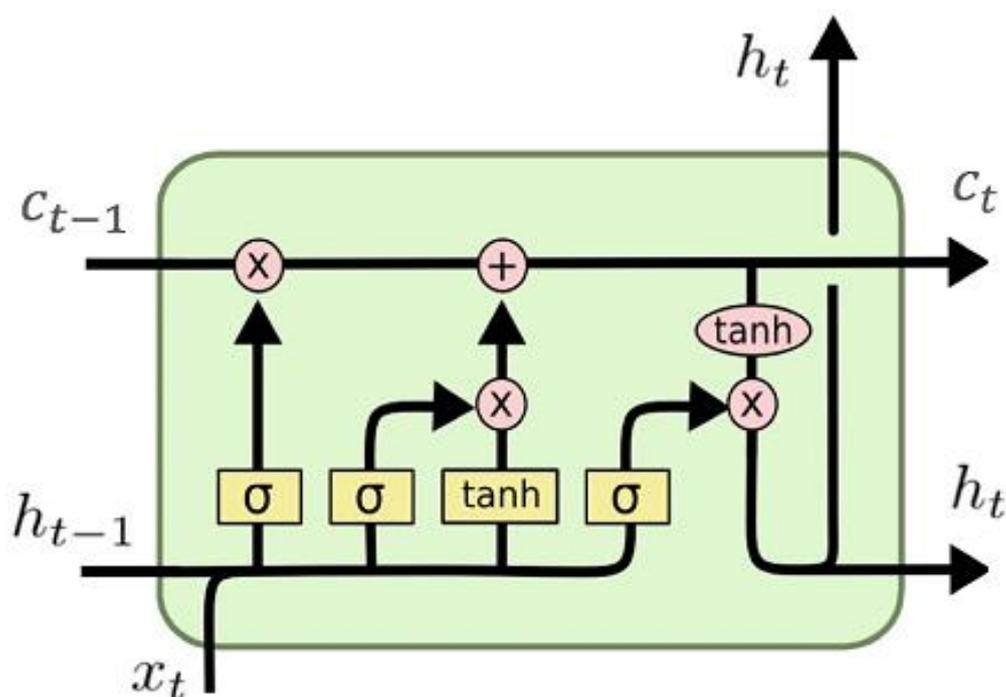
C'est un apprentissage qui se fait sans la présence d'étiquettes de données. Dans ce cas, le réseau apprend la représentation interne ou des fonctionnalités importantes pour découvrir des relations ou une structure inconnue dans les données d'entrée. Le regroupement, la réduction de la dimensionnalité et les techniques génératives sont souvent considérés comme des approches d'apprentissage non supervisé. Il y a plusieurs membres de la famille du Deep learning qui sont bons pour le clustering et la réduction de dimensionnalité non linéaire, y compris : encodeurs automatiques (AE), Machines Boltzmann restreintes (RBM) et le GAN récemment développé. En outre, les RNN, tels que LSTM (Long-Short Term Memory neural network) et RL, sont également utilisés pour l'apprentissage non supervisé dans de nombreux domaines d'application. [16]

## I.9 Modèles du Deep Learning

Nous allons nous traiter sur cette section les modèles de Deep Learning utilisés dans notre travail (LSTM et GRU).

### I.9.1 Long Short-Term Memory (LSTM)

L'architecture LSTM consiste en un ensemble de sous-réseaux récurrents particuliers (appelés blocs de mémoire) situés au niveau de la couche cachée, chacun des sous-réseaux contient une ou plusieurs cellules de mémoire. Ces blocs de mémoire spéciaux permettent le maintien d'information en mémoire pour de longues périodes de temps. LSTM est proposé par Hochrieter et Shamid Huber, il est apparu comme un modèle efficace pour plusieurs problèmes d'apprentissage qui sont liés aux données séquentielles.



**Figure I.9** : Architecture d'un bloc LSTM

Les symboles utilisés ici ont la signification suivante :

$x_t$  : l'entrée courante à l'instant  $t$

$h_{t-1}$  : la sortie à l'instant  $t-1$

$b_f$  : un vecteur de biais

$W_f$  : une matrice de poids

$\sigma$  : fonction sigmoïde utilisée

$N_t$ : valeur candidate

$C_t$  : état interne

Tanh : la fonction d'activation tangente hyperbolique

$h_t$  : sortie obtenue

Un réseau LSTM typique est composé de blocs de mémoire appelés cellules. Deux états sont en cours transféré à la cellule suivante, l'état de la cellule et l'état caché.

L'état cellulaire est la chaîne principale de flux de données, qui permet aux données de circuler en avant essentiellement inchangées. Cependant, certaines transformations linéaires peuvent se produire. Les données peuvent être ajoutées ou supprimées de l'état de la cellule via des portes sigmoïdes. Une porte est similaire à une couche ou à une série d'opérations matricielles, qui contiennent différents poids individuels.

Les LSTM sont conçus pour éviter le problème de dépendance à long terme car ils utilisent des portes pour contrôler le processus de mémorisation.

- La première étape de la construction d'un réseau LSTM consiste à identifier les informations qui ne sont pas requises et qui seront manquées de la cellule à cette étape. Ce processus d'identification et d'exclusion de données est décidé par la fonction sigmoïde, qui prend la sortie de la dernière unité LSTM ( $h_{t-1}$ ) au temps  $t - 1$  et l'entrée courante  $X_t$  au temps  $t$ . De plus, la fonction sigmoïde détermine quelle partie de l'ancienne sortie doit être éliminée. Cette porte est appelée la porte de l'oubli (ou  $f_t$ ) ; où  $f_t$  est un vecteur avec des valeurs allant de 0 à 1, correspondant à chaque nombre dans l'état de cellule,  $C_{t-1}$  . [17]

$$f_t = \sigma(W_f [h_t, X_t] + b_f) \tag{I.4}$$

Ici,  $\sigma$  est la fonction sigmoïde, et  $W_f$  et  $b_f$  sont les matrices de poids et le biais, respectivement, la porte du oubliez.

- L'étape suivante consiste à décider et à stocker les informations de la nouvelle entrée  $X_t$  dans l'état de cellule ainsi qu'à mettre à jour l'état de cellule. Cette étape contient deux parties, la couche sigmoïde et la seconde la couche  $\tanh$ . Tout d'abord, la couche sigmoïde décide si les nouvelles informations doivent être mises à jour ou ignorées (0 ou 1), et deuxièmement, la fonction  $\tanh$  donne du poids aux valeurs qui sont passées, en décidant de leur niveau d'importance (-1 à 1). Les deux valeurs sont multipliées pour mettre à jour le nouvel état de cellule. Cette nouvelle mémoire est ensuite ajoutée à l'ancienne mémoire  $C_{t-1}$  résultant en  $C_t$ . [17]

$$i_t = \sigma(W_i [h_{t-1}, X_t] + b_i), \quad (I.5)$$

$$N_t = \tanh(W_n [h_{t-1}, X_t] + b_n), \quad (I.6)$$

$$C_t = C_{t-1} f_t + N_t i_t. \quad (I.7)$$

Ici,  $C_{t-1}$  et  $C_t$  sont les états de cellule au temps  $t - 1$  et  $t$ , tandis que  $W$  et  $b$  sont les matrices de poids et le biais, respectivement, de l'état de la cellule.

- Dans la dernière étape, les valeurs de sortie  $h_t$  sont basées sur l'état de la cellule de sortie  $O_t$  mais il s'agit d'une version filtrée. Tout d'abord, une couche sigmoïde décide quelles parties de l'état de la cellule parviennent à la sortie. Ensuite, la sortie de la porte sigmoïde  $O_t$  est multipliée par les nouvelles valeurs créées par la couche  $\tanh$  à partir de l'état de la cellule  $C_t$ , avec une valeur comprise entre -1 et 1. [17]

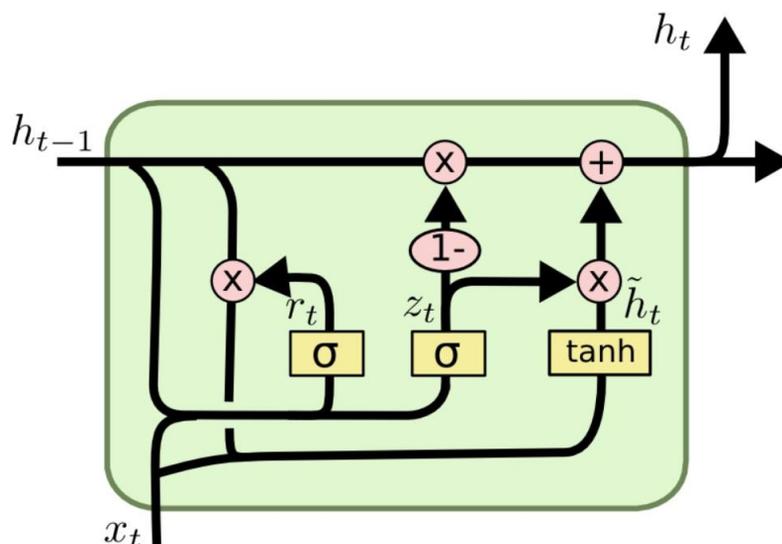
$$O_t = \sigma(W_o [h_{t-1}, X_t] + b_o) \quad (I.8)$$

$$h_t = O_t \tanh(C_t) \quad (I.9)$$

Ici,  $W_o$  et  $b_o$  sont les matrices de poids et le biais, respectivement, de la porte de sortie.

### I.9.2 Gated Recurrent Units (réseau de neurones récurrents à portes)

GRU est proposée en 2014, c'est par LSTM que GRU dispose également de portails, le premier est le portail de mise à jour qui combine les portails d'oubli et d'entrée de LSTM. L'activation de ce portail est liée aux unités avec des dépendances à long terme. La deuxième porte, cela de réinitialisation qui permet à l'unité d'oublier le passé si elle est fermée ( $rt = 0$ ) signifie que l'unité est capable de lire uniquement le premier code de la chaîne d'entrée, l'activation de ce portail est liée aux unités avec des dépendances à court terme. [18]



**Figure I.10** : La structure d'une cellule mémoire GRU.

Nous définissons les équations dans la cellule GRU comme suit : [19]

Portail de mise à jour :

$$z_t = \sigma(w_z x_t + w_z h_{t-1}) \quad (\text{I.10})$$

Porte de réinitialisation :

$$r_t = \sigma(w_r x_t + w_r h_{t-1}) \quad (\text{I.11})$$

$w$  : Matrice de poids.

L'activation du filtre est calculée comme suit :

$$\tilde{h}_t = \tanh(w * x_t + w(r_t * h_{t-1})) \quad (\text{I.12})$$

La valeur finale de GRU (achèvement linéaire) :

$$h_t = (1 - z_t)h_{t-1} + z_t * \tilde{h}_t \quad (\text{I.13})$$

Etant donné que GRU réduit les besoins en mémoire par rapport à LSTM, il résout également les problèmes RNN, mais aussi avoir des inconvénients, dont la multiplicité des états cachés qui compliquent le calcul et les besoins en mémoire plus que RNN.

## I.10 Conclusion

En conclusion de ce chapitre dédié à l'apprentissage automatique et à l'apprentissage profond nous pouvons constater que ces deux domaines ont connu des avancées significatives et ont transformé la façon dont nous abordons les problèmes d'intelligence artificielle.

L'apprentissage automatique en exploitant les algorithmes et les techniques de modélisation a permis aux machines d'apprendre à partir des données et de prendre des décisions intelligentes. Nous avons exploré différents types d'apprentissage notamment l'apprentissage supervisé, l'apprentissage non supervisé, l'apprentissage semi-supervisé et l'apprentissage par renforcement. Chacun de ces types offre des approches uniques pour résoudre des problèmes spécifiques.

Parmi les algorithmes étudiés, l'arbre de décision et le modèle Random Forest se sont révélés particulièrement puissants. L'arbre de décision permet de prendre des décisions basées sur une série de tests conditionnels tandis que le modèle Random Forest utilise un ensemble d'arbres de décision pour améliorer la précision et la robustesse du modèle.

En ce qui concerne l'apprentissage profond nous avons exploré différentes architectures, telles que l'apprentissage supervisé, l'apprentissage semi-supervisé, l'apprentissage non supervisé et l'apprentissage par transfert. Ces architectures reposent sur l'utilisation de réseaux de neurones profonds, tels que le Long Short-Term Memory (LSTM) et les Gated Recurrent Units (GRU), qui permettent de capturer des dépendances temporelles complexes dans les données.

# **Chapitre II**

## **Généralités sur les systèmes photovoltaïques**

## II.1 Introduction

L'énergie photovoltaïque est la transformation directe de l'énergie de la lumière du soleil en énergie électrique au moyen des cellules solaires fabriquées à base du matériau semi-conducteur généralement au silicium. L'association en série et en parallèle de ces cellules donne lieu à un module photovoltaïque. Pour obtenir une puissance d'utilisation désirée, ces modules peuvent être également assemblés en série et en parallèle pour former un générateur PV. La puissance produite par le générateur PV dépend du niveau d'éclairement et de la température de jonction de la cellule. [20]

## II.2 Energie solaire

Le soleil est une source quasiment inépuisable d'énergie qui envoie à la surface de la terre un rayonnement qui représente chaque année environ 8400 fois la consommation énergétique. Cela correspond à une puissance instantanée reçue de 1 kilowatt crête par mètre carré ( $KWc/m^2$ ) répartie sur tout le spectre, de l'ultraviolet à l'infrarouge. Les déserts de notre planète reçoivent en 6 heures plus d'énergie du soleil que n'en consomme l'humanité en une année. L'énergie solaire est produite et utilisée selon plusieurs procédés :

### II.2.1 L'énergie solaire thermique

Qui consiste tout simplement à produire de la chaleur grâce à des panneaux sombres. On peut aussi produire avec de la vapeur à partir de la chaleur du soleil puis convertir la vapeur en électricité.

### II.2.2 L'énergie solaire photovoltaïque

Qui consiste à produire directement de l'électricité à partir de la lumière à l'aide de panneaux solaires. Cette forme d'énergie est déjà exploitée dans de nombreux pays, surtout dans les pays ou les régions ne disposant pas de ressources énergétiques conventionnelles tels que les hydrocarbures ou le charbon. [21]

Le soleil est une étoile formée de gaz extrêmement chauds, il est constitué de 75% d'hydrogène, 23% d'hélium et 2% d'autres molécules. Les atomes d'hydrogène sont transformés en hélium par une réaction de fusion thermonucléaire massive. La masse est alors transformée en énergies selon la formule d'Einstein ( $E=mc^2$ ). Cette réaction maintient /la surface du soleil à une température approximative de 5800 °K. L'énergie du soleil est transformée sous forme de rayonnement dans l'espace de manière uniforme et dans toutes les

directions. Lorsqu'elle a parcouru 150 millions de kilomètres du soleil vers la terre, sa densité extraterrestre totale diminue à  $1367 \text{ W/m}^2$ . Cette valeur est connue sous le nom de constante solaire. En une heure, la terre reçoit suffisamment d'énergie du soleil ( $1,2 \cdot 10^{17} \text{ W}$ ) pour combler ses besoins pour presque une année. Les panneaux photovoltaïques permettent de profiter de cette énergie gratuite en la transformant en énergie électrique. [22]

### **II.3 Rayonnement solaire**

Le rayonnement solaire est un rayonnement électromagnétique émis par la surface de soleil qui se propage à une vitesse de  $3 \cdot 10^8 \text{ km/s}$ . Pour l'analyse du rayonnement à la surface de la terre, il est important de connaître l'intensité du rayonnement et sa composition. Le rayonnement solaire incident subit une série de réflexions, dispersions et absorptions au cours de sa route vers le sol. En raison des fluctuations climatiques son incidence totale est la somme des composants directs, diffus et réfléchis comme présentés dans la figure II.2 [23]. Ils se décomposent comme suit :

#### **II.3.1 Le rayonnement direct**

Faisceau direct d'énergie solaire qui est intercepté par la surface sans aucune interaction avec les particules dans l'atmosphère.

#### **II.3.2 Le rayonnement diffus**

Rayonnement intercepté qui est diffusé dans l'atmosphère par les gaz et les aérosols.

#### **II.3.3 Le rayonnement réfléchi (ou albédo)**

Ce rayonnement est réfléchi par le sol et les surfaces environnantes. De cette manière, l'index masse d'air (AM) a été créé, ce qui correspond à la normalisation du chemin emprunté par la lumière du soleil dans l'atmosphère par rapport au chemin le plus court possible, étant associé à la composition spectrale des rayons solaires qui tombent à la surface de la terre. L'absence de changement de l'ensoleillement solaire serait une masse d'air zéro ou AM 0. La masse d'air AM 0 est prise en compte dans les applications spatiales. La masse d'air AM 1 correspond à la trajectoire perpendiculaire à la surface de la terre. La masse d'air AM 1.5 est considéré comme le spectre de référence pour le test sur panneau photovoltaïque à la surface de la terre.

L'indice AM peut être approximativement calculé par l'équation II.1, et La figure II.1 illustre le spectre du rayonnement solaire reçu pour différentes conditions AM [24].

$$Am = \frac{1}{\cos \theta}$$

II.1

$\theta$  : Est l'angle du soleil depuis le zénith

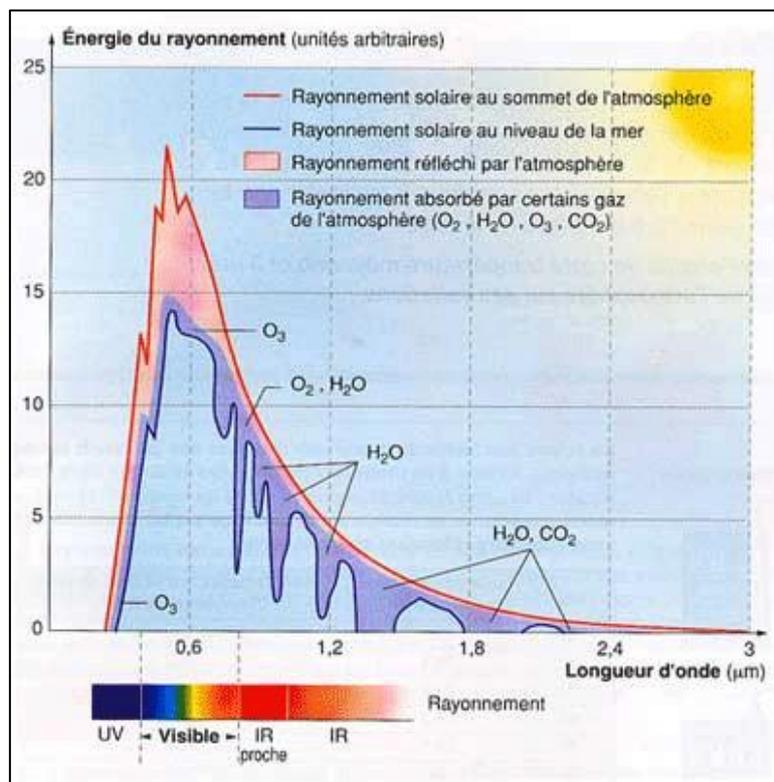
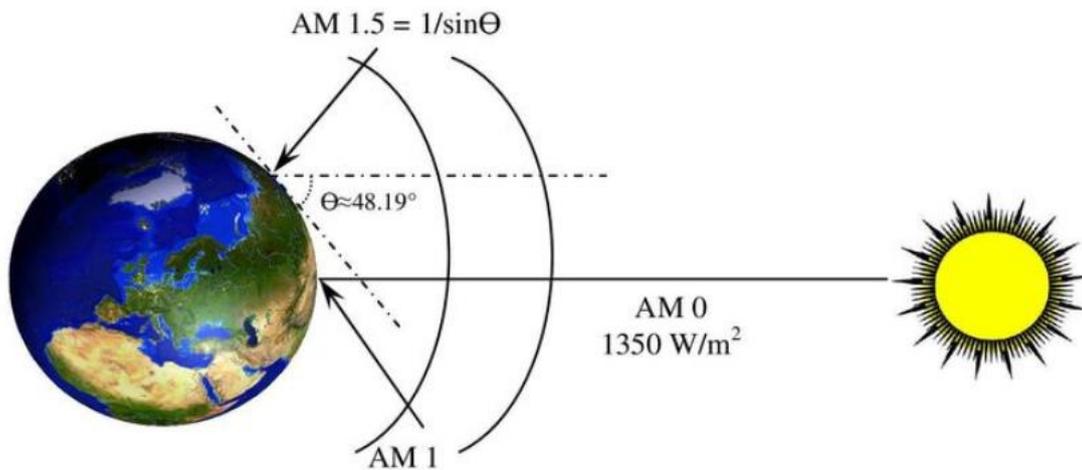


Figure II.1 : Rayonnement solaire mesuré au sommet de l'atmosphère au niveau de mer



**Figure II.2 :** Normes de mesures du spectre d'énergie émis par le soleil, notion de la convention AM

#### II.4 Conversion photovoltaïque

La conversion de l'énergie solaire en énergie électrique repose sur l'effet photoélectrique, c'est-à-dire sur la capacité des photons à créer des porteurs de charge (électrons et trous) dans un matériau. Lorsqu'un semi-conducteur est illuminé avec un rayonnement de longueur d'onde appropriée (l'énergie des photons doit être au moins égale à celle du gap énergétique du matériau), l'énergie des photons absorbés permet des transitions électroniques depuis la bande de valence vers la bande de conduction du semi-conducteur, générant ainsi des paires électron-trou qui peuvent contribuer au transport du courant (photoconductivité) par le matériau lorsqu'on le polarise. Si on illumine maintenant une jonction PN, les paires électron-trou qui sont créées dans la zone de charge d'espace de la jonction sont immédiatement séparées par le champ électrique qui règne dans cette région, et entraîné dans les zones neutres de chaque côté de la jonction. Si le dispositif est isolé, il apparaît une différence de potentiel aux bornes de la jonction (photo tension) ; s'il est connecté à une charge électrique extérieure, on observe le passage d'un courant alors qu'on n'applique aucune tension au dispositif. C'est le principe de base d'une cellule photovoltaïque. [25]

#### II.5 Le solaire photovoltaïque

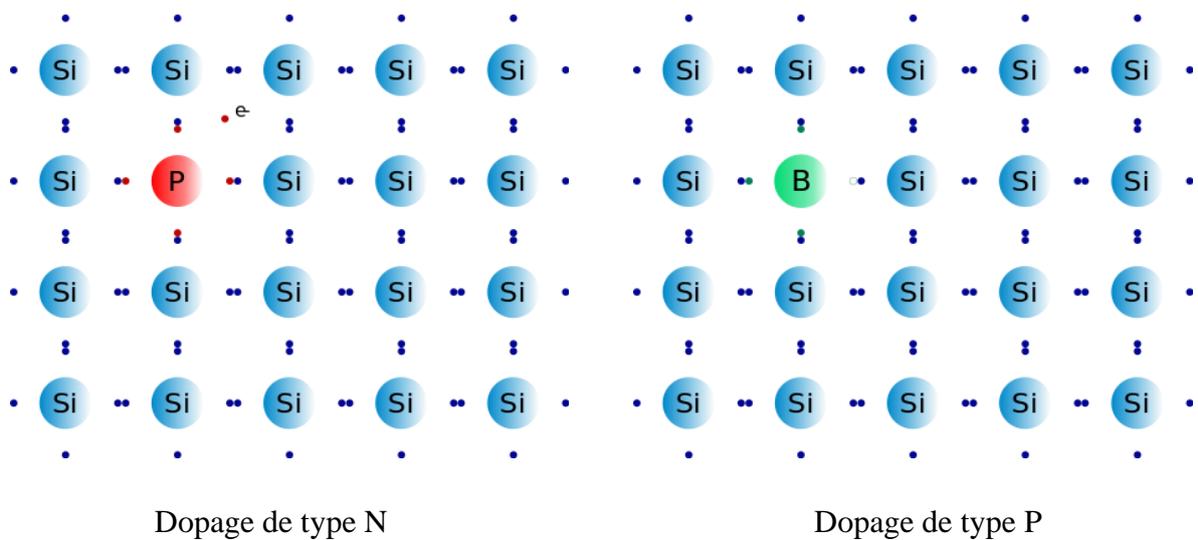
L'énergie solaire photovoltaïque est l'une des énergies renouvelables et la plus utilisée. Elle est transmise à la terre à travers l'espace sous forme de photons et de rayonnement

électromagnétique. Elle consiste à convertir directement le rayonnement solaire en électricité. Ont utilisant des convertisseurs photovoltaïques ou cellules photovoltaïques qui représentent l'élément de base dans la conversion photovoltaïque. [26]

Le mot « photovoltaïque », souvent abrégé par les lettres PV, a été formé à partir des mots « Photo », un mot grec signifiant lumière, et « Volta », le nom du physicien italien Alessandro Volta, qui a inventé la pile électrochimique en 1800. L'effet photovoltaïque, c'est la conversion directe de l'énergie solaire en électricité. Les cellules solaires photovoltaïques sont des semi-conducteurs capables de convertir directement la lumière en électricité. Cette conversion, appelée photovoltaïque, a été découverte par E. Becquerel en 1839. [27]

## II.6 L'effet photovoltaïque

Par définition, les semi-conducteurs sont des matériaux solides qui ont une conductivité électrique entre les conducteurs et les isolateurs. Ces matériaux se comportent comme des isolants à basses températures et comme conducteurs aux températures hautes. Le silicium, un bon exemple de semi-conducteur, est très utilisé dans la fabrication de cellules photovoltaïques. [28] L'atome du silicium a quatre électrons dans la dernière couche de valence et lorsqu'il est pur, il n'a pas d'électrons libres, donc il n'est pas d'un bon conducteur électrique. Pour modifier ce paramètre d'autres éléments sont ajoutés au réseau de silicium cristallin pur, ce processus est appelé dopage électronique et les impuretés (au moyen du Phosphore et du Bore) ajoutées dépendent de la nature semi-conductrice [29]. Avec l'ajout de Phosphore, qui a cinq électrons dans la dernière couche de valence, la structure silicium est modifiée, qui ont maintenant un électron libre, ou silicium porteur de charges négatives (silicium de type N). Cet électron dans l'excès ne pourra pas se coupler sur le réseau, attaché de manière lâche au noyau et avec une petite quantité d'énergie, ce qui permet aux porteurs de charge de passer de la bande de valence à la bande de conduction [30]. Dans le cas contraire, en introduisant des atomes avec seulement trois électrons de liaison, comme c'est le cas du Bore, il manquera un électron pour satisfaire les liaisons avec les atomes de silicium. Ce manque d'électron est appelé trou et il arrive qu'avec peu d'énergie thermique, un électron d'un site voisin puisse se déplacer vers cette position, provoquant le trou. Par conséquent, on dit que le bore est un accepteur d'électrons ou un dopant type P. [31]



**Figure II.3** : Présente le silicium de type N et le silicium de type P

## II.7 Cellules photovoltaïques

### II.7.1 Définition

Les cellules solaires sont des composants optoélectroniques qui transforment directement la lumière solaire en électricité par un processus appelé « effet photovoltaïque », car le soleil est la source lumineuse la plus intense de notre planète, et la plupart des applications sont en plein air. Mais certains préfèrent l'expression énergie lumière, pour insister sur le fait que toute source de lumière, même artificielle (ampoules, tubes fluorescents...), peut générer de l'électricité à travers une cellule solaire [32].

### II.7.2 Type des cellules photovoltaïque

Il existe différentes technologies pour la fabrication de cellules solaires et chacune d'elles a ses propres caractéristiques. Parmi les plus utilisées, sont ceux à base de silicium (monocristallin, poly cristallin ou amorphe), mais de nouvelles technologies sont testées chaque jour pour obtenir des rendements plus élevés à moindre coût [28]

#### II.7.2.1 Silicium monocristallin

Le silicium cristallin est actuellement l'option la plus populaire pour les cellules commerciales, bien que beaucoup d'autres matériaux soient disponibles. Le terme « cristallin » implique que tous les atomes dans le matériau PV actif font partie d'une structure cristalline simple où il n'y a aucune perturbation dans les arrangements ordonnés des atomes [33].

Les Panneaux monocristallins ce sont celles qui ont le meilleur rendement mais aussi celle qui ont le cout le plus élevé, du fait d'une fabrication compliquée. Ils se reconnaissent par une couleur plus uniforme et foncée et ils résultent de l'assemblage de plusieurs cellules [34].



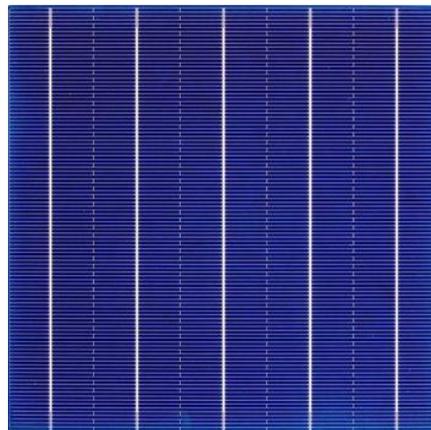
**Figure II.4** : Cellule silicium monocristallin

### II.7.2.2 Silicium polycristallin

Il est composé de petits grains de silicium cristallin. Les cellules à base de silicium polycristallin sont moins efficaces que les cellules à base de silicium monocristallin. Les joints de grains dans le silicium polycristallin gênent l'écoulement des électrons et réduisent le rendement de puissance de la cellule. L'efficacité de conversion PV pour une cellule à base de silicium poly cristallin modèle commercial s'étend entre 10 et 14% [35].

### II.7.2.3 Les cellules polycristallins

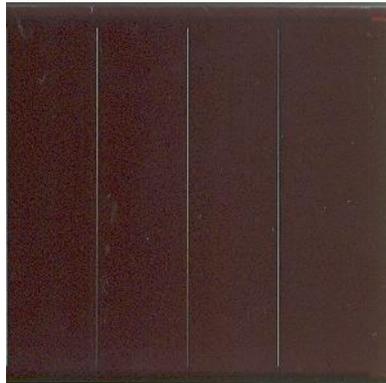
Ils se reconnaissent par une couleur bleutée et non uniforme. Leur fabrication à partir de copeaux de silicium utilise toute la surface du panneau, il n'y a donc pas d'espaces d'assemblage comme dans les panneaux monocristallins [36].



**Figure II.5** : Cellule silicium Polycristallin**II.7.2.4 Silicium amorphe (a-si)**

Le silicium amorphe est déposé en couche mince sur une plaque de verre ou un autre support souple. L'organisation irrégulière de ses atomes lui confère en partie une mauvaise semi conduction. Les cellules amorphes sont utilisées partout où une solution économique est recherchée ou lorsque très peu d'électricité est nécessaire, par exemple pour l'alimentation des montres, des calculatrices, ou des luminaires de secours. Elles se caractérisent par un fort coefficient d'absorption, ce qui autorise de très faibles épaisseurs, de l'ordre du micron. Par contre son rendement de conversion est faible (de 7 à 10 %) et les cellules ont tendance à se dégrader plus rapidement sous la lumière [36].

Les panneaux silicium amorphe ou couche mince : Apparus plus récemment, ces panneaux se reconnaissent par une couleur grise à marron foncé et uniforme avec un grain très fin. Cette uniformité est due à un procédé de fabrication de dépôt du silicium en couches minces [37].

**Figure II.6** : cellule silicium amorphe

Voici un tableau comparatif des technologies les plus utilisées dans les cellules photovoltaïques : [38] [39]

Matériau	Rendement	Longévité	Caractéristiques	Principales utilisations
Silicium mono Cristallin	15 à 20 % STC (26,7% en laboratoire)	20 à 30 ans	<ul style="list-style-type: none"> <li>• Très performant.</li> <li>• Stabilité de production.</li> <li>• Méthode de production coûteuse et laborieuse.</li> </ul>	Aérospatiale, Modules pour toits, façades...
Silicium poly Cristallin	12 à 17 % STC (21,9 % en laboratoire)	20 à 30 ans	<ul style="list-style-type: none"> <li>• Adapté à la production à grande échelle.</li> <li>• Stabilité de production.</li> <li>• Plus de 50% du marché mondial.</li> </ul>	Modules pour toits, façades, générateurs...
Amorphe	5 à 10 % STC (13% en laboratoire)	< 10 ans	<ul style="list-style-type: none"> <li>• Peut fonctionner sous la lumière fluorescente.</li> <li>• Fonctionnement si faible luminosité.</li> <li>• Fonctionnement par temps couvert.</li> <li>• • Fonctionnement si ombrage partiel • La puissance de sortie varie dans le temps. En début de vie, la puissance délivrée est de 15 à 20% supérieure à la valeur nominale et se stabilise après quelques mois.</li> </ul>	Appareils électroniques (montres, calculatrices...), intégration dans le bâtiment
Composite Mono cristallin (GaAs)	25-28.8 % STC (30 % en laboratoire)		<ul style="list-style-type: none"> <li>• Lourd, fissure facilement.</li> </ul>	Systèmes de concentrateurs Aérospatiale (satellites)

Composite poly Cristallin (CdS, CdTe, CulnGaSe2,	10 à 16 % (21% en laboratoire)		<ul style="list-style-type: none"> <li>Nécessite peu de matériaux mais certains contiennent des substances polluantes</li> </ul>	Appareils électroniques (montres, calculatrices...), intégration dans le bâtiment
---	-----------------------------------	--	--	---

**Tableau II.1:** Tableau comparatif des différentes technologies les plus utiliser d'une cellule photovoltaïque.

## II.8 Modélisation d'un système photovoltaïque

La modélisation du générateur photovoltaïque est le processus de représentation mathématique ou informatique du comportement et des caractéristiques d'un système photovoltaïque. Elle vise à prédire la production d'énergie électrique du générateur en fonction des conditions environnementales, des caractéristiques du panneau solaire et des autres composants du système.

### II.8.1 Modèle de la cellule solaire

La cellule solaire est l'élément de base d'un générateur photovoltaïque. La modélisation du comportement électrique du module photovoltaïque peut être étendue au générateur photovoltaïque selon sa configuration, nombre de modules en série et en parallèle. [40]

- **Le courant de court-circuit  $I_{cc}$**

Il définit la quantité des paires électron-trou créé et qui traverse la jonction sans recombinaison entre les longueurs de diffusion des électrons et trous ( $L_n$  et  $L_p$ ) respectivement

$$I_{cc} = qG(L_n + L_p) \quad \text{II.2}$$

$I_{cc}$  : Courant de court circuit de la cellule.

- **La tension en circuit ouvert  $V_{oc}$**

C'est la tension pour laquelle la diode à l'obscurité, fournit un courant nul. Elle est obtenue à partir de l'équation :

$$V_{oc} = V_{th} \ln \frac{I_{ph}}{I_s} \quad \text{II.3}$$

$$V_{th} = \frac{AKT_c}{q} \quad \text{II.4}$$

$I_s$  : Courant de saturation de la diode

$V_{th}$  : Tension thermique.

A : Coefficient d'idéalité de la diode.

q : Charge d'un électron  $1.6 \cdot 10^{-19}$  C.

$T_c$  : Température de la jonction.

K : Constante de Boltzmann  $1.38 \cdot 10^{-23}$  (J / K)

- **Le facteur de forme FF**

Il définit l'efficacité de la cellule solaire, il est obtenu comme suit :

$$FF = \frac{V_m I_m}{V_{oc} I_{sc}} \quad \text{II.5}$$

$V_m$  : Tension de point de puissance maximale (V).

$I_m$  : Courant de point de puissance maximale (A).

- **Point de puissance maximale**

Est le point ( $V_m$ ,  $I_m$ ) où la puissance dissipée dans la charge est maximale.

$$P_m = V_m \cdot I_m \quad \text{II.6}$$

- **La puissance du flux lumineux incident**

$$P_{in} = S \cdot E \quad \text{II.7}$$

E : L'ensoleillement qui représente la puissance lumineuse reçue par unité de surface ( $W/m^2$ ).

S : La surface effective des cellules.

- **Température de jonction**

La température de la jonction  $T_c$  de la cellule est donnée par la relation suivante :

$$T_c = T_a + \frac{NOCT-20}{800} E \quad \text{II.8}$$

Avec :

NOCT : La température nominale de fonctionnement de la cellule solaire

$T_a$  : La température ambiante.

- **Le rendement de conversion d'énergie  $\eta$**

C'est le rapport de la puissance maximale générée et de la puissance du rayonnement solaire Incident

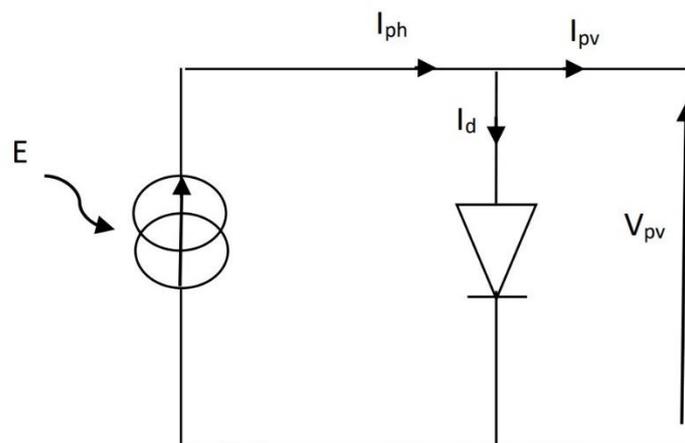
$$\eta = FF \frac{I_{cc} V_m}{P_{in}} \quad \text{II.9}$$

$$\text{Ou } \eta = \frac{I_m V_m}{S.E} \quad \text{II.10}$$

## II.8.2 Les modèles mathématiques

On rencontre dans la littérature plusieurs modèles du générateur photovoltaïque qui diffèrent entre eux par la procédure et le nombre de paramètres intervenants dans le calcul de la tension et du courant finals du générateur photovoltaïque [40].

### a) Cellule idéale



**Figure II.7** : schéma électrique équivalent d'une cellule photovoltaïque idéale.

$$I = I_{ph} + I_d \quad \text{II.11}$$

$$I_d = I_s e^{\left(\frac{V_{pv}}{AKTc} - 1\right)} \quad \text{II.12}$$

I : Courant disponible

$I_{ph}$  : Courant produit par la photopile

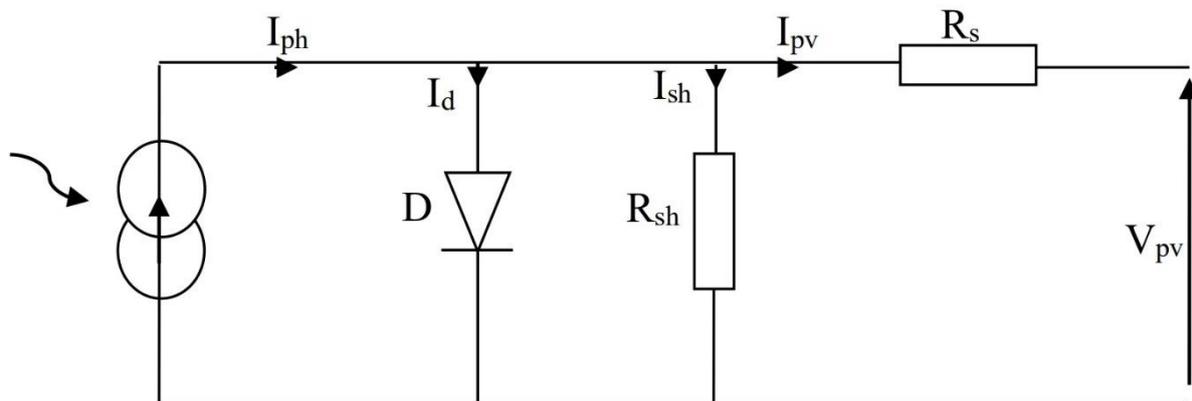
V : Tension aux bornes de la cellule

$\frac{KT}{q} = 26 \text{ mV}$  à  $T_c = 300 \text{ °K}$  pour le silicium polycristallin.

$I_s$  : Courant de polarisation de la diode ou courant de saturation

### b) Modèle à une diode

Le modèle à une diode est le plus utilisé et il donne de très bonnes précisions. La conversion de l'énergie de la lumière en énergie électrique est modélisée par un générateur de courant



**Figure II.8** : Schéma électrique équivalent du modèle à une diode.

Le courant généré par le module est donné par la loi de Kirchhoff :

$$I_{pv} = I_{ph} - I_d - I_{sh} \quad \text{II.13}$$

Tel que :

$I_{pv}$  : Le courant délivré par le module.

$I_{ph}$  : Photo courant.

$I_d$  : Courant de diode.

$I_{sh}$  : Le courant shunt.

Le courant produit par la photopile est donné par :

$$I_{ph} = P_1 E [1 + P_2 (E - E_{ref}) + P_3 (T_c - T_{cref})] \quad \text{II.14}$$

Tel que :

$E_{ref}$  : Ensoleillement de référence

$T_c$  : Température absolue de la cellule

$T_{cref}$  : Température de référence

Le courant de la diode est donné par :

$$I_d = I_s e^{\frac{q(V+R_s I_{pv})}{A n_s K T_j} - 1} \quad \text{II.15}$$

Avec :

$$I_s = P_4 T_c^3 e^{\left(-\frac{E_g}{K T_c}\right)} \quad \text{II.16}$$

Avec :  $E_g = 1.1 \text{ eV}$

$E_g$  : L'énergie de gap (paramètre caractéristique du matériau, pour le Silicium).

$I_s$  : Courant de saturation de la diode ou courant à l'obscurité.

$I_{sh}$  : Le courant de la résistance shunt est donné par :

$$I_{sh} = \frac{(v+R_s I_{pv})}{R_{sh}} \quad \text{II.17}$$

Tel que :

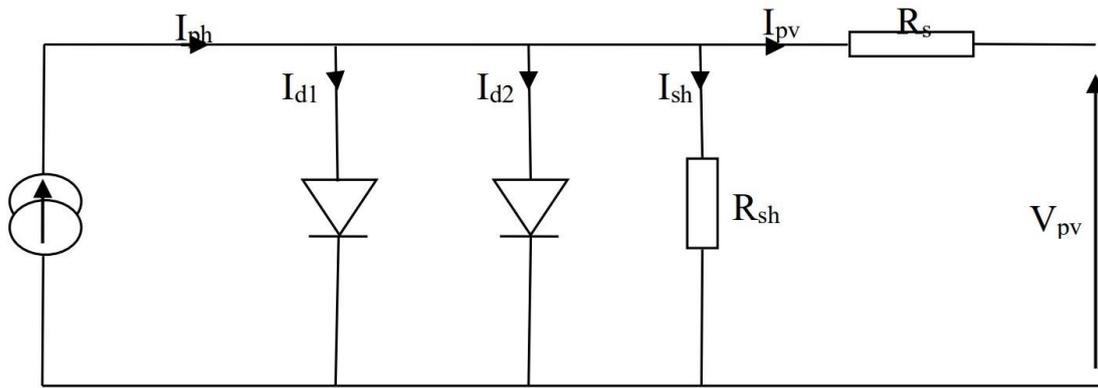
$R_s$  : Résistance série qui est due aux différentes résistances électriques que le courant rencontre sur son parcours (résistance de contacte).

$R_{sh}$  : Résistance shunt qui prend en compte les fuites inévitables de courant qui intervient entre les bornes opposées positive et négative d'une cellule. D'où l'expression de la caractéristique courant-tension (I-V) qui est donnée par l'équation suivante :

$$I_{pv} = P_1 E \left[ 1 + P_2 (E - E_{ref}) + P_3 (T_c - T_{cref}) \right] - \frac{V + R_s I}{R_{sh}} - P_4 T_c^3 e^{\left( -\frac{E_g}{KT_c} \right)} e^{\left( \frac{q(V + R_s I)}{AKT_c} - 1 \right)} \quad \text{II.18}$$

**c) Modèle à deux diodes**

Ce modèle est représenté par son schéma électrique équivalent qui se compose d'une source de courant modélisent le flux lumineux, les pertes sont modélisées par deux résistances, une résistance shunt  $R_{sh}$ , une résistance série  $R_s$  et deux diodes pour la polarisation de la cellule et le phénomène de la recombinaison des porteurs minoritaires. L'avantage de ce modèle c'est qu'il établit en appliquant seulement des données standard.



**Figure II.9** : Schéma électrique équivalent du modèle deux diodes.

Le courant généré par le module est donné par la loi de Kirchhoff

$$I_{pv} = P_1 E \left[ 1 + P_2 (E - E_{ref}) + P_3 (T_c - T_{cref}) \right] - P_4 T_c^3 e^{\left( -\frac{E_g}{KT_c} \right)} e^{\left( \frac{q(V + R_s I_{pv})}{A n_s K T_c} - 1 \right)} - P_5 T_c^3 e^{\left( -\frac{E_g}{2KT_c} \right)} e^{\left( \frac{q(V + R_s I_{pv})}{2 A n_s K T_c} - 1 \right)} - \frac{V_{pv} + R_s I_{pv}}{R_{sh}} \quad \text{II.19}$$

**II.9 Influence de la température et de l'éclairement**

La caractéristique d'une cellule PV (ou d'un générateur PV) est directement dépendante de l'éclairement et de la température.

**II.9.1 L'éclairement**

Les variations du courant et de la puissance en fonction de la tension pour différents niveaux d'éclairements à température maintenue constante 25°C, Figure II.7, montrent

clairement l'existence de maxima sur les courbes de puissance correspondant aux Points de Puissance Maximale  $P_{max}$ . Lorsque l'irradiation varie pour une température donnée, le courant de court-circuit  $I_{cc}$  varie proportionnellement à l'irradiation. Dans un même temps, la tension de circuit ouvert  $V_{co}$  (à vide) varie très peu. [41]

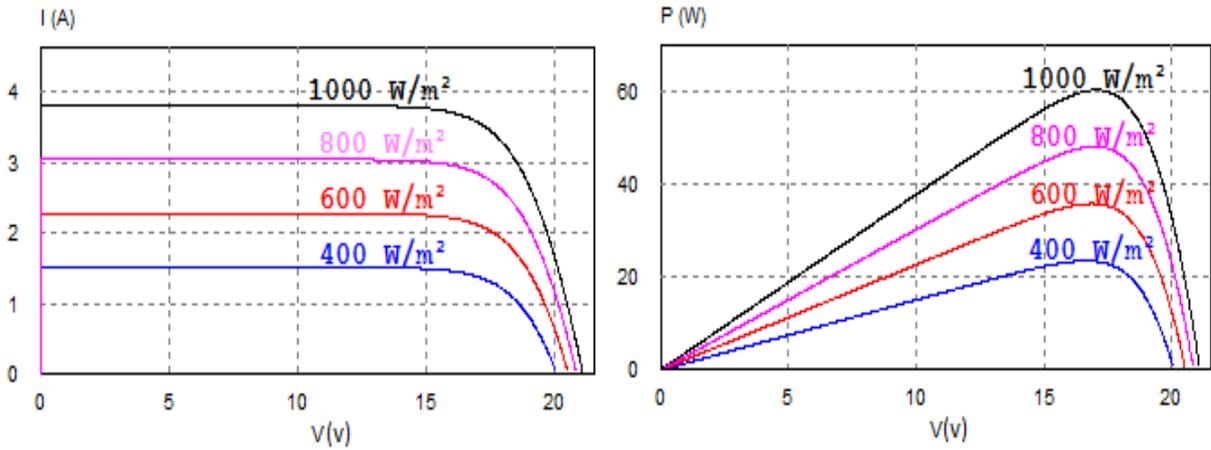


Figure II.10 : Evolution de la caractéristique I(V) et P(V) en fonction de l'irradiation.

### II.9.2 La température

La température est un paramètre très important dans le comportement des cellules solaires. La température a également une influence sur la caractéristique d'un générateur PV. La Figure 2 présente la variation des caractéristiques d'une cellule PV en fonction de la température à un éclairement donné. L'éclairement est ici fixé à  $1000 \text{ w.m}^{-2}$ .

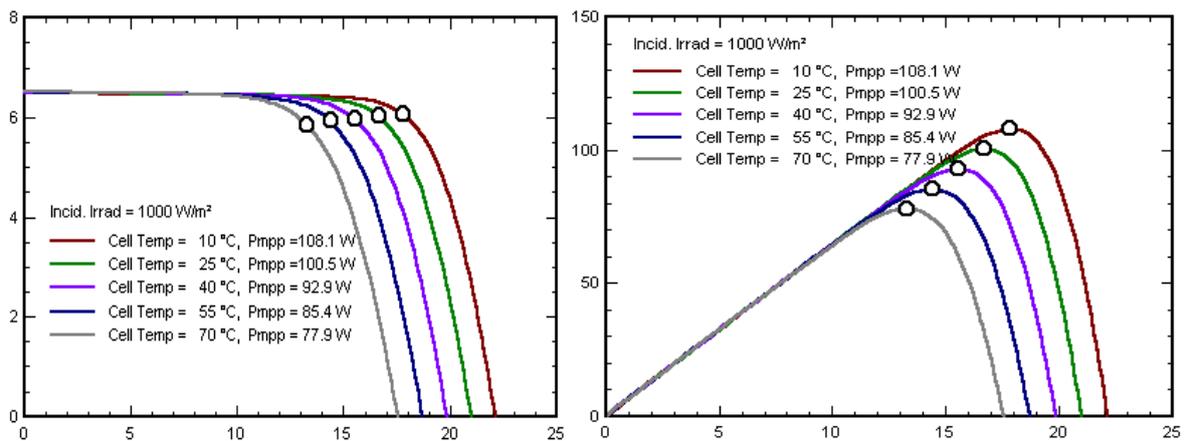


Figure II.11 : Evolution de la caractéristique I(V) et P(V) pour différentes températures.

Par contre, si la température croît à irradiation constante, la tension à vide  $V_{co}$  décroît avec la température. Plus la température est élevée plus  $V_{co}$  est faible et le courant de court-circuit  $I_{cc}$  augmente avec la température. Cette hausse est nettement moins importante que la baisse de tension. L'influence de la température sur  $I_{cc}$  peut être négligée dans la majorité des cas.

Donc la température et l'éclairement sont donc les deux principaux paramètres qui vont modifier la caractéristique d'un générateur PV. Ces deux paramètres devront donc être étudiés avec soin lors de la mise en place d'une installation PV. [41]

## II.10 Avantages et inconvénients de l'énergie photovoltaïque

### II.10.1 Les avantages

- Une haute fiabilité, elle n'est pas composée par des pièces mobiles qui la rendent particulièrement appropriée aux régions isolées.
- Les panneaux photovoltaïques ont le caractère modulaire qui permet un montage simple et adaptable à des besoins énergétique divers. Ils résistent à des applications de grande puissance.
- L'énergie photovoltaïque est une source d'énergie gratuite qui n'a pas besoin ni de combustion ni transport.
- Une très grande qualité sur le plan écologique car il n'est pas polluant, silencieux et il ne comporte aucune perturbation sur le milieu [42].

### II.10.2 Les inconvénients

- Une haute technologie pour la fabrication et requiert des investissements d'un coût élevé.
- Un faible rendement de conversion d'un module (la limite théorique pour une cellule au silicium cristallin 28%).
- Les générateurs photovoltaïques ne sont pas compétitifs par rapport aux générateurs diesel que pour des faibles demandes d'énergie en région isolée.
- La fiabilité et les performances du système restent équivalentes pour tant qu'une batterie et les composants de régulation associés soient judicieusement choisis. Les installations de grandes dimensions s'occupent un grand espace [42].

**II.11 Conclusion**

Dans ce chapitre, Nous avons rappelé quelques notions sur les systèmes photovoltaïques. Il présente les bases de l'énergie solaire, ses différentes formes et les composants des systèmes photovoltaïques. Il aborde également la modélisation des systèmes photovoltaïques et l'influence de la température et de l'éclairement solaire sur leurs performances.

## **Chapitre III**

# **La prévision météorologique et le langage de développement**

### **III.1 Introduction**

La production d'électricité à partir de systèmes photovoltaïques (PV) dépend étroitement des conditions météorologiques, notamment de l'irradiance solaire et de la température. La capacité à prédire avec précision ces deux variables clés est essentielle pour optimiser la gestion des systèmes PV et maximiser leur rendement énergétique. Dans ce contexte, l'utilisation de techniques de prédiction météorologique avancées, telles que l'apprentissage automatique et l'apprentissage profond, offre de nouvelles opportunités pour améliorer la gestion des systèmes PV.

Pour mettre en œuvre ces techniques, il est souvent nécessaire d'utiliser des outils logiciels tels que Python et ses bibliothèques dédiées à l'apprentissage automatique et à l'analyse de données. Python est un langage de programmation populaire et polyvalent qui offre une grande flexibilité et une large gamme de bibliothèques spécialisées, telles que TensorFlow, Keras, Scikit-learn et PyTorch, qui sont largement utilisées pour développer des modèles de prévision météorologique.

### **III.2 Définition de la prévision météorologiques**

La prévision météorologique pour la gestion de systèmes PV consiste à estimer les conditions météorologiques futures, telles que l'irradiance solaire, la température et d'autres paramètres pertinents, afin d'optimiser l'efficacité et la performance des systèmes photovoltaïques. Cette prévision permet aux gestionnaires de systèmes PV de prendre des décisions éclairées sur la planification de la production d'électricité, l'ajustement des paramètres des panneaux solaires, la gestion de la charge des batteries et d'autres stratégies d'exploitation pour maximiser l'utilisation de l'énergie solaire disponible. En utilisant des techniques avancées telles que l'apprentissage automatique et l'apprentissage profond, les prévisions météorologiques peuvent être améliorées en capturant les relations complexes entre les variables météorologiques et les performances des systèmes PV ce qui permet une gestion plus efficace et une optimisation du rendement énergétique.

### **III.3 L'impact des conditions météorologiques sur l'efficacité des systèmes PV**

#### **III.3.1 L'irradiance solaire**

L'irradiance solaire représente la quantité de rayonnement solaire atteignant les panneaux solaires, et son niveau varie en fonction des conditions météorologiques telles que

l'ensoleillement, la couverture nuageuse et les variations saisonnières. La prédiction précise de l'irradiance solaire à court terme permet aux gestionnaires de systèmes PV d'anticiper les variations de la production d'énergie solaire, ce qui leur permet de prendre des décisions éclairées en temps réel pour optimiser l'utilisation de l'énergie solaire disponible.

### **III.3.2 La température**

En plus de l'irradiance solaire, la température ambiante a également un impact significatif sur les performances des systèmes PV. L'élévation de la température des panneaux solaires entraîne une diminution de leur efficacité, ce qui réduit la production d'électricité. Par conséquent, la prédiction précise de la température ambiante permet aux gestionnaires de systèmes PV de mettre en place des stratégies de refroidissement adéquates pour minimiser l'impact négatif de la chaleur sur les performances des panneaux solaires.

### **III.4 Les méthodes de prévision météorologique**

Les méthodes traditionnelles de prédiction météorologique reposent souvent sur des modèles numériques basés sur des équations physiques simplifiées. Bien que ces approches soient utiles pour fournir des prévisions à plus long terme, elles peuvent être limitées en termes de résolution temporelle et spatiale, ainsi que de modélisation précise des processus météorologiques complexes. En revanche, l'apprentissage automatique et l'apprentissage profond offrent la possibilité d'apprendre directement à partir de données historiques pour capturer les relations non linéaires et les interactions complexes entre les variables météorologiques, telles que l'irradiance solaire et la température, et les performances des systèmes PV.

Pour atteindre cet objectif, ce mémoire se concentrera sur le développement de modèles prédictifs basés sur l'apprentissage automatique et l'apprentissage profond, en utilisant des données météorologiques historiques et en intégrant des variables telles que l'irradiance solaire, la température, la pression atmosphérique, l'humidité, la vitesse du vent et la couverture nuageuse. Ces modèles seront entraînés sur des ensembles de données volumineux et diversifiés afin d'apprendre les motifs complexes et les relations entre ces variables météorologiques et les performances des systèmes PV.

### **III.5 Horizons de prévision**

#### **III.5.1 Les prévisions à court terme**

Les prévisions météorologiques à court terme jouent un rôle crucial dans la gestion des systèmes photovoltaïques (PV). Elles permettent d'estimer avec précision les conditions météorologiques à venir, notamment l'irradiance solaire, la température et d'autres paramètres pertinents pour la performance des systèmes PV. Ces prévisions à court terme, qui englobent généralement des horizons de quelques minutes à quelques jours, revêtent une importance primordiale dans les opérations quotidiennes du marché de l'énergie solaire.

#### **III.5.2 Les prévisions à moyen terme**

Les prévisions météorologiques à moyen terme ont généralement une durée de quelques jours à plusieurs semaines à venir, selon les besoins et les pratiques spécifiques. Elles sont conçues pour fournir des informations météorologiques sur une période plus étendue que les prévisions à court terme.

#### **III.5.3 Les prévisions à long terme**

Les prévisions à long terme, dont les délais sont mesurés en mois, en trimestres ou même en années. Les prévisions météorologiques à long terme comportent généralement une précision moindre par rapport aux prévisions à court et moyen terme. Cela est dû à plusieurs facteurs, tels que la complexité des interactions climatiques à grande échelle, les incertitudes liées aux modèles climatiques et la difficulté de prévoir avec précision les variations climatiques sur de longues périodes.

### **III.6 Avantages de la prédiction météorologique**

L'utilisation des prévisions météorologiques dans la gestion des systèmes photovoltaïques (PV) offre plusieurs avantages significatifs :

- a) Optimisation de la production d'énergie : Grâce à l'analyse des conditions météorologiques anticipées, les gestionnaires peuvent ajuster les stratégies de gestion de l'énergie solaire pour maximiser la production d'électricité. En utilisant des prévisions météorologiques précises, notamment en ce qui concerne l'irradiance solaire, la température et d'autres paramètres, ils peuvent optimiser les opérations des systèmes PV.

- b) Planification efficace : Les prévisions météorologiques permettent une planification à court, moyen et long terme en fonction des conditions météorologiques attendues. Cela aide les gestionnaires à organiser les opérations, à planifier les charges et les décharges des batteries, et à prendre des décisions éclairées pour maximiser l'utilisation de l'énergie solaire.
- c) Gestion proactive des risques : Les prévisions météorologiques aident à identifier les conditions météorologiques extrêmes potentielles, telles que les tempêtes ou les chutes de neige, permettant aux gestionnaires de prendre des mesures préventives pour minimiser les perturbations et les pertes associées à la production d'énergie solaire.
- d) Prévention des pertes financières : En anticipant les variations de la production d'énergie solaire à partir des prévisions météorologiques, les gestionnaires peuvent prendre des mesures pour minimiser les pertes financières résultant d'une production insuffisante ou d'une surproduction d'électricité.
- e) Planification stratégique basée sur les tendances : Les prévisions météorologiques à plus long terme aident les gestionnaires à prendre des décisions stratégiques en matière de dimensionnement des installations, d'investissements futurs et de choix de site. En tenant compte des tendances climatiques, ils peuvent optimiser les performances des systèmes PV sur une période prolongée.

### **III.7 Défis dans les prévisions météorologique pour la gestion des systèmes PV**

- a) Précision des données d'entrée : Les prévisions météorologiques dépendent de la disponibilité de données précises et fiables, telles que les mesures d'irradiance solaire et de température. La précision de ces données est essentielle pour garantir des prévisions météorologiques précises. Cependant, la collecte et la qualité de ces données peuvent varier en fonction des sources et des emplacements, ce qui peut entraîner des incertitudes dans les prévisions.
- b) Complexité des interactions climatiques : Les systèmes climatiques sont extrêmement complexes et impliquent de nombreuses interactions entre les variables météorologiques. Il est difficile de modéliser avec précision ces interactions à long terme, ce qui limite la capacité à faire des prévisions météorologiques fiables sur de plus longues périodes.
- c) Incertitude des modèles climatiques : Les modèles climatiques utilisés pour les prévisions à long terme sont basés sur des hypothèses et des approximations qui introduisent des

incertitudes. La variabilité naturelle du climat et les événements extrêmes peuvent également rendre les prévisions à long terme plus incertaines.

- d) Variabilité spatiale et régionale : Les conditions météorologiques peuvent varier considérablement d'un endroit à l'autre, ce qui nécessite des modèles et des prévisions adaptés à des régions spécifiques. Les données locales et la modélisation fine échelle sont nécessaires pour tenir compte de cette variabilité spatiale et régionale.
- e) Sensibilité aux changements climatiques : Les prévisions météorologiques pour la gestion des systèmes PV doivent également tenir compte des changements climatiques à long terme. Les schémas météorologiques peuvent changer en raison du réchauffement climatique et des modifications des conditions atmosphériques, ce qui ajoute une complexité supplémentaire aux prévisions à long terme.
- f) Adaptation aux conditions en temps réel : Les prévisions météorologiques à court terme doivent être mises à jour régulièrement pour refléter les conditions en temps réel. L'adaptation rapide aux changements météorologiques est essentielle pour une gestion efficace des systèmes PV et pour prendre des décisions opérationnelles en temps réel.

## III.8 Le langage de développement (Python)

### III.8.1 Définition

Python est l'un des langages de programmation les plus populaires et largement utilisés dans le domaine de l'apprentissage automatique (machine Learning) et du Deep Learning.



**Figure III.1** : Langage de programmation python

### III.8.2 Les avantages de Python

Les raisons pour lesquelles Python est souvent préféré pour ces tâches :

- 1. Large écosystème** : Python dispose d'une vaste bibliothèque standard ainsi que d'un écosystème riche en bibliothèques tierces dédiées à l'apprentissage automatique et au Deep Learning. Des bibliothèques telles que NumPy, Pandas, Scikit-learn et TensorFlow offrent des fonctionnalités puissantes pour manipuler les données, créer des modèles d'apprentissage automatique et entraîner des réseaux de neurones.
- 2. Syntaxe claire et concise** : Python se distingue par sa syntaxe simple, lisible et expressive, ce qui facilite la compréhension du code. Cela rend le processus de développement, de débogage et de maintenance des modèles d'apprentissage automatique plus fluide et efficace.
- 3. Flexibilité** : Python est un langage polyvalent qui peut être utilisé pour une variété de tâches, allant de la manipulation des données à la visualisation, en passant par le déploiement des modèles. Cette flexibilité permet aux praticiens d'expérimenter et de développer des solutions complètes en utilisant un seul langage.
- 4. Communauté active** : Python bénéficie d'une communauté d'utilisateurs très active et engagée, ce qui se traduit par une disponibilité abondante de ressources en ligne, de forums de discussion, de tutoriels et de codes sources ouverts. Cela facilite l'apprentissage, la résolution de problèmes et la collaboration avec d'autres chercheurs et praticiens.
- 5. Intégration aisée** : Python peut être intégré à d'autres langages de programmation et outils couramment utilisés dans le domaine de l'apprentissage automatique. Par exemple, il est possible d'utiliser des bibliothèques Python avec des outils tels que R, Java ou C++, ce qui offre une plus grande flexibilité dans le choix des technologies et des infrastructures.
- 6. Plateformes et frameworks populaires** : De nombreux frameworks populaires pour l'apprentissage automatique et le Deep Learning, tels que TensorFlow, Keras, PyTorch et Scikit-learn, sont développés en Python. Ces frameworks fournissent des API conviviales et des outils haut niveau qui simplifient le processus de création, d'entraînement et d'évaluation des modèles. <https://docs.python.org/fr>

### III.8.3 Les bibliothèques de python

Alors que La référence du langage Python décrit exactement la syntaxe et la sémantique du langage Python, ce manuel de référence de la Bibliothèque décrit la bibliothèque standard distribuée avec Python. Il décrit aussi certains composants optionnels typiquement inclus dans les distributions de Python.

La bibliothèque standard de Python est très grande, elle offre un large éventail d'outils comme le montre la longueur de la table des matières ci-dessous. La bibliothèque contient des modules natifs (écrits en C) exposant les fonctionnalités du système telles que les interactions avec les fichiers qui autrement ne seraient pas accessibles aux développeurs Python, ainsi que des modules écrits en Python exposant des solutions standardisées à de nombreux problèmes du quotidien du développeur. Certains de ces modules sont définis explicitement pour encourager et améliorer la portabilité des programmes Python en abstrayant des spécificités sous-jacentes en API neutres.

#### III.8.3.1 Pandas



**Figure III.2** : Bibliothèque pandas

Pandas est la bibliothèque la plus complète en ce qui concerne la manipulation de données. On peut comparer pandas à un “Excel sous stéroïdes” : elle permet de travailler avec :

- Des tableaux de données en deux dimensions (lignes et colonnes) appelés DataFrames,
- Des Panels, c'est-à-dire des ensembles de données en trois ou quatre dimensions !

Avec la bibliothèque Pandas, tu peux importer des données depuis un fichier .csv afin de les nettoyer (par exemple pour éliminer toutes les lignes vides), les transformer ou les compléter.

Pandas permet également d'effectuer des calculs statistiques sur tes données, par exemple avec la méthode.mean() qui permet d'obtenir la moyenne des valeurs contenues dans une colonne.

Le principal avantage de Pandas par rapport à Numpy est de permettre de nommer les colonnes et les lignes, ce qui offre bien plus de lisibilité dans tes projets.

### III.8.3.2 Numpy

Numpy contient des modules de gestion de données et de calcul. Elle permet de gérer facilement des bases de données : on les appelle les **numpy arrays**. Ce sont des listes, ou bien des listes de listes.



Figure III.3 : Bibliothèque Numpy

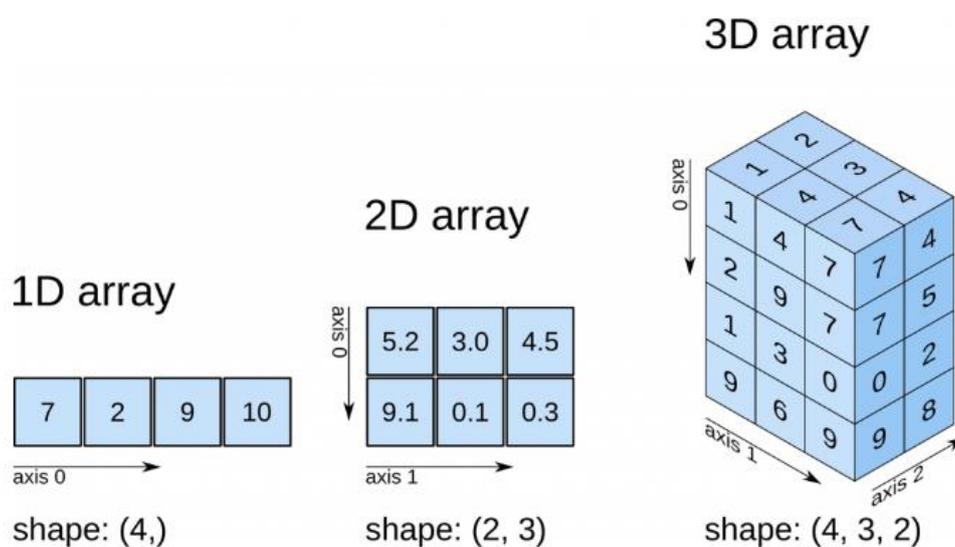


Figure III.4 : Les formes de matrices en numpy

L'avantage de Numpy est de pouvoir créer rapidement une base de données, avec des instructions simples que Python comprend. Par exemple, l'instruction `np.zeros(10)` renvoie une liste de dix chiffres, tous égaux à 0.

Ensuite, Numpy permet d'effectuer des opérations particulièrement rapidement. En effet, avec Numpy une opération effectuée sur un "array" s'applique à chaque terme de cet array. Cela évite de devoir exécuter une boucle FOR ou une boucle WHILE, ce qui est parfois une opération assez lente.

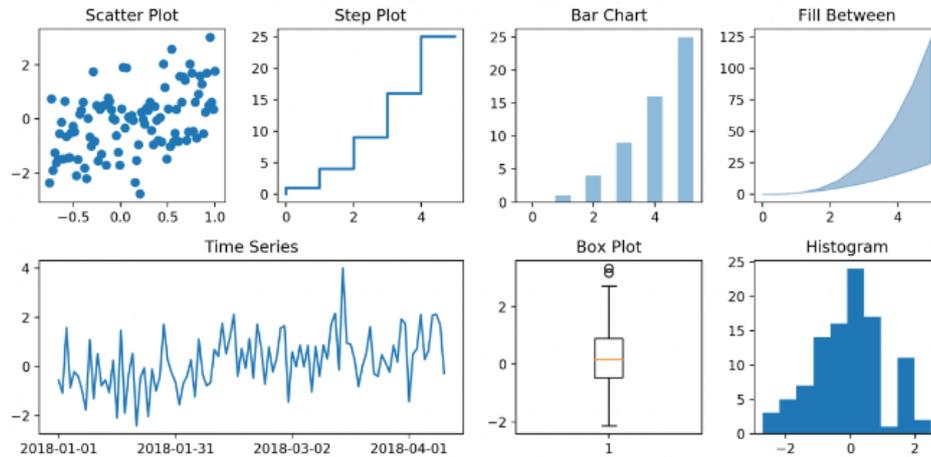
### III.8.3.3 Matplotlib



**Figure III.5 :** Bibliothèque matplotlib

Matplotlib est un module de dataviz très plaisant : c'est en effet lui qui te permet, en une ligne de code, de créer des graphiques qui modélisent les données sur lesquelles tu travailles. Matplotlib s'utilise en synergie avec Numpy ou Pandas.

Matplotlib offre une large variété de types de graphes qui s'adaptent à tous les besoins : histogrammes, boîtes à moustache, courbes, scatter plots, camemberts...



**Figure III.6:** Les types des graphes en Bibliothèque Matplotlib.

### III.8.3.4 Scikit learn



**Figure III.7 :** Bibliothèque scikit learn

Scikit learn, aussi appelée sklearn, est le module de machine learning le plus populaire. Cette bibliothèque permet d'effectuer d supervisé ou non-supervisé. Elle est également performante en modélisation statistique, particulièrement pour la classification et la régression.

Sklearn propose de choisir un modèle parmi différentes classes de modèles, puis de le paramétrer, de l'entraîner et enfin de le tester sur tes données.

Cette bibliothèque est appropriée pour les usages business et notamment pour la mise en place de modèles prédictifs. On peut citer Spotify, qui s'en sert pour ses algorithmes de prédiction musicale, ainsi que Booking.com pour définir ses recommandations d'hôtels.

### III.8.3.5 Keras

Keras est une bibliothèque open source pour la création d'applications de Deep learning. Keras est écrit en Python et fournit une interface commune pour différents backends de Deep learning tels que TensorFlow et Theano. Le Deep Learning fait partie du machine learning et se base sur les réseau de neurones artificiels.



**Figure III.8** : Bibliothèque Keras

### III.8.3.6 TensorFlow



**Figure III.9** : Bibliothèque TensorFlow

TensorFlow est le Framework de Deep Learning développé par Google. Comme PyTorch, il permet de développer des réseaux de neurones profonds.

TensorFlow donne accès à des fonctionnalités et des techniques de Deep Learning très avancées, mais on reproche souvent à ce Framework d'être un peu compliqué à prendre en main. Cette complexité est compensée par la grande flexibilité qu'offre TensorFlow, dès lors qu'on maîtrise l'outil.

## III.8.4 L'environnement de développement

### III.8.4.1 Le navigateur Anaconda

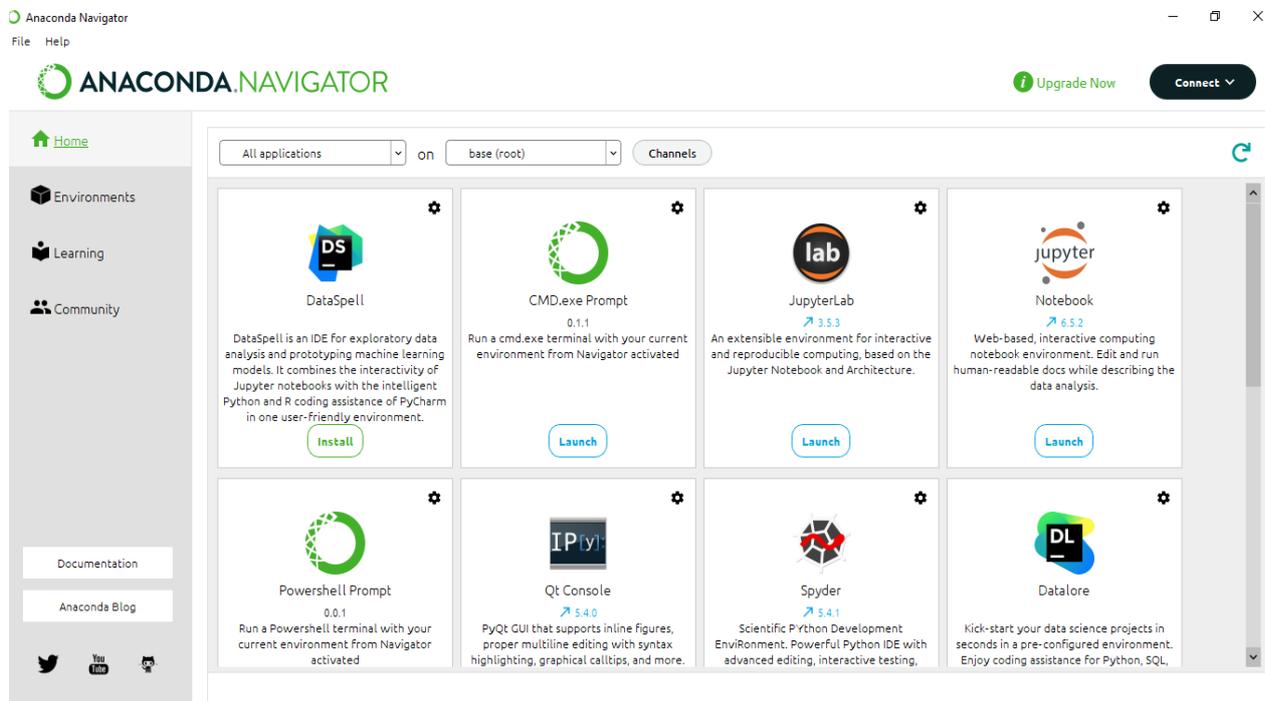


Figure III.10 : le navigateur Anaconda

Le navigateur Anaconda, également connu sous le nom d'Anaconda Navigator, est une interface graphique conviviale fournie avec la distribution Anaconda. Anaconda est une plateforme populaire pour la science des données et l'apprentissage automatique, qui facilite l'installation et la gestion des packages, des environnements virtuels et des outils liés à Python.

- ✓ **Environnements** : Vous pouvez créer et gérer des environnements virtuels isolés pour travailler sur différents projets. Ces environnements permettent de maintenir des configurations Python spécifiques et de gérer les dépendances des packages.
- ✓ **Packages** : Vous pouvez rechercher, installer et mettre à jour des packages Python directement à partir du navigateur Anaconda. Il propose une liste de packages couramment utilisés en science des données et en apprentissage automatique.
- ✓ **Outils** : Le navigateur Anaconda propose également des liens vers divers outils et applications couramment utilisés dans l'écosystème de la science des données. Cela inclut des éditeurs de code tels que Jupyter Notebook et JupyterLab, des environnements de développement intégrés (IDE) comme Spyder, ainsi que des applications pour la visualisation des données, la gestion des bases de données et plus encore.

- ✓ **Documentation** : Le navigateur Anaconda fournit également un accès direct à la documentation officielle d'Anaconda, aux tutoriels et aux ressources d'apprentissage pour faciliter l'utilisation de la plateforme.

### III.8.4.1.1 Jupyter

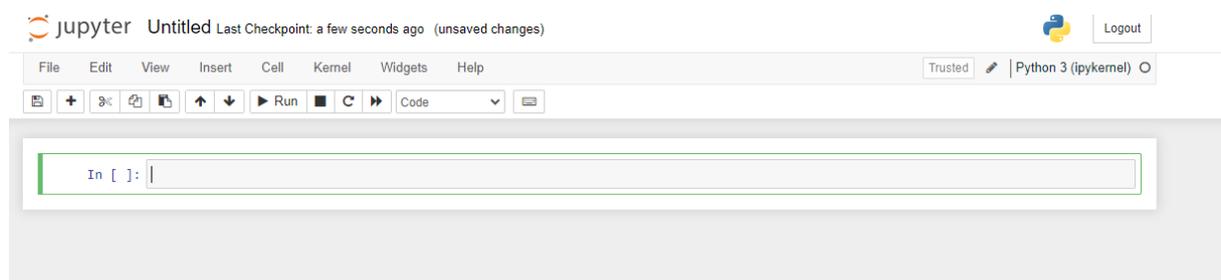
#### III.8.4.1.2 Définition

Jupyter Notebook est une application web open-source qui permet de créer et de partager des documents interactifs appelés "notebooks". Il est inclus dans la distribution Anaconda Navigator, qui est une plateforme populaire pour la gestion et l'utilisation d'environnements de développement pour le langage de programmation Python et d'autres langages.



**Figure III.11** : le notebook Jupyter

Un notebook Jupyter est composé de cellules contenant du texte (au format Markdown) ou du code (généralement du code Python). Les cellules de code peuvent être exécutées individuellement, ce qui permet une exploration interactive des données et une expérimentation rapide. Les cellules de texte permettent d'ajouter des explications, des instructions ou des commentaires au code. Il offre également des fonctionnalités avancées, telles que la visualisation de données, la création de graphiques interactifs, l'intégration d'images et de vidéos, l'exécution de commandes système, etc. Il permet également d'intégrer des équations mathématiques, des tableaux, des liens, des widgets interactifs et d'autres éléments interactifs dans les notebooks.



**Figure III.12** : la page de travail en Jupyter

L'interface web de Jupyter Notebook permet de créer, d'ouvrir, de modifier et de sauvegarder des notebooks localement. On peut également les partager avec d'autres utilisateurs en exportant les notebooks au format HTML, PDF ou via des services de partage en ligne.

### III.8.4.1.3 Description des jeux de données

<b>Nom et Objectif</b>	June data, Ils sont des données sur les irradiances et les températures																																				
<b>Source des données</b>	Un laboratoire en Malaisie																																				
<b>Statistiques sommaires</b>	<pre>June_data.describe()</pre> <table border="1"> <thead> <tr> <th></th> <th>Day</th> <th>Irradiance</th> <th>Temperature</th> </tr> </thead> <tbody> <tr> <td>count</td> <td>7775.000000</td> <td>7775.000000</td> <td>7775.000000</td> </tr> <tr> <td>mean</td> <td>14.000900</td> <td>138.909891</td> <td>32.351383</td> </tr> <tr> <td>std</td> <td>7.789478</td> <td>218.359044</td> <td>8.779683</td> </tr> <tr> <td>min</td> <td>1.000000</td> <td>0.000000</td> <td>22.720000</td> </tr> <tr> <td>25%</td> <td>7.000000</td> <td>0.000000</td> <td>25.830000</td> </tr> <tr> <td>50%</td> <td>14.000000</td> <td>4.000000</td> <td>28.390000</td> </tr> <tr> <td>75%</td> <td>21.000000</td> <td>193.730000</td> <td>37.610000</td> </tr> <tr> <td>max</td> <td>27.000000</td> <td>936.320000</td> <td>62.370000</td> </tr> </tbody> </table>		Day	Irradiance	Temperature	count	7775.000000	7775.000000	7775.000000	mean	14.000900	138.909891	32.351383	std	7.789478	218.359044	8.779683	min	1.000000	0.000000	22.720000	25%	7.000000	0.000000	25.830000	50%	14.000000	4.000000	28.390000	75%	21.000000	193.730000	37.610000	max	27.000000	936.320000	62.370000
	Day	Irradiance	Temperature																																		
count	7775.000000	7775.000000	7775.000000																																		
mean	14.000900	138.909891	32.351383																																		
std	7.789478	218.359044	8.779683																																		
min	1.000000	0.000000	22.720000																																		
25%	7.000000	0.000000	25.830000																																		
50%	14.000000	4.000000	28.390000																																		
75%	21.000000	193.730000	37.610000																																		
max	27.000000	936.320000	62.370000																																		
<b>Format du fichier</b>	Le format CSV (Comma-Separated Values) est largement utilisé en raison de sa simplicité, de sa compatibilité et de sa facilité d'utilisation																																				
<b>Licence et restrictions d'utilisation</b>	Ces données sont privées, seuls les étudiants et les chercheurs dans ce domaine ont le droit de l'obtenir.																																				

**Tableau III.1** : Description des données.

### III.8.4.1.4 Prétraitement des données (normalisation, nettoyage)

Le prétraitement des données est une étape essentielle dans le processus d'analyse des données et de développement de modèles d'apprentissage automatique. Il implique la manipulation, la transformation et la normalisation des données brutes afin de les rendre

adaptées à l'utilisation par des algorithmes d'apprentissage automatique ou dans l'apprentissage profond.

### **1- Nettoyage des données**

Les données brutes peuvent contenir des erreurs, des valeurs manquantes, du bruit ou des valeurs aberrantes. Le prétraitement permet de détecter et de corriger ces problèmes, ce qui garantit la qualité des données utilisées pour l'analyse.

### **2- Normalisation des données**

Les données peuvent être mesurées dans différentes échelles et unités, ce qui peut entraîner des biais indésirables dans l'analyse. La normalisation des données permet de mettre les variables à la même échelle, ce qui facilite la comparaison et l'interprétation des résultats.

#### **III.8.4.1.5 Séparation des ensembles de données (entraînement, test)**

La séparation des données en ensembles de formation (train) et de test est une pratique courante en apprentissage automatique et en science des données. Elle est effectuée dans le but d'évaluer les performances d'un modèle sur des données non vues lors de son entraînement.

Permet les raisons principales pour lesquelles nous séparons les données en ensembles d'entraînement et de test :

#### **1. Évaluation impartiale du modèle**

En utilisant un ensemble de test indépendant, nous pouvons obtenir une évaluation impartiale des performances du modèle. Cela nous permet de comprendre comment le modèle généralise sur de nouvelles données et d'estimer son efficacité réelle.

#### **2. Prévention du surapprentissage (overfitting)**

Si nous évaluons un modèle sur les mêmes données sur lesquelles il a été formé, il risque de surapprendre ces données spécifiques et d'avoir des performances optimistes. En séparant les données en ensembles de formation et de test, nous pouvons détecter si le modèle souffre de surapprentissage en observant une différence significative entre ses performances sur les ensembles de formation et de test.

#### **3. Tuning des hyperparamètres**

Lorsque nous ajustons les hyperparamètres d'un modèle, tels que la profondeur d'un arbre de décision ou le nombre d'itérations dans un algorithme d'apprentissage en profondeur, nous avons besoin d'un ensemble de validation supplémentaire. La séparation des données en

ensembles de formation, de validation et de test nous permet de régler les hyperparamètres sur l'ensemble de validation et de maintenir l'ensemble de test pour l'évaluation finale du modèle.

#### **4. Estimation des performances réelles**

En utilisant un ensemble de test distinct, nous obtenons une estimation plus réaliste des performances du modèle sur de nouvelles données. Cela nous donne une idée de l'exactitude attendue lorsque le modèle est utilisé dans des scénarios réels, ce qui est crucial pour prendre des décisions éclairées sur son déploiement et son utilisation.

En pratique, les données sont généralement divisées en trois ensembles : un ensemble de formation pour l'entraînement du modèle, un ensemble de validation pour le réglage des hyperparamètres et un ensemble de test pour l'évaluation finale. Cependant, la manière précise de diviser les données peut varier en fonction de la taille du jeu de données et des exigences spécifiques du problème.

### **III.8.5 Métriques de comparaison**

#### **III.8.5.1 MAE**

MAE signifie Mean Absolute Error, ce qui se traduit en français par "Erreur Moyenne Absolue". Il s'agit d'une métrique utilisée pour évaluer l'exactitude d'un modèle prédictif ou d'un algorithme. Le MAE mesure la magnitude moyenne des erreurs entre les valeurs prédites et les valeurs réelles dans un ensemble de données.

Pour calculer le MAE, vous suivez généralement ces étapes :

1. Rassemblez un ensemble de données avec les valeurs réelles et les valeurs prédites correspondantes.
2. Pour chaque point de données, trouvez la différence absolue entre la valeur prédite et la valeur réelle.
3. Additionnez toutes les différences absolues.
4. Divisez la somme totale par le nombre de points de données pour calculer la moyenne.

La formule du MAE est la suivante :

$$\text{MAE} = (1/n) * \sum |\text{réel} - \text{prédit}|$$

Où :

- MAE est l'Erreur Moyenne Absolue.
- n est le nombre de points de données dans l'ensemble de données.
- $\Sigma$  représente la somme des différences absolues pour tous les points de données.

Le MAE fournit une mesure de la proximité moyenne des valeurs prédites par rapport aux valeurs réelles. Il est utile pour comparer différents modèles ou algorithmes et déterminer lequel présente une meilleure précision en termes d'exactitude.

### III.8.5.2 MSE

MSE signifie Mean Squared Error, ce qui se traduit en français par "Erreur quadratique moyenne". Il s'agit d'une autre métrique couramment utilisée pour évaluer la performance des modèles prédictifs ou des algorithmes.

Pour calculer le MSE, vous suivez généralement ces étapes :

1. Rassemblez un ensemble de données avec les valeurs réelles et les valeurs prédites correspondantes.
2. Pour chaque point de données, calculez la différence au carré entre la valeur prédite et la valeur réelle.
3. Faites la somme de toutes les différences au carré.
4. Divisez la somme totale par le nombre de points de données pour calculer la moyenne.

La formule du MSE est la suivante :

$$\text{MSE} = (1/n) * \Sigma (\text{réel} - \text{prédit})^2$$

Où :

- MSE représente l'Erreur quadratique moyenne.
- n est le nombre de points de données dans l'ensemble de données.
- $\Sigma$  représente la somme des différences au carré pour tous les points de données.

Le MSE fournit une mesure de la différence quadratique moyenne entre les valeurs prédites et les valeurs réelles. Contrairement au MAE, le MSE accorde plus de poids aux erreurs plus importantes. Il est couramment utilisé dans les problèmes de régression pour évaluer la performance globale d'un modèle et comparer différents modèles ou algorithmes.

### III.8.5.3 RMSE

RMSE Est l'écart type des résidus (erreurs de prédiction). Les résidus sont une mesure de la distance par rapport aux points de données de la ligne de régression ; Le RMSE est une mesure de la répartition de ces résidus. En d'autres termes, il vous indique à quel point les données sont concentrées autour de la ligne de meilleur ajustement. L'erreur quadratique moyenne est couramment utilisée en climatologie, en prévision et en analyse de régression pour vérifier les résultats expérimentaux. [44]

### Conclusion

Ce chapitre a abordé divers aspects de la prévision météorologique pour la gestion des systèmes photovoltaïques (PV). Nous avons examiné l'importance de la prévision météorologique pour optimiser l'efficacité des systèmes PV, en mettant l'accent sur l'impact de l'irradiance solaire et de la température. Nous avons exploré les différentes méthodes de prévision météorologique à court terme, moyen terme et long terme. Les prévisions météorologiques offrent des avantages tels que l'optimisation de la production d'énergie, la planification des opérations et la réduction des risques. Cependant, elles font face à des défis tels que la fiabilité des données, la complexité des interactions climatiques et l'incertitude des modèles. L'utilisation du langage de programmation Python et de ses bibliothèques spécialisées, telles que Pandas, Numpy, Matplotlib, Scikit-learn, Keras et TensorFlow, facilite le développement de modèles de prévision météorologique. L'environnement de développement Anaconda et l'utilisation de Jupyter Notebook sont des outils pratiques pour analyser les données météorologiques. En utilisant des prévisions météorologiques précises, il est possible d'améliorer la gestion des systèmes PV et de maximiser leur rendement énergétique.

# **Chapitre IV**

## **Implémentation des résultats**

## IV.1. Choix des méthodes de machines learning

### IV.1.1. La régression

La régression est une méthode couramment utilisée en apprentissage automatique (machine learning) pour résoudre des problèmes de prédiction ou d'estimation de valeurs numériques continues. Elle permet de modéliser la relation entre une variable cible (la variable que l'on souhaite prédire) et un ensemble de variables prédictives (les variables d'entrée).

### IV.1.2. Pourquoi la régression

- 1. Prédiction de valeurs continues :** La régression est utilisée lorsque l'on souhaite prédire des valeurs numériques continues. Par exemple, prédire le prix d'une maison en fonction de ses caractéristiques telles que la superficie, le nombre de chambres, etc.
- 2. Relation entre variables :** La régression permet de modéliser la relation entre les variables d'entrée et la variable cible. Elle permet de comprendre comment les variables prédictives influencent la variable cible et d'identifier les facteurs les plus importants.

### IV.1.3. Les méthodes choisies

#### 1. L'arbre de décision (décision tree)

Un arbre de décision (decision tree en anglais) est un modèle d'apprentissage automatique supervisé qui est utilisé pour la classification et la régression. Il représente une hiérarchie de décisions et de conditions qui permettent de prendre des décisions ou de prédire des valeurs en fonction des caractéristiques (features) d'un ensemble de données.

#### 2. La forêt aléatoire (Random Forest)

La forêt aléatoire (Random Forest en anglais) est un modèle d'apprentissage automatique qui combine plusieurs arbres de décision pour effectuer des prédictions plus robustes et précises.

Le principe de base des forêts aléatoires est de créer un ensemble d'arbres de décision, où chaque arbre est entraîné sur un sous-ensemble aléatoire des données d'entraînement, ainsi que sur un sous-ensemble aléatoire des caractéristiques (features) disponibles. Ces deux mécanismes d'aléatoire contribuent à réduire la corrélation entre les arbres individuels et à améliorer la généralisation du modèle.

## IV.2. Choix des méthodes de Deep Learning

### IV.2.1. La méthode des réseaux de neurones récurrents (RNN)

La méthode des réseaux de neurones récurrents (RNN) est couramment utilisée en apprentissage profond (Deep Learning) en raison de ses capacités à traiter des données séquentielles ou temporelles.

### IV.2.2. Pourquoi la méthode de réseaux de neurones récurrents

- 1. Modélisation de la dépendance temporelle :** Les RNN sont spécialement conçus pour traiter des données séquentielles dans lesquelles l'ordre des éléments importe. Ils peuvent capturer les dépendances temporelles entre les éléments d'une séquence, ce qui les rend adaptés à des tâches telles que la prédiction de séries temporelles, la traduction automatique ou la génération de texte.
- 2. Longue mémoire contextuelle :** Contrairement aux réseaux de neurones traditionnels, les RNN ont une mémoire interne qui leur permet de conserver des informations sur les étapes précédentes de la séquence. Cela leur permet de prendre en compte le contexte à long terme lors de la prédiction ou de la classification. Par conséquent, les RNN sont efficaces pour modéliser des relations complexes à travers le temps.
- 3. Taille variable des entrées/sorties :** Les RNN peuvent traiter des séquences de longueurs variables en ajustant automatiquement leur état interne en fonction de la longueur de la séquence d'entrée. Cela les rend très utiles pour des tâches telles que la classification de texte, où les phrases peuvent avoir des longueurs différentes.

### IV.2.3. Les méthodes choisies

#### Mémoire à court terme (LSTM)

LSTM (Long Short-Term Memory) est un type de réseau de neurones récurrents (RNN) qui a été conçu pour capturer et exploiter les dépendances à long terme dans les données séquentielles, comme les séquences de texte, de sons ou de données temporelles.

La principale caractéristique des LSTM réside dans leur capacité à conserver et à mettre à jour des informations sur des périodes de temps prolongées, ce qui les distingue des réseaux de neurones récurrents traditionnels qui ont du mal à gérer les dépendances à long terme. Cela est rendu possible grâce à l'utilisation de cellules mémoire spéciales et de portes de contrôle.

### ✚ Les gated recurrent units (GRU)

GRU sont une variante de RNN qui ont été développées pour surmonter certains problèmes rencontrés par les réseaux de neurones récurrents classiques, tels que le problème du gradient qui disparaît ou explose.

## IV.3. Méthodologie expérimentale

### IV.3.1. L'arbre de décision

#### ➤ La prédiction de température

##### 1. Sélection des données pour la modélisation

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn import linear_model
weather_data = pd.read_csv("June_data.csv")
weather_data
```

	Day	Time	W/m^2	Irradiance	h	°C	Temperature	m/s
0	1	00:00	0	0.00	51069.52	126.93	28.93	0.55
1	1	00:05	0	0.00	51069.60	126.93	28.90	0.19
2	1	00:10	0	0.00	51069.68	126.93	28.75	0.21
3	1	00:15	0	0.00	51069.77	126.93	28.73	0.00
4	1	00:20	0	0.31	51069.85	126.93	28.68	0.00
...	...	...	...	...	...	...	...	...
8058	28	23:35	0	0.00	51735.59	126.93	27.09	0.00
8059	28	23:40	0	0.06	51735.67	126.93	27.12	0.00
8060	28	23:45	0	0.00	51735.76	126.93	27.20	0.00
8061	28	23:50	0	0.00	51735.84	126.93	27.05	0.00
8062	28	23:55	0	0.00	51735.93	126.93	27.08	0.00

8063 rows × 8 columns

## 2. Exploration et préparation des données

```
weather_data.isnull()
```

	Day	Irradiance	Temperature	hour	minute
0	False	False	False	False	False
1	False	False	False	False	False
2	False	False	False	False	False
3	False	False	False	False	False
4	False	False	False	False	False
...	...	...	...	...	...
7770	False	False	False	False	False
7771	False	False	False	False	False
7772	False	False	False	False	False
7773	False	False	False	False	False
7774	False	False	False	False	False

7775 rows x 5 columns

```
weather_data.apply(pd.isnull).sum()
```

```
Day          0
Time         0
W/m^2       0
Irradiance  0
h           0
°C          0
Temperature  0
m/s        288
dtype: int64
```

## 3. Sélectionner le 'Target' et choix de 'features'

```
weather_features = ["Day", "hour", "minute"]
X = weather_data[weather_features]
y = weather_data.Temperature
```

## 4. Séparation des ensembles de données

```
from sklearn.model_selection import train_test_split
train, test = train_test_split(weather_data, test_size=0.2, shuffle=False)
```

## 5. Spécifier et adapter le modèle

```
from sklearn.tree import DecisionTreeRegressor
D = DecisionTreeRegressor(random_state=1)
```

## 6. Evaluation de modèle

```
predictions= D.predict(test[predictors])
predictions
```

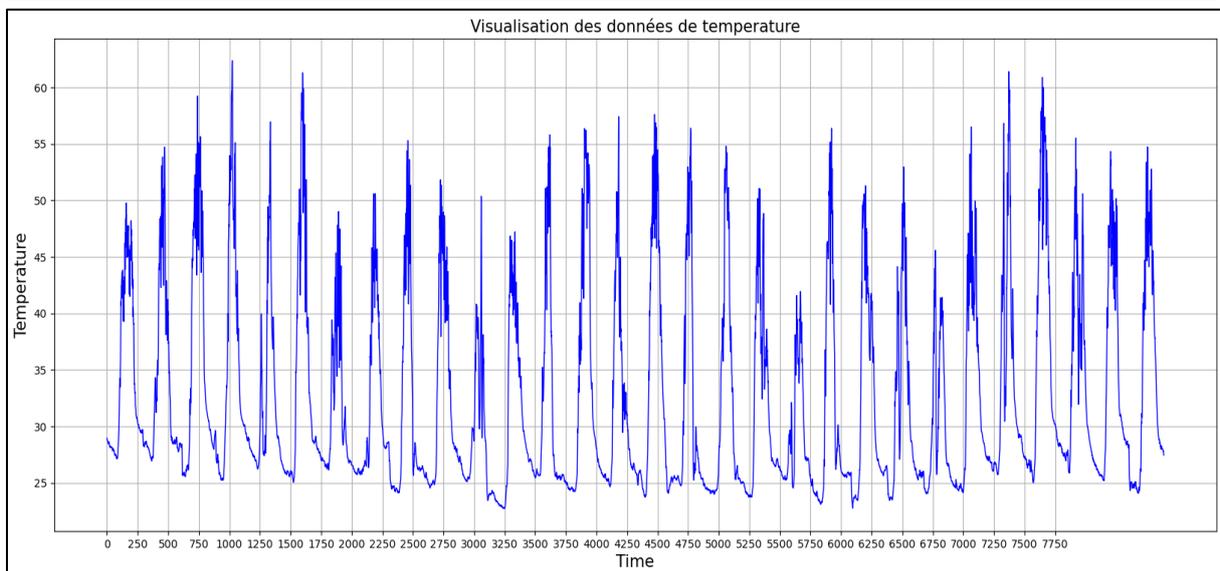
## 7. Combiner les valeurs actuelles avec les valeurs prédites

```
combined= pd.concat([test["Temperature"], pd.Series(predictions, index=test.index)], axis=1)
combined.columns= ["actual","predictions"]
combined
```

## 8. Tracer les graphes

### ➤ Tracer les valeurs actuelles de température

```
plt.figure(figsize=(20,8))
plt.xticks(np.arange(0, 8000, step=250))
plt.grid('true')
plt.plot(weather_data["Temperature"],color='blue', linewidth=1)
plt.xlabel('Time',fontsize=15)
plt.ylabel('Temperature',fontsize=15)
plt.title("Visualisation des données de température",fontsize=15)
```

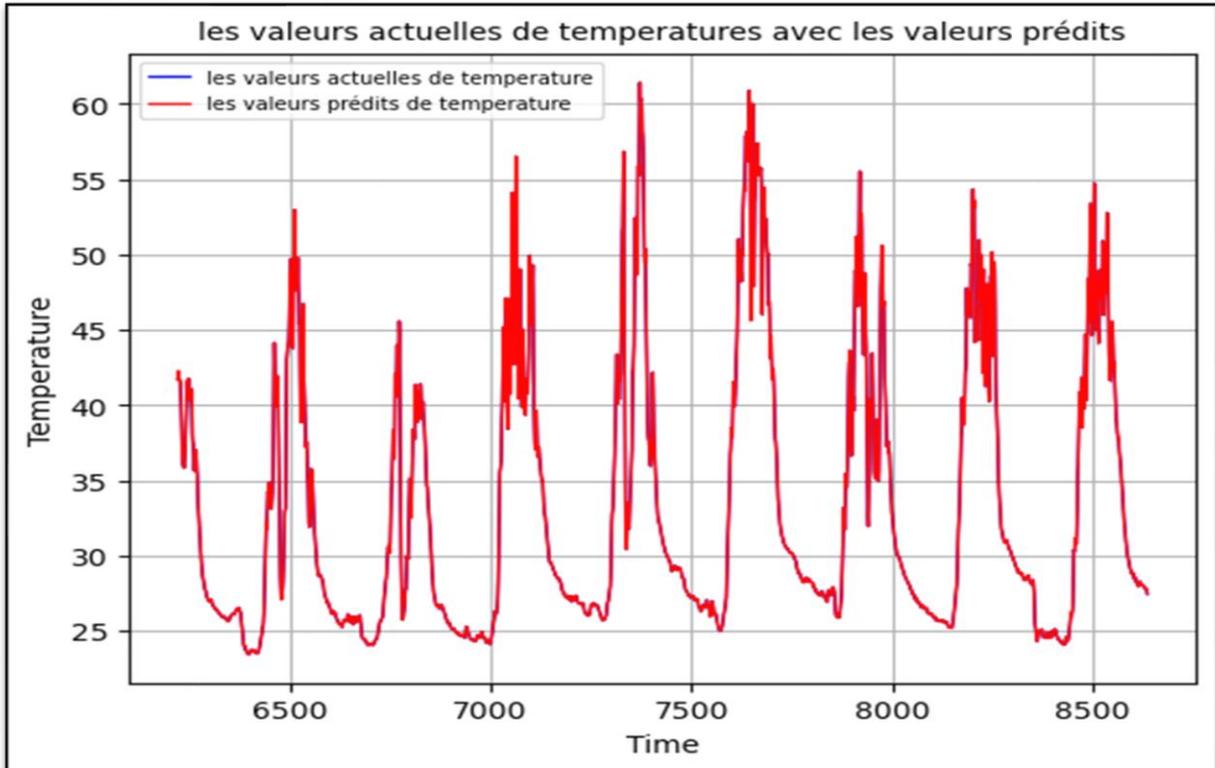


**Figure IV.1:** Visualisation des données de température

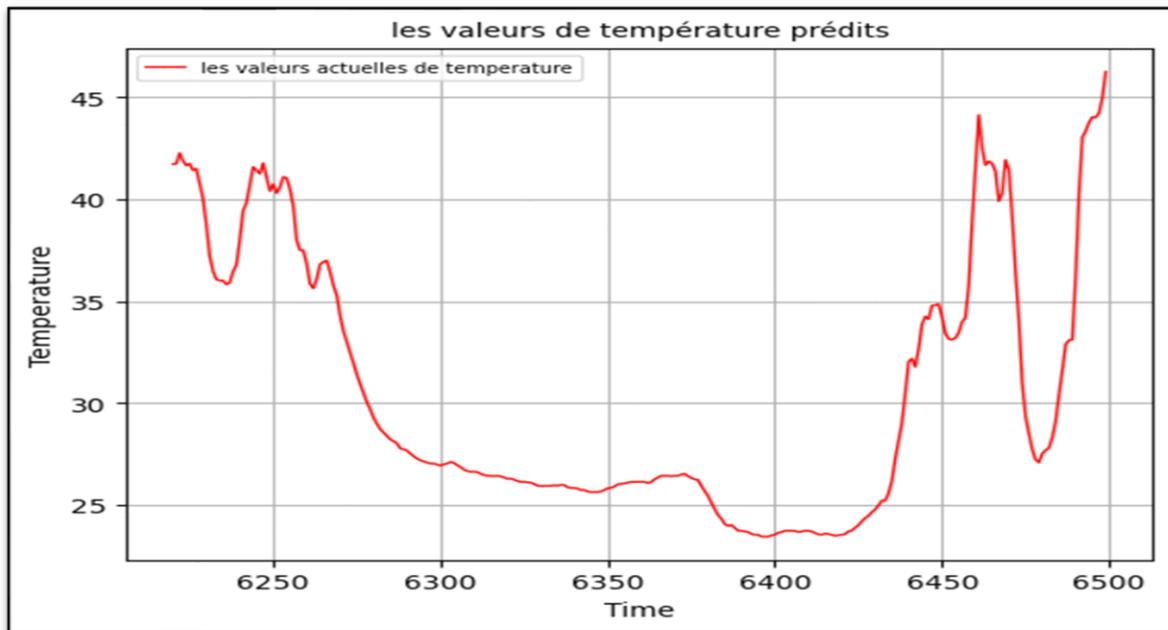
Ce graphe représente la visualisation de température en fonction des nombres des données les points ou il y a des pics sont les points qui reflet sur la mi-journée

➤ Les valeurs actuelles de test de température avec les valeurs prédites

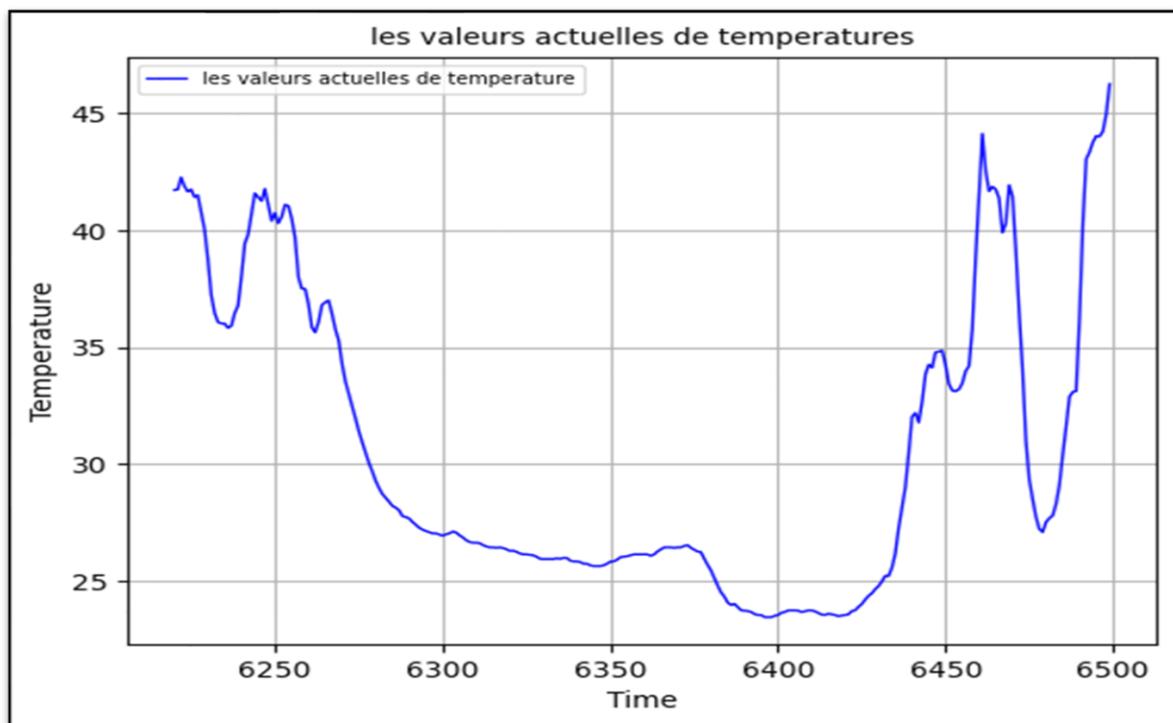
```
combined["actual"].plot(color='blue',linewidth=1)  
combined["predictions"].plot(color='red',linewidth=1)
```



**Figure IV.2:** les valeurs actuelles de température avec les valeurs prédites



**Figure IV.3:** Les valeurs de température prédit par l'arbre de décision

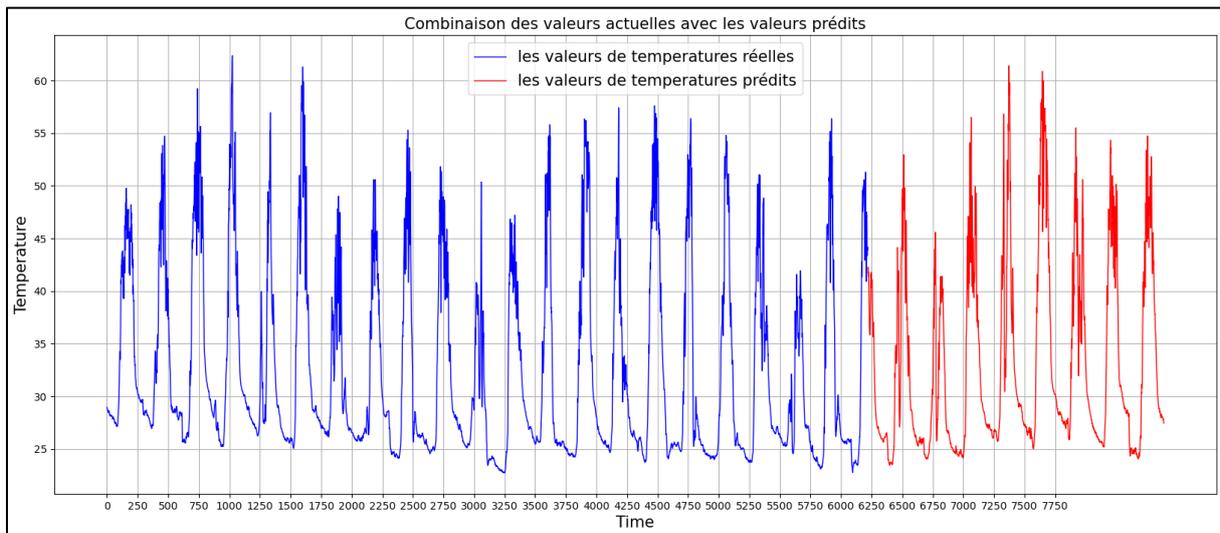


**Figure IV.4:** Les valeurs actuelles de température

D'après c'est deux graphes, on peut constater qu'il existe une conformité entre les valeurs actuelles et les valeurs prédites par les modèles de l'arbre de décision.

➤ Les valeurs actuelles de température avec les valeurs prédites de test

```
x1=weather_data["Temperature"][:6219]
x3=combined["predictions"]
plt.figure(figsize=(20,8))
plt.grid('true')
plt.xticks(np.arange(0, 8000, step=250))
plt.plot(x1,color='blue',linewidth=1)
plt.plot(x3,color='red',linewidth=1)
plt.xlabel('Time',fontsize='15')
plt.ylabel('Temperature',fontsize='15')
plt.legend(['les valeurs de temperatures réelles','les valeurs de temperatures prédits'],fontsize='15')
plt.title("Combinaison des valeurs actuelles avec les valeurs prédits",fontsize=15)
plt.show()
```



**Figure IV.5:** Combinaison des valeurs actuelles avec les valeurs prédites par DT

La combinaison des valeurs actuelles avec les valeurs prédites de température nous permet d'assurer les performances de ce modèle

## 9. Métriques d'évaluation pour la comparaison

### ➤ Erreur absolue Moyenne

```
from sklearn.metrics import mean_absolute_error
mean_absolute_error(test["Temperature"], predictions)

1.59929233129283e-17
```

### ➤ Erreur quadratique Moyenne

```
from sklearn.metrics import mean_squared_error
mean_squared_error(test["Temperature"], predictions)

5.681827741784779e-32
```

### ➤ Racine carré de l'erreur quadratique moyenne

```
import math
MSE=mean_squared_error(test["Temperature"], predictions)
RMSE=math.sqrt(MSE)
print(RMSE)

2.3836584784286486e-16
```

## 10. La prédiction des nouvelles valeurs

Pour cela on va déclarer une matrice qui faire la prédiction d'une date donnée en utilisant les « features » (voir annexe A)

## ➤ La prédiction de l'irradiance

### 1. La déférence est dans le choix de cible « Target »

```
I=weather_data.Irradiance
weather_features =["Day","hour","minute"]
X = weather_data[weather_features]
```

### 2. Spécifier et adapter le modèle

```
D.fit(X,I)
```

```
DecisionTreeRegressor
DecisionTreeRegressor(random_state=1)
```

### 3. Evaluation de modèle

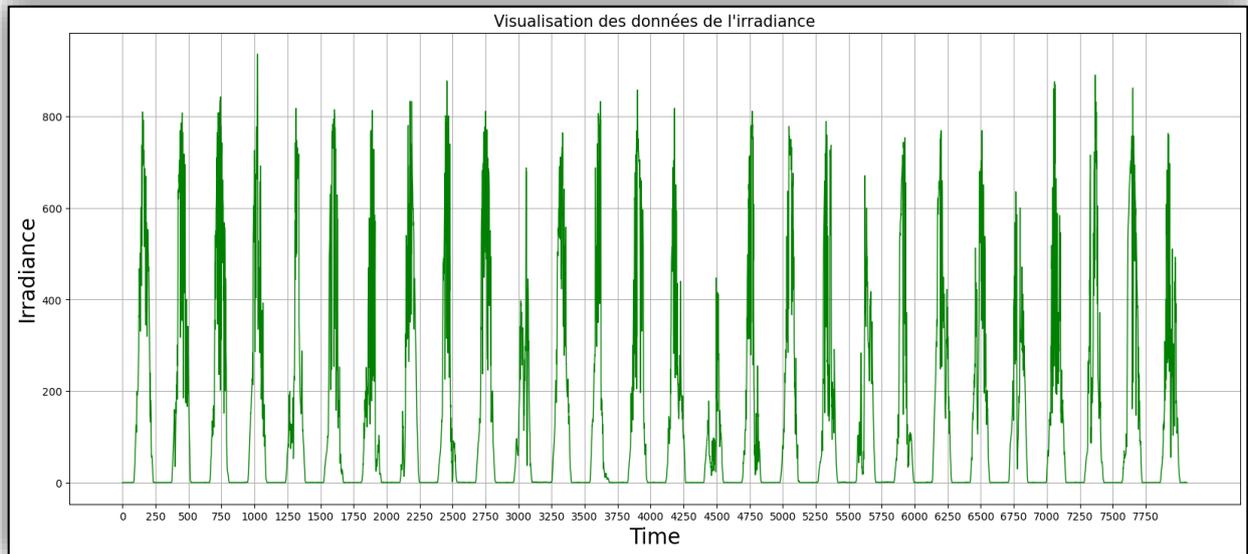
```
predictions= D.predict(test[predictors])
predictions
array([[340.16, 371.68, 412.42, ..., 0. , 0. , 0. ]])
```

### 4. Combiner les valeurs actuelles avec les valeurs prédites

	actual	predictions
<b>6220</b>	340.16	340.16
<b>6221</b>	371.68	371.68
<b>6222</b>	412.42	412.42
<b>6223</b>	369.29	369.29
<b>6224</b>	346.45	346.45
...	...	...
<b>7770</b>	0.00	0.00
<b>7771</b>	0.00	0.00
<b>7772</b>	0.00	0.00
<b>7773</b>	0.00	0.00
<b>7774</b>	0.00	0.00

## 5. Tracer les graphes

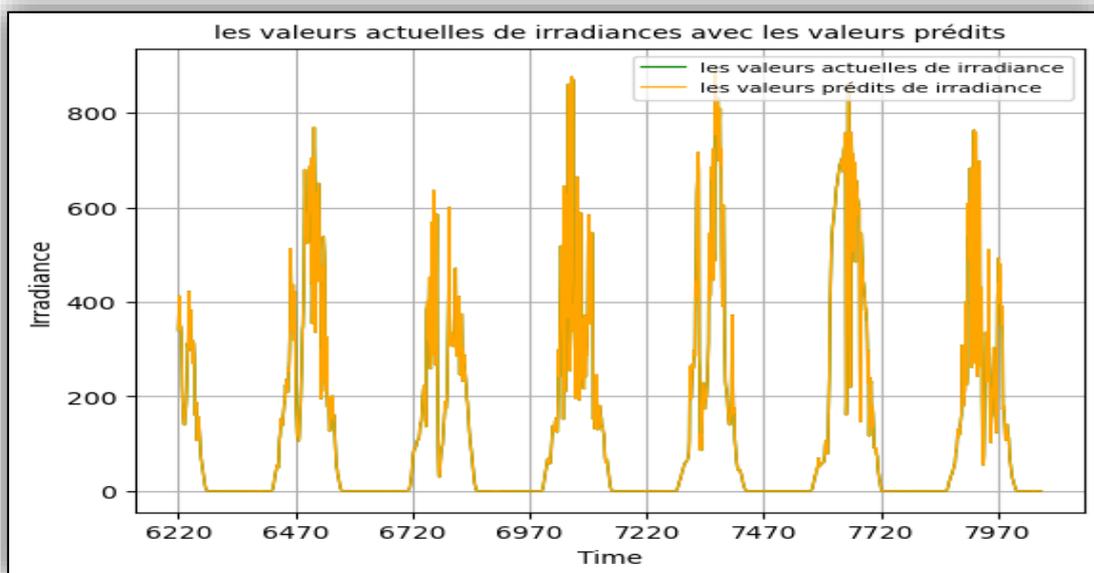
### ➤ Tracer les valeurs actuelles de l'irradiance



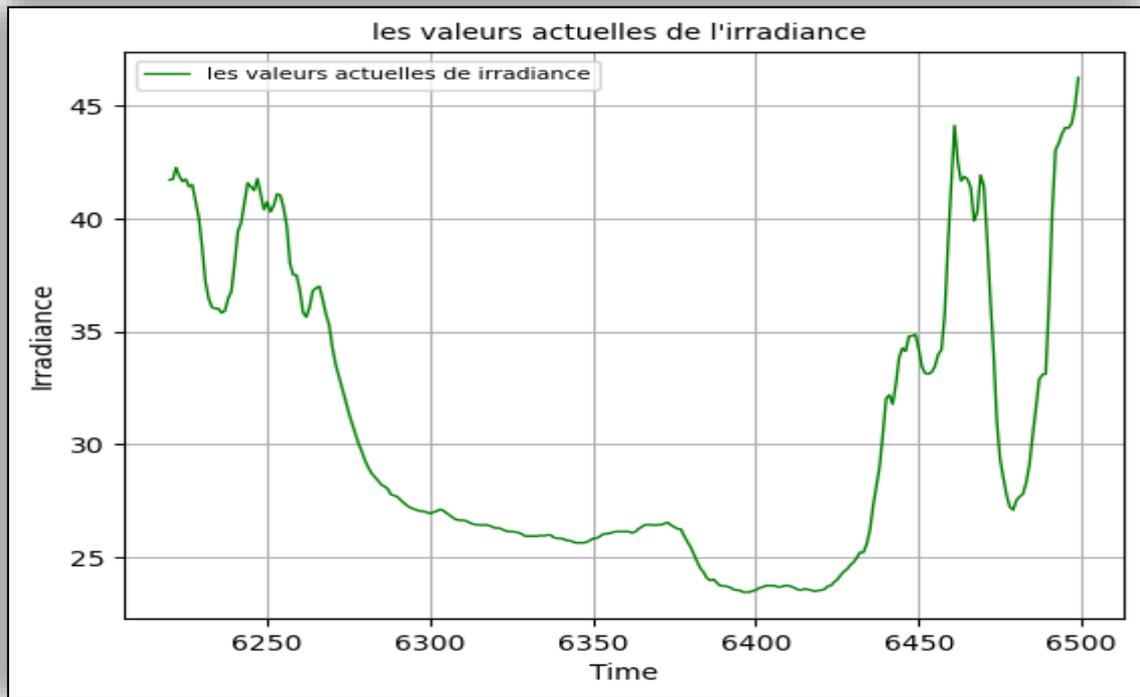
**Figure IV.6 :** Visualisation des données de l'irradiance

Ce graphique représente la visualisation de l'irradiance en fonction des nombres des données les points ou il y a des pics sont les points qui reflet sur la mi-journée

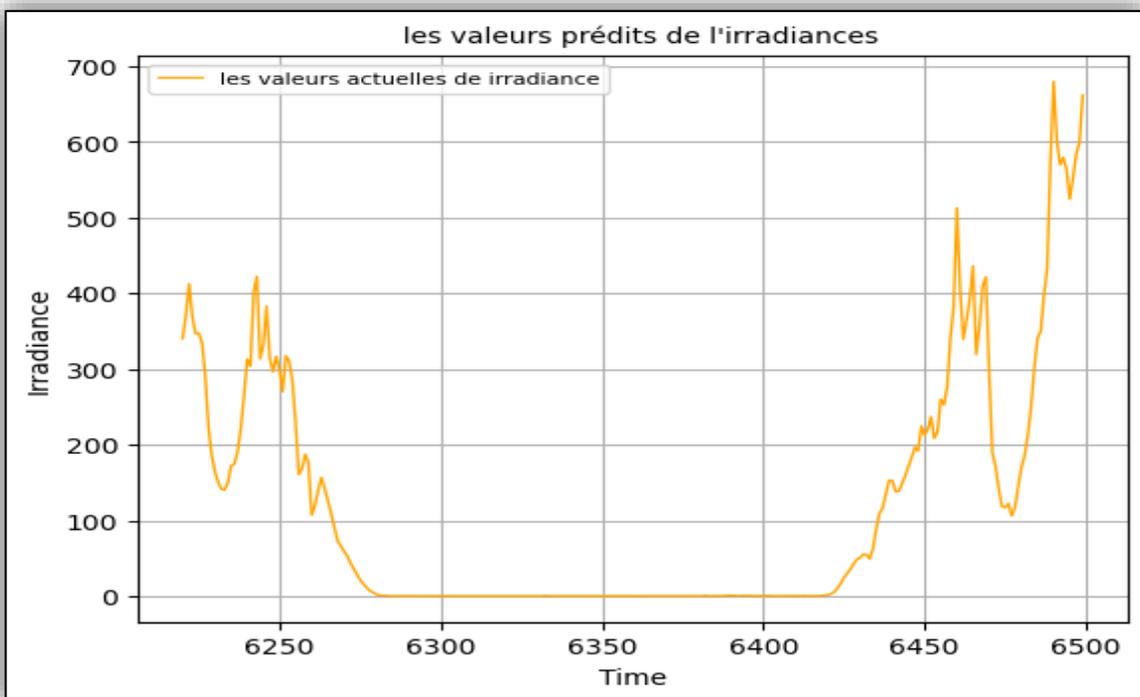
### ➤ Les valeurs actuelles de test de l'irradiance avec les valeurs prédites



**Figure IV.7 :** les valeurs actuelles de l'irradiance avec les valeurs prédites par DT



**Figure IV.8 :** Les valeurs actuelles de l'irradiance



**Figure IV.9:** les valeurs prédites de l'irradiance

Dans ces deux graphes on prend un intervalle des valeurs actuelle et le même intervalle pour les valeurs prédites, et on remarque qu'il existe une conformité entre les deux types de valeurs

#### 6. Les erreurs

MAE	MSE	RMSE
0.18	4.17	2.04

**Tableau IV.1** : Les résultats des métriques pour l'irradiance (Arbre de décision).

### IV.3.2. Le forêt aléatoire

Le point déférent entre la méthode de l'arbre de décision et le forêt aléatoire est la déclaration de modèle :

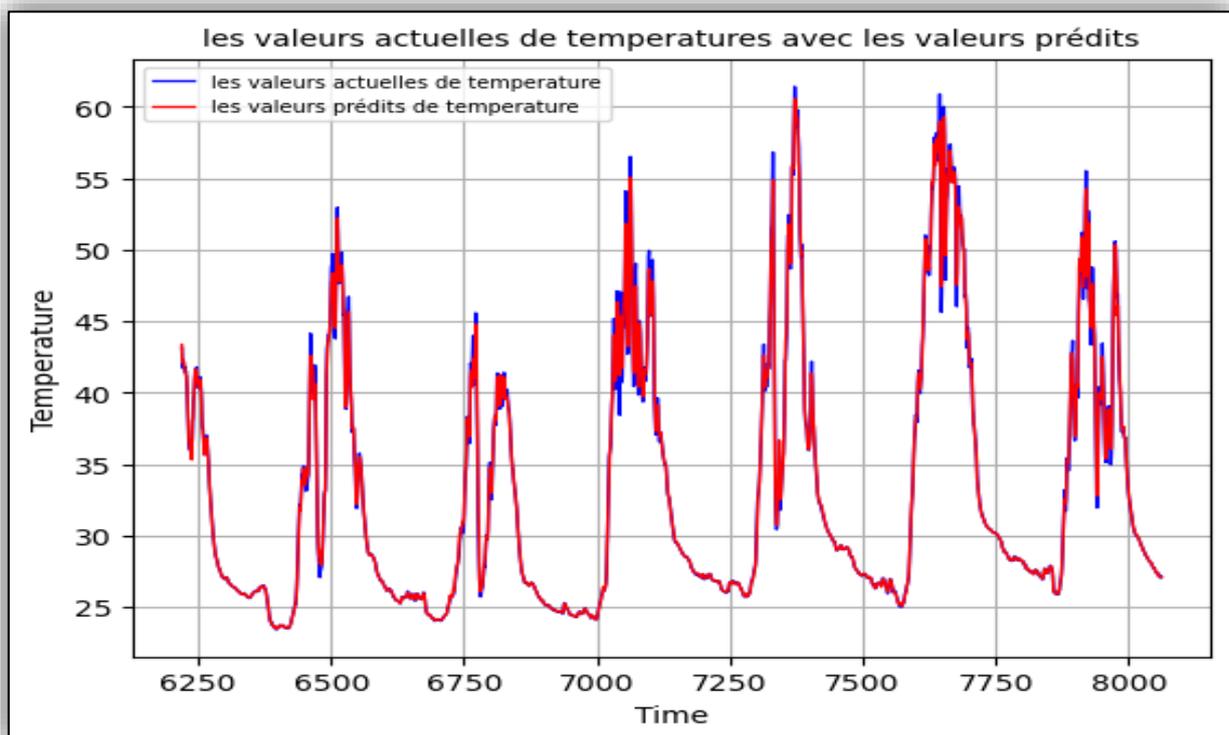
#### ✚ Pour la prédiction de température

##### 1. Spécifier et adapter le modèle

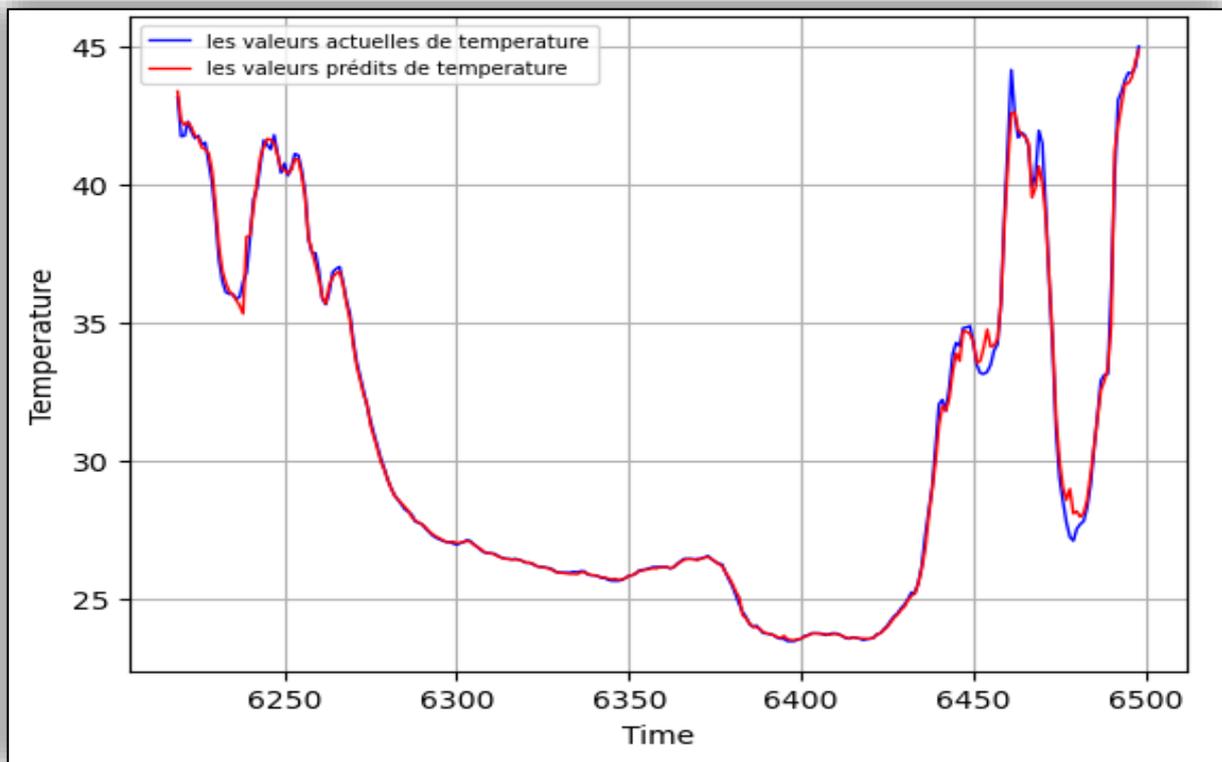
```
from sklearn.ensemble import RandomForestRegressor
forest_model = RandomForestRegressor(random_state=1)
predictors=weather_features
forest_model.fit(X,y)
```

RandomForestRegressor  
RandomForestRegressor(random\_state=1)

##### 2. Les graphes



**Figure IV.10:** les valeurs actuelles de température et les valeurs prédites en forêt aléatoire



**Figure IV.11 :** Un intervalle des valeurs prédites et les valeurs actuelles de la température

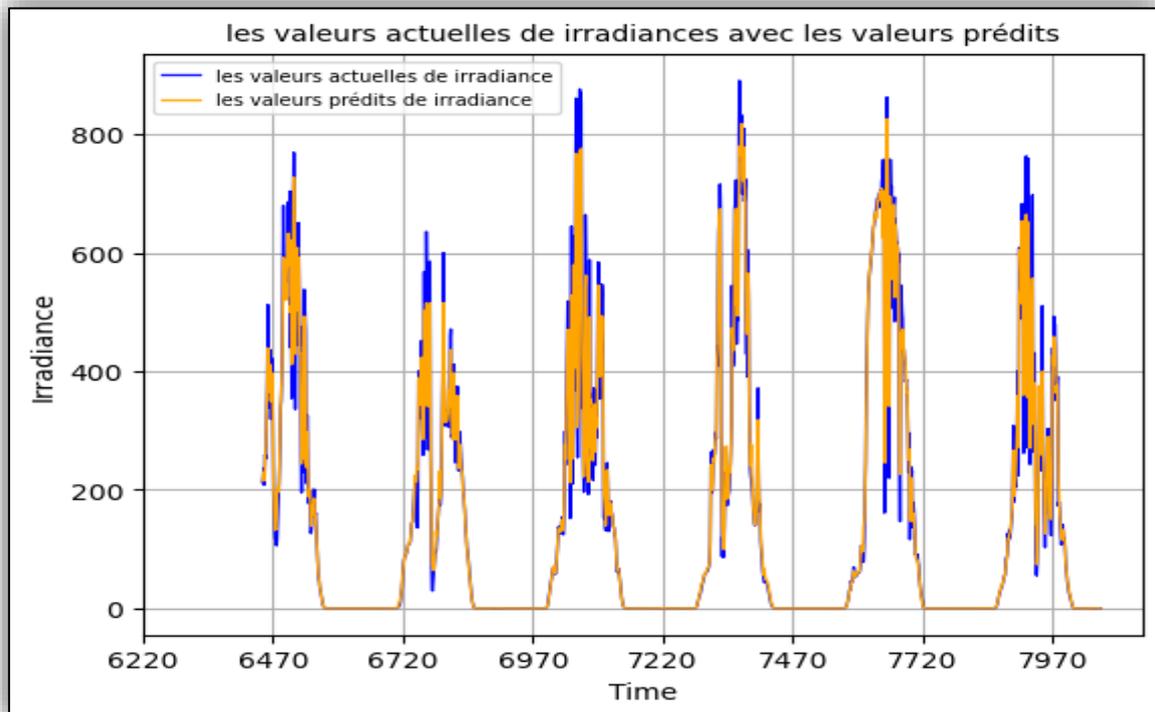
## ✚ Pour la prédiction de l'irradiance

### 1. Spécifier et adapter le modèle

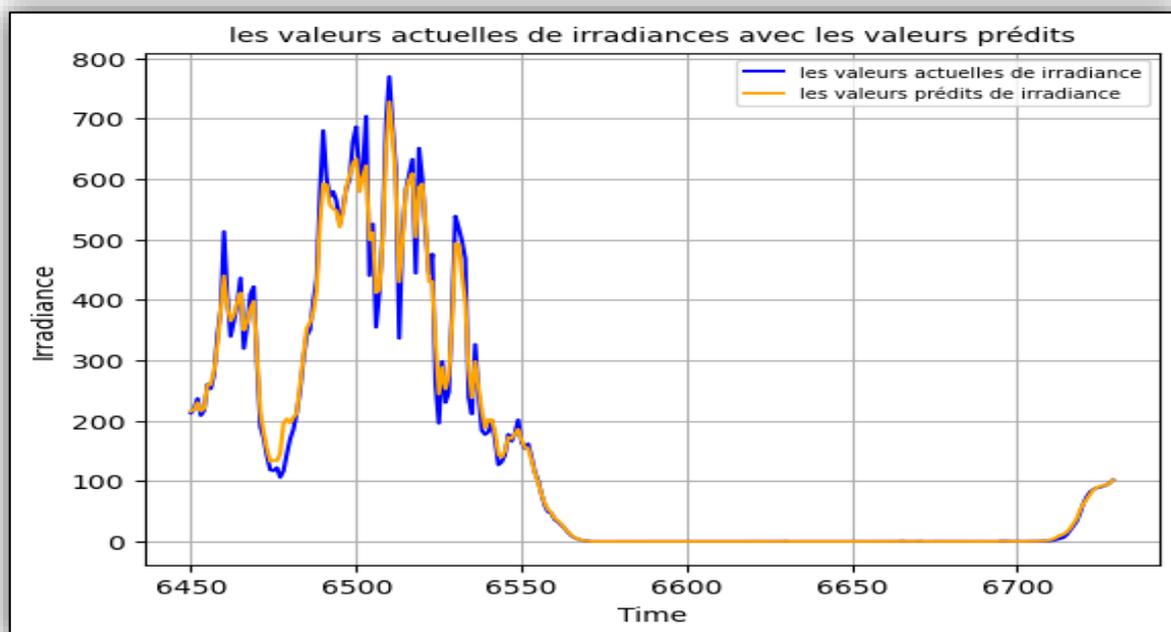
```
I=weather_data.Irradiance  
weather_features=["Day","hour","minute"]  
X = weather_data[weather_features]
```

```
D.fit(X,I)
```

```
DecisionTreeRegressor  
DecisionTreeRegressor(random_state=1)
```



**Figure IV.12 :** les valeurs actuelles de l'irradiance et les valeurs prédites en forêt aléatoire



**Figure IV.13 :** Un intervalle des valeurs prédites et les valeurs actuelles de l'irradiance

## 2. Les erreurs

	<b>Température</b>	<b>L'irradiance</b>
<b>Erreur absolue Moyenne</b>	0.20937987138263517	10.357283199008055
<b>Erreur quadratique Moyenne</b>	0.18278701316398638	554.7308538003596
<b>Racine carré de l'erreur quadratique moyenne</b>	0.4275359787947517	23.55272497611178

**Tableau IV.2 :** Les résultats des métriques pour la température et l'irradiance

(Forêts aléatoires).

Pour le modèle des forêt aléatoires on constate qu'il existe des écarts entre les valeurs prédites et les valeurs actuelles dans quelques points et au contraire au modèle de l'arbre de décision, ce modèle a des erreurs plus élevées.

### IV.3.3. LSTM

#### ✚ Pour la prédiction de température

##### 1. Sélection des données pour la modélisation

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from tensorflow import keras
from tensorflow.keras import layers
weather_data = pd.read_csv("June_data.csv")
weather_data
```

##### 2. Séparation des ensembles de données

```
from sklearn.model_selection import train_test_split
train, test = train_test_split(weather_data, test_size=0.2, shuffle=False)
```

##### 3. Sélectionner le ‘Target’ et choix de ‘features’

```
weather_data = weather_data.drop(columns=["Day", "Time", "Irradiance", "W/m^2", "h", "°C", "m/s"], axis=1)
weather_data
```

Temperature	
0	28.93
1	28.90
2	28.75
3	28.73
4	28.68
...	...
8058	27.09
8059	27.12
8060	27.20
8061	27.05
8062	27.08

#### 4. Normalisation des données

```
from sklearn.preprocessing import MinMaxScaler
scaler=MinMaxScaler()
training_data=scaler.fit_transform(train)
training_data

array([[0.15662043],
       [0.15586381],
       [0.15208071],
       ...,
       [0.54148802],
       [0.53467844],
       [0.51525851]])
```

#### 5. Réalisation de modèle

```
from tensorflow import keras
from keras.models import Sequential
from keras.layers import Dense,Dropout
from keras.layers import LSTM

model=keras.Sequential()
model.add(LSTM(128, activation='relu', input_shape=(X_train.shape[1],1)))
model.add(Dense(1))
```

```
model.compile(
    optimizer="adam",
    loss="mae",
)
```

#### 6. Adaptation de modèle

```
model.fit(X_train,y_train,epochs=50,batch_size=50)
```

#### 7. Evaluation de modèle

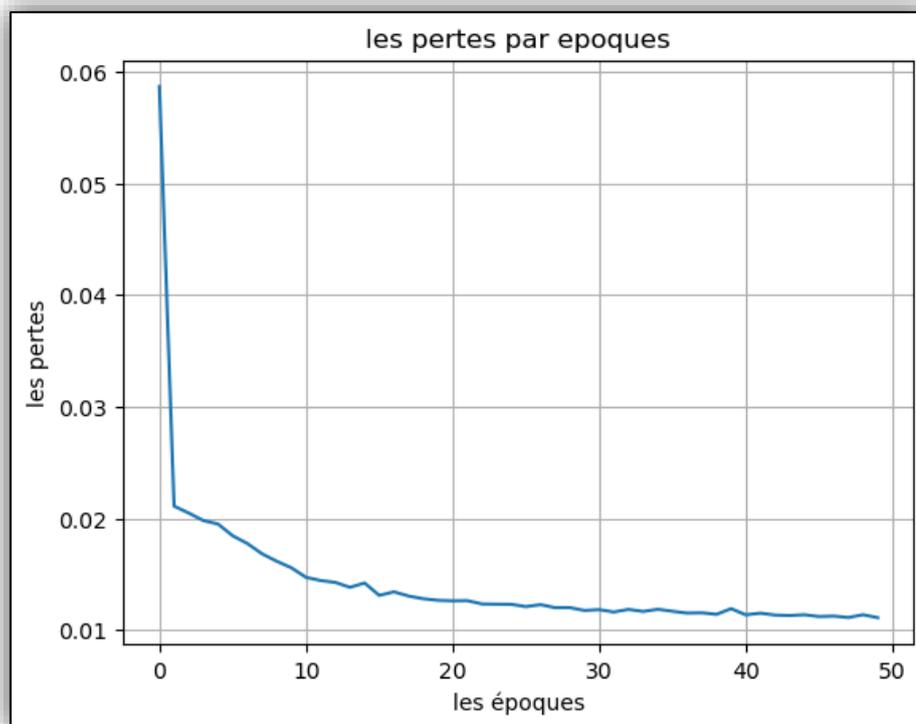
```
y_predictions=model.predict(X_test)

49/49 [=====] - 0s 2ms/step
```

## 8. Tracer les graphes

### ➤ Tracer le graphe de loss

```
loss_per_epoch = model.history.history['loss']  
plt.plot(range(len(loss_per_epoch)),loss_per_epoch)
```



**Figure IV.14:** Les pertes par époques pour la prédiction de température en utilisant LSTM

➤ Les valeurs actuelles de test de température avec les valeurs prédites

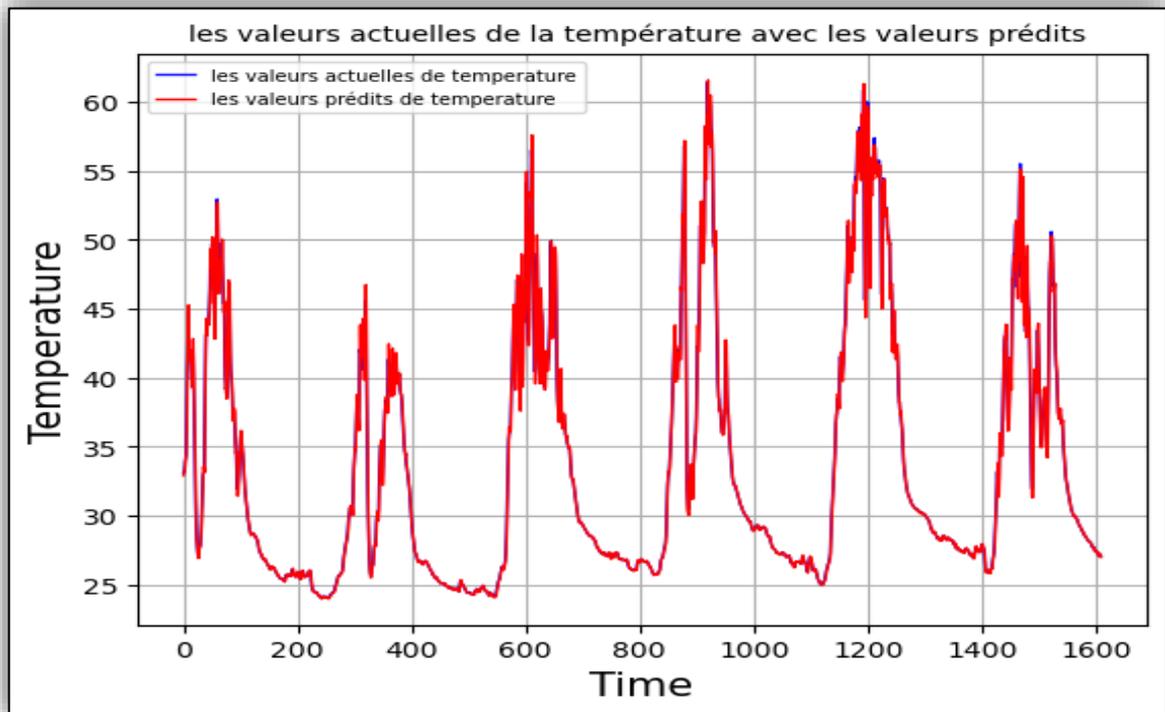


Figure IV.15 : les valeurs actuelles de température et les valeurs prédites par LSTM

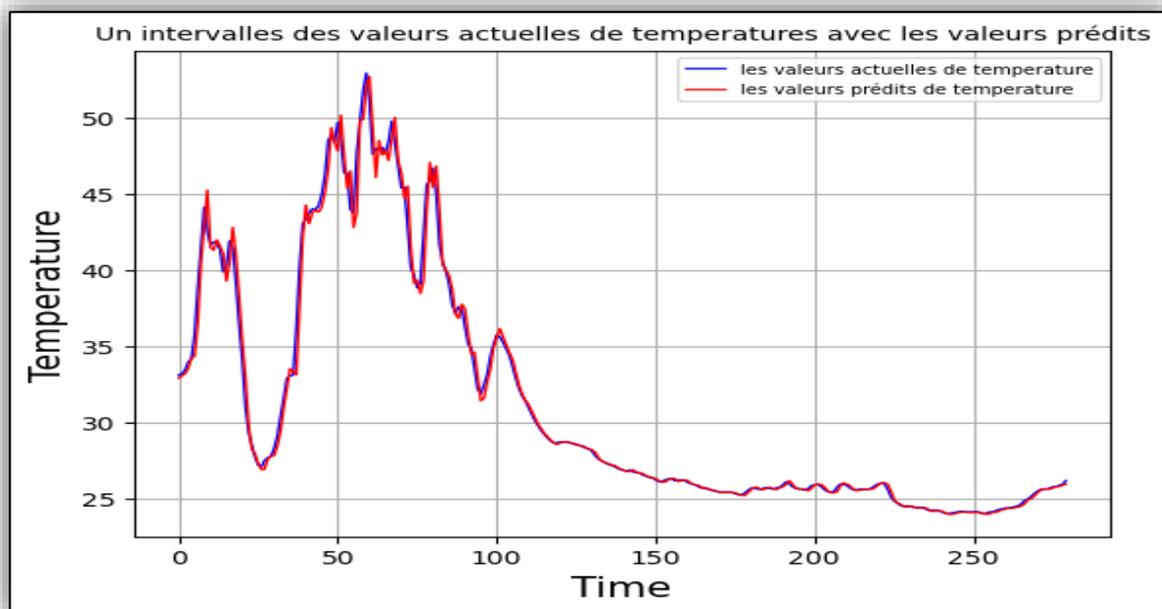


Figure IV.16 : Un intervalle des valeurs actuelles de la température et des valeurs prédites par LSTM

## ✚ Pour la prédiction de l'irradiance

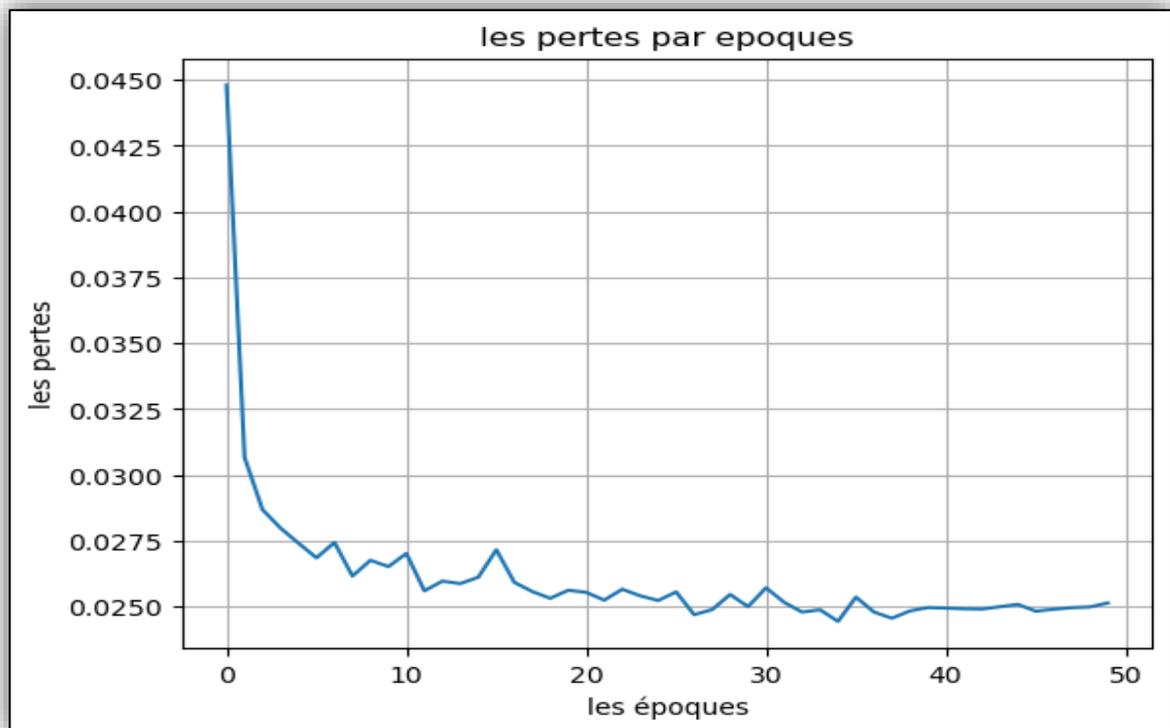
### 1. Choix de « Target »

```
weather_data=weather_data.drop(columns=["Day","Time","Temperature","W/m^2","h","°C","m/s"], axis=1)
weather_data
```

	Irradiance
0	0.00
1	0.00
2	0.00
3	0.00
4	0.31
...	...
7770	0.00
7771	0.00
7772	0.00
7773	0.00
7774	0.00

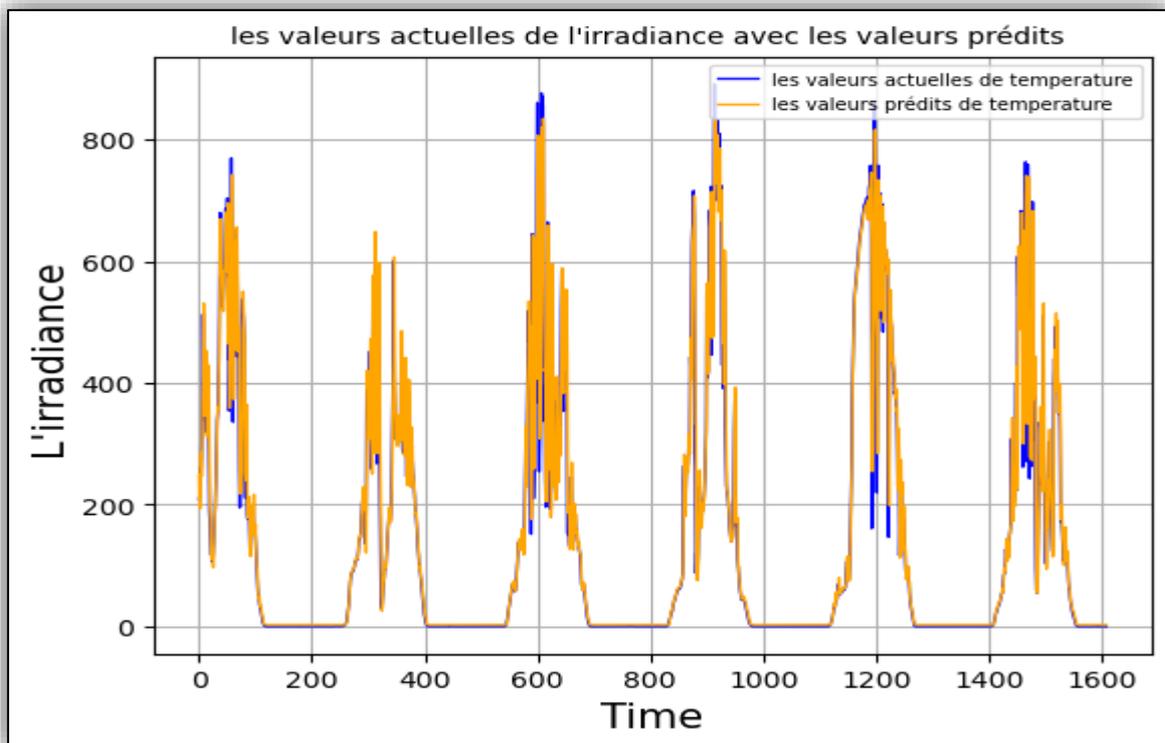
### 2. Les graphes de prédiction

#### ➤ Tracer le graphe de loss

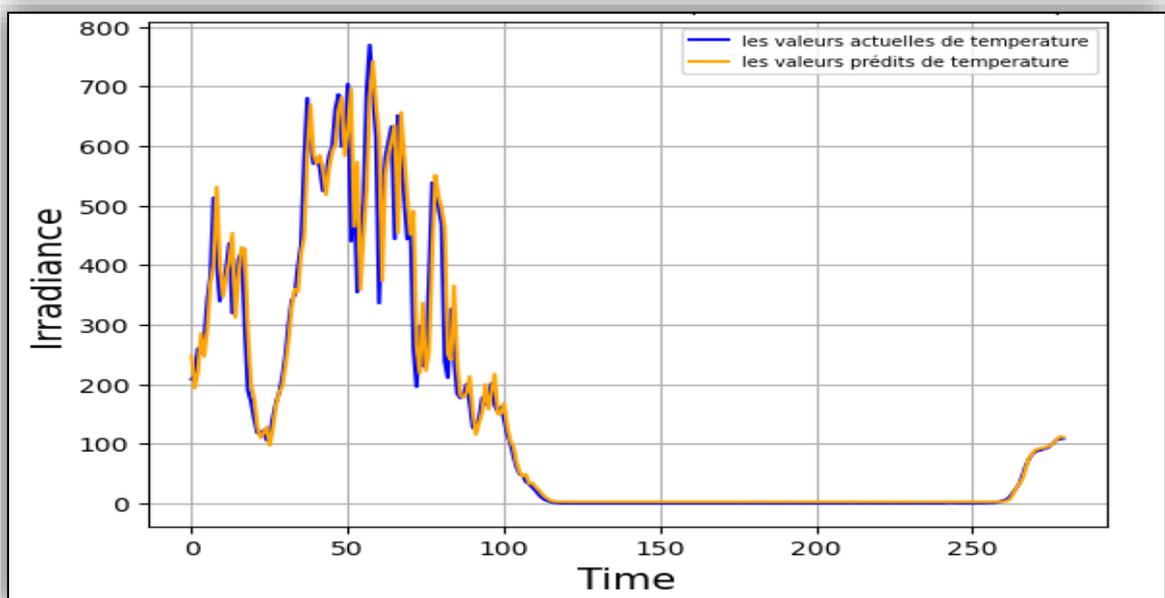


**Figure IV.17 :** Les pertes par époques pour la prédiction de L'irradiance en utilisant LSTM

➤ Les graphes des valeurs actuelles avec les valeurs prédites



**Figure IV.18 :** Les valeurs actuelles de test de l'irradiance avec les valeurs prédites par LSTM



**Figure IV.19:** Un intervalle les valeurs actuelles de test de l'irradiance avec les valeurs prédites par LSTM

### 3. Les erreurs

	<b>Température</b>	<b>L'irradiance</b>
<b>Erreur absolue Moyenne</b>	0.47522708542154457	26.805805011328708
<b>Erreur quadratique Moyenne</b>	0.8974580673457563	4017.623312481415
<b>Racine carré de l'erreur quadratique moyenne</b>	0.9473426346078573	63.38472459892379

**Tableau IV.3** : Les résultats des métriques pour la température et l'irradiance (LSTM).

D'après le graphe de fonction loss de modèle LSTM, on estime que ce modèle a des bonnes performances, et on voit ça dans les graphes qui combine entre les valeurs actuelles et les valeurs prédites.

### IV.3.4. GRU

#### 1. Sélection des données pour la modélisation

```
weather_data = pd.read_csv('June_data.csv')
```

#### 2. Sélectionner le 'Target' et choix de 'features'

```
scaler = MinMaxScaler(feature_range=(0, 1))  
weather_data['Temperature'] = scaler.fit_transform(weather_data['Temperature'].values.reshape(-1, 1))
```

#### 3. Séparation des ensembles de données

```
train_data, test_data = train_test_split(weather_data, test_size=0.2, shuffle=False)
```

```
train_X = train_data['Temperature'].values[:-1]  
train_y = train_data['Temperature'].values[1:]  
test_X = test_data['Temperature'].values[:-1]  
test_y = test_data['Temperature'].values[1:]
```

```
train_X = np.reshape(train_X, (train_X.shape[0], 1, 1))  
test_X = np.reshape(test_X, (test_X.shape[0], 1, 1))
```

#### 4. Réalisation de modèle

```
model = Sequential()  
model.add(GRU(64, input_shape=(1, 1)))  
model.add(Dense(1))  
model.compile(optimizer='adam', loss='mean_squared_error')
```

#### 5. Adaptation de modèle

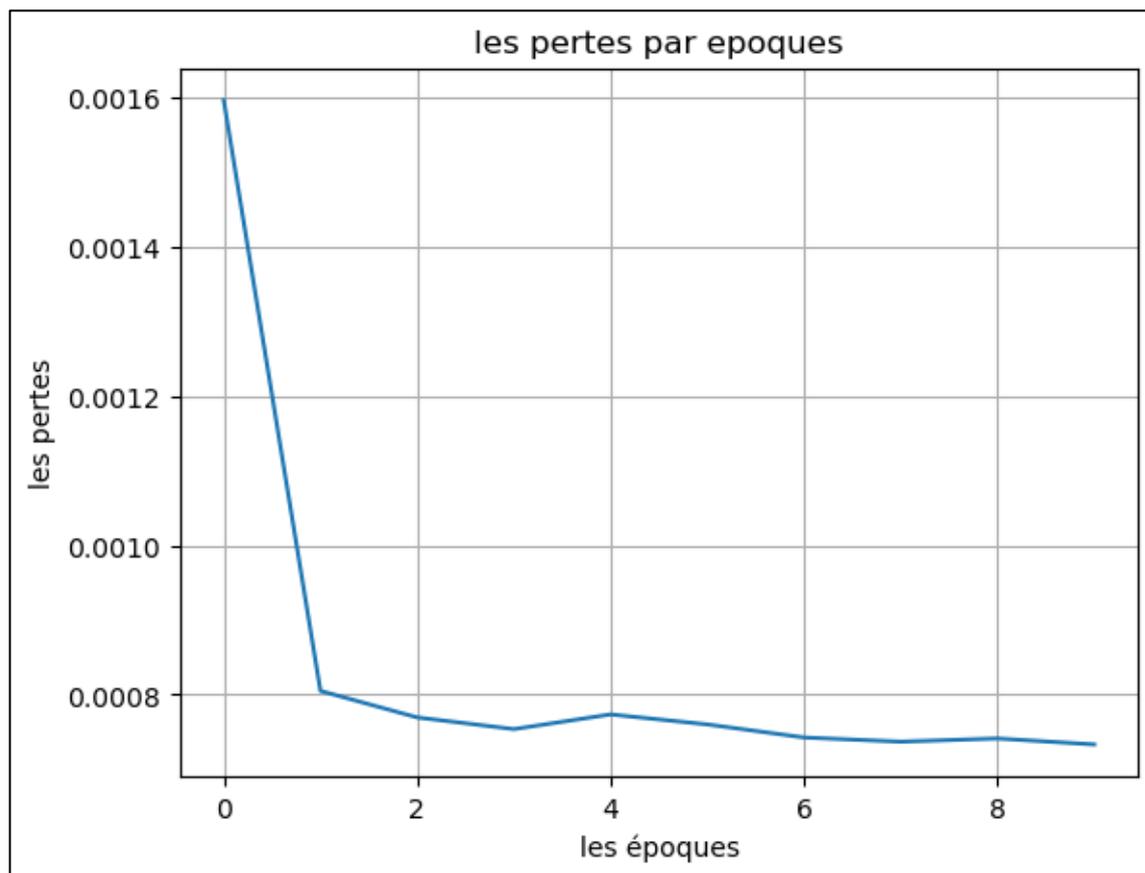
```
model.fit(train_X, train_y, epochs=10, batch_size=1)
```

#### 6. Evaluation de modèle

```
train_predictions = model.predict(train_X)  
test_predictions = model.predict(test_X)  
  
195/195 [=====] - 1s 1ms/step  
49/49 [=====] - 0s 2ms/step
```

## 7. Tracer les graphes

### ➤ Tracer le graphe de loss



**Figure IV.20** : Les pertes par époques pour la prédiction de la température en utilisant GRU

➤ Les valeurs actuelles de test de température avec les valeurs prédites

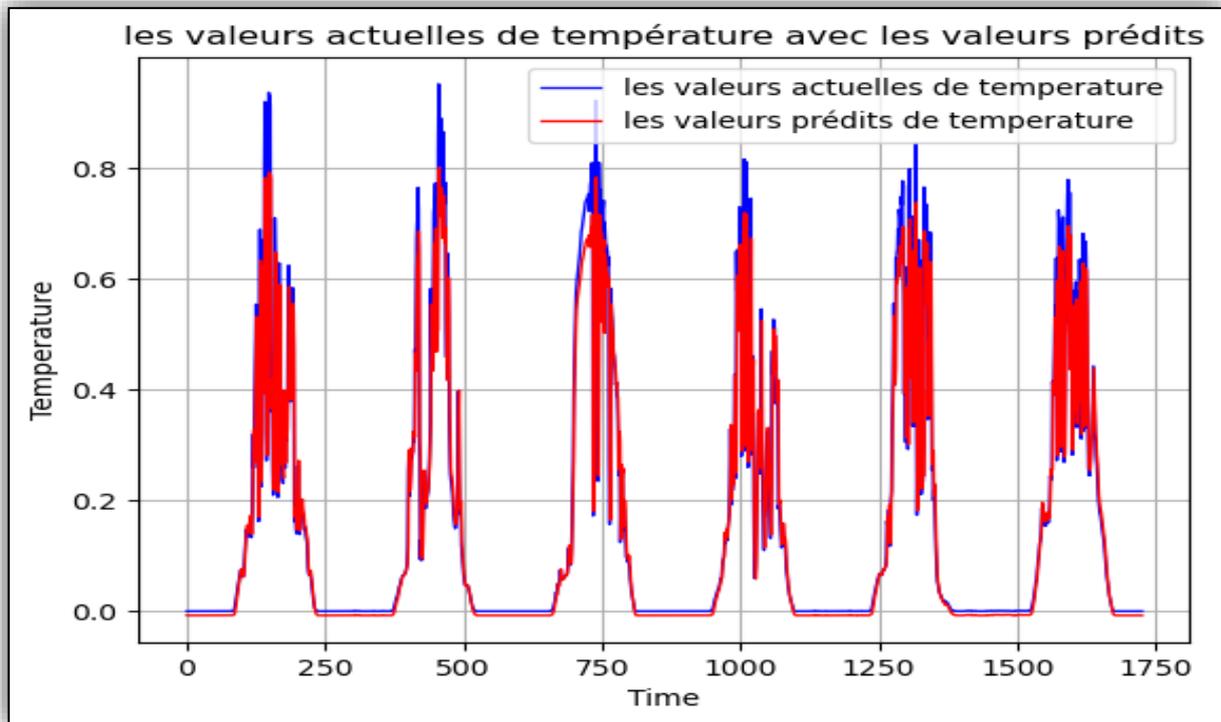


Figure IV.21 : les valeurs actuelles de température et les valeurs prédites par GRU

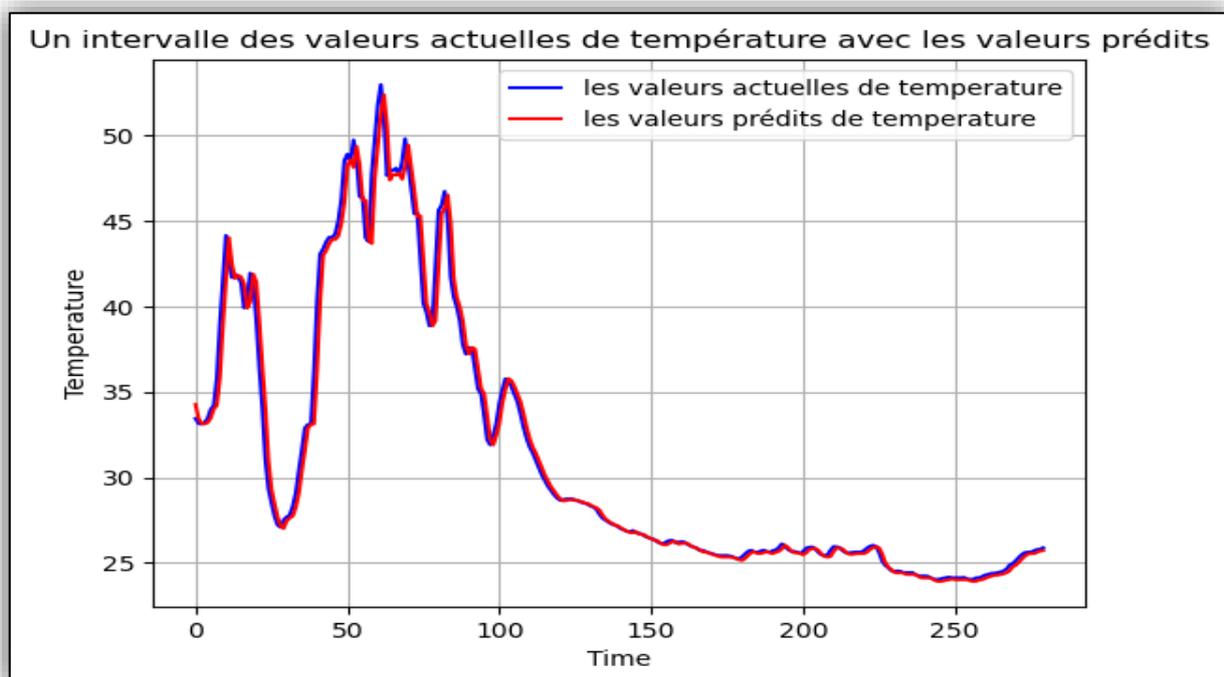


Figure IV.22 : Un intervalle les valeurs actuelles de test de température avec les valeurs prédites par GRU

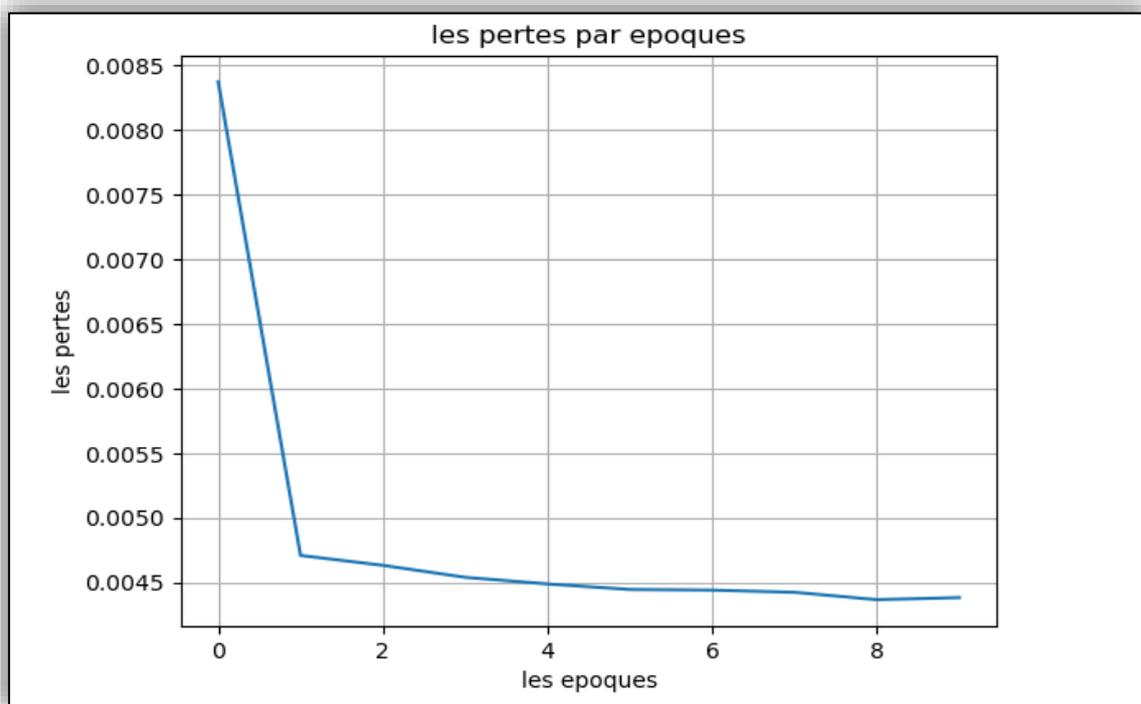
## ✚ Pour la prédiction de l'irradiance

### I.5 Sélectionner le 'Target' et choix de 'features'

```
scaler = MinMaxScaler(feature_range=(0, 1))  
weather_data['Irradiance'] = scaler.fit_transform(weather_data['Irradiance'].values.reshape(-1, 1))
```

### Les graphes de prédiction

#### ➤ Tracer le graphe de loss



**Figure IV.23 :** Les pertes par époques pour la prédiction de l'irradiance en utilisant GRU

➤ Les graphes de la prédiction d'irradiances

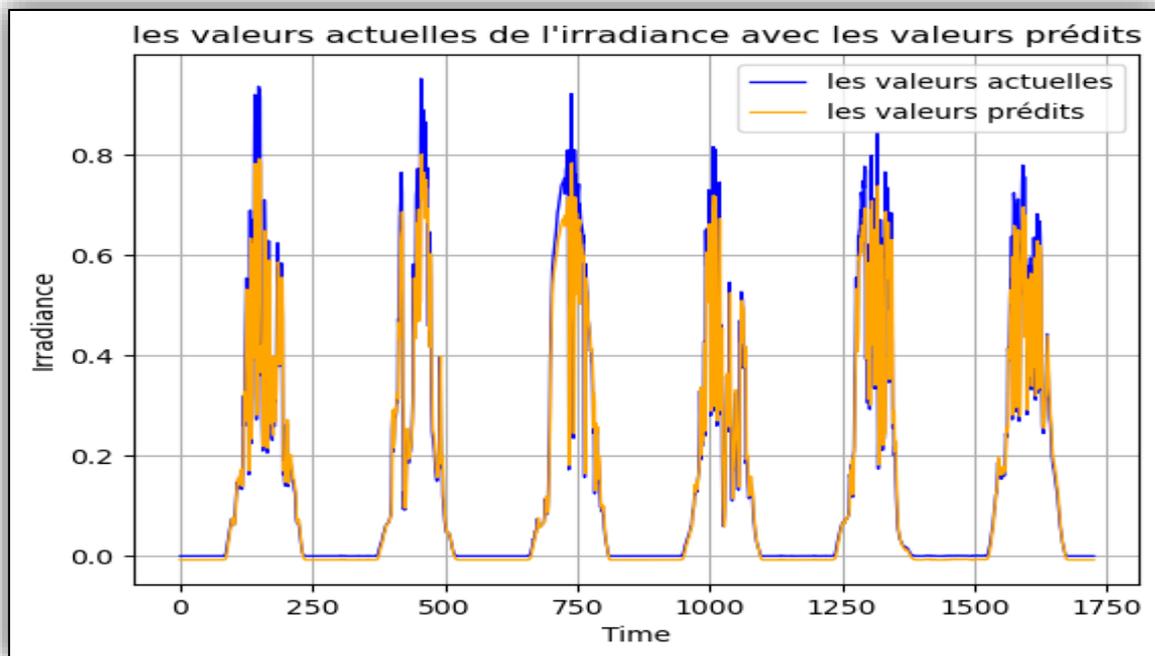


Figure IV.24 : les valeurs actuelles de l'irradiance et les valeurs prédites par GRU

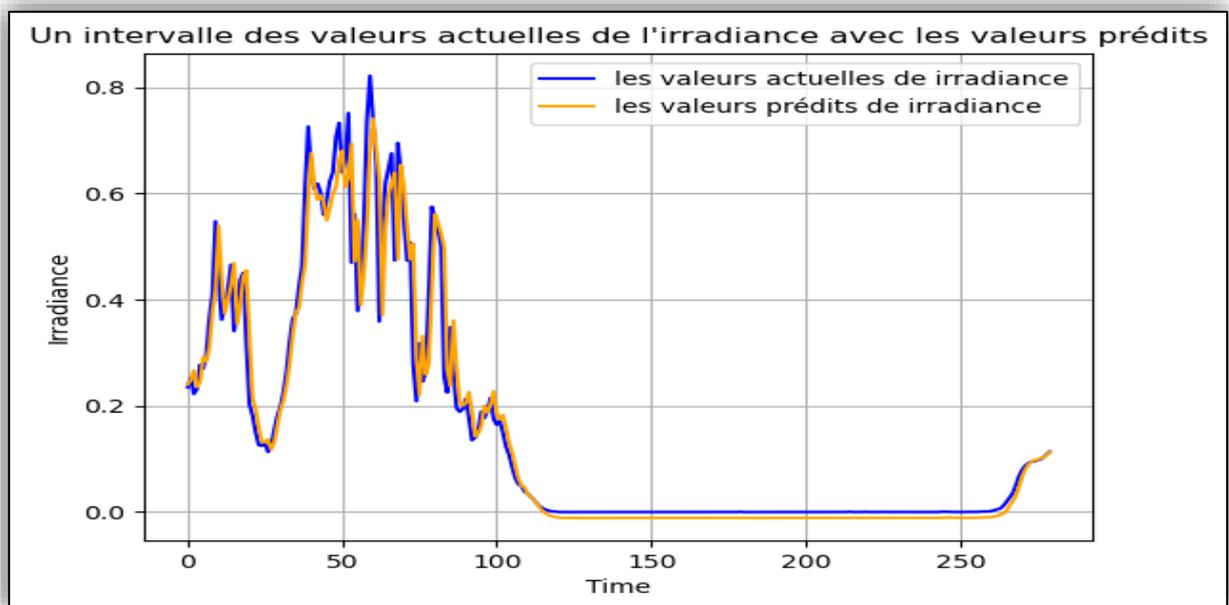


Figure IV.25 : Un intervalle les valeurs actuelles de test de l'irradiance avec les valeurs prédites par GRU

**II.5 Les erreurs**

	<b>Température</b>	<b>L'irradiance</b>
<b>Erreur absolue Moyenne</b>	0.7074079976026615	0.03410163946184485
<b>Erreur quadratique Moyenne</b>	1.25957189884132	0.00454893697354292 24
<b>Racine carré de l'erreur quadratique moyenne</b>	1.1223065084197454	0.06744580767952091

**Tableau IV.4 :** Les résultats des métriques pour la température et l'irradiance (GRU).

Pour le modèle de GRU, les métriques sont bonnes par rapport au modèle de LSTM, mais on remarque qu'il y a une déférence entre les valeurs actuelles et les valeurs prédites dans plusieurs intervalles.

#### IV.4. Comparaison entre les modèles

La comparaison entre les différents modèles est basée sur l'utilisation des métriques d'évaluation, pour cela on va les valeurs de MAE, MSE et RMSE des modèles étudiés.

Les métriques		Machine learning		Deep Learning	
		DT	RT	LSTM	GRU
MAE	IRR	0.18	10.35	26.80	0.034
	TEMP	1.59e-17	0.21	0.47	0.71
MSE	IRR	4.17	554.73	4017.62	0.004
	TEMP	5.68e-32	0.18	0.89	1.26
RMSE	IRR	2.04	23.55	63.38	0.07
	TEMP	2.38e-16	0.43	0.94	1.12

**Tableau IV.5 :** Comparaison entre les valeurs des métriques des différents modèles

L'analyse des modèles formés sur l'ensemble de données dispartité notable dans les erreurs.

Les observations peuvent être organisées dans la liste suivante :

- ✓ Les modèles Décision tree et GRU démontrent les meilleures performances.
- ✓ Le modèle de LSTM présente un écart considérable par rapport aux autres modèles, affichant un grand écart pour la prédiction de l'irradiance
- ✓ Les forêts aléatoires et LSTM fonctionnent mal sur cet ensemble de données, indiquant une baisse de leurs capacités prédictives.

### IV.5. Examen des performances des modèles

Dans cette partie on va examiner notre modèle pour confirmer la précision des résultats que on va obtenir, en utilisant les données des jours 29 et 30 (voir annexe B).

D	H	M	Les valeurs actuelles		Les valeurs prédites			
					Machine learning		Deep Learning	
					DT	RF	LSTM	GRU
29	2	25	IRR	0.06	0.00	0.00	0.10	0.00
			TEMP	26.22	27.39	27.9543	25.69	26.72
29	11	30	IRR	662.2	640.94	575.6308	256.98	120.05
			TEMP	53.08	51.21	50.8620	51.70	54.89
30	12	35	IRR	594.55	234.17	270.4282	4.34	6.18
			TEMP	45.76	46.50	47.5511	32.57	15.35
30	17	5	IRR	249	320.93	300.8602	3.14	6.07
			TEMP	41.87	46.89	46.0713	30.31	15.37

**Tableau IV.6 :** les valeurs prédites pour examiner les performances des modèles

D'après ce tableau on peut conclure 4 points principaux :

1-pour le premier jour prédit (le jour 29 après les données qu'on a utilisées) tous les modèles donnent des valeurs proches au réelles.

2-il y a une diversité entre les modèles pour la prédiction de l'irradiance, les modèles de Deep Learning affichent des valeurs prédites largement déférentes pour les valeurs actuelles

3- Plus on s'éloignons du premier jour, plus l'écart entre les valeurs actuelles et les valeurs prédites augmente.

4- En générale, on peut remarquer que l'arbre de décision donne les meilleurs résultats.

## IV.6. Conclusion

En conclusion, notre étude comparative des méthodes de machine Learning et Deep Learning a mis en évidence plusieurs aspects importants. Tout d'abord, nous avons constaté que le machine learning basé sur des modèles de l'arbre de décision et la forêt aléatoire, offre une approche solide pour résoudre de nombreux problèmes de régression. Ces méthodes sont relativement simples à mettre en œuvre et interpréter, ce qui les rend accessibles même aux praticiens moins expérimentés.

Il convient également de noter que le Deep Learning nécessite souvent des ressources plus importantes en termes de puissance de calcul et de données d'entraînement. Les architectures de réseaux de neurones profonds sont souvent plus complexes et nécessitent un volume de données plus important pour obtenir des performances optimales. Par conséquent, l'application du Deep Learning peut être plus coûteuse en termes de temps et de ressources, ce qui doit être pris en compte lors du choix de la méthode appropriée.

En fin de compte, le choix entre les méthodes de machine Learning et Deep Learning dépend du contexte spécifique de la tâche, des ressources disponibles et le types des données. Si on traite des problèmes relativement simples avec des ensembles de données limités, le machine learning traditionnel peut offrir des résultats satisfaisants. Cependant, si vous êtes confronté à des tâches complexes nécessitant une compréhension approfondie des données, le Deep Learning se révèle souvent être la meilleure approche, offrant une précision supérieure et une capacité à extraire des motifs et des caractéristiques subtiles.

En somme, notre étude comparative souligne l'importance de choisir la bonne méthode en fonction de la complexité de la tâche et le type des données, et dans notre cas le modèle de l'arbre de décision nous donne une précision très proche au valeurs réelles et on a aussi observer qu'il y a une limitation pour cette étude, c'est que le taille de données influence sur la durée de la prédiction et la précision de la prédiction.

### Conclusion générale

En conclusion, notre étude a démontré que l'application de techniques d'apprentissage automatique, à la fois des modèles de machine Learning et de Deep Learning, présente un potentiel considérable pour améliorer les performances des panneaux photovoltaïques. Les résultats obtenus ont montré que ces approches peuvent contribuer à la prédiction précise de la production d'énergie solaire, à l'optimisation des paramètres des panneaux, et à la détection rapide de tout dysfonctionnement.

Nous avons comparé plusieurs modèles de machine learning, tels que l'arbre de décision et les forêts aléatoires, ainsi que des modèles de Deep Learning, tels que les réseaux de neurones récurrents (RNN). Chaque modèle a été évalué en termes de précision, de robustesse aux variations environnementales et de temps de calcul.

Cependant, il convient de noter que le choix du modèle dépendra également des données disponibles, de la complexité du problème et des contraintes spécifiques du domaine. Une analyse approfondie des caractéristiques des données sont nécessaires pour sélectionner le modèle le plus approprié.

En outre, la disponibilité de grandes quantités de données de qualité est essentielle pour entraîner avec succès les modèles de Deep Learning. L'accès à des données fiables et représentatives des performances des panneaux photovoltaïques est un défi majeur, mais des initiatives de collecte de données à grande échelle et de partage de données pourraient aider à surmonter cette limitation.

Dans l'ensemble, notre étude souligne le potentiel des modèles de machine Learning et de Deep Learning pour améliorer les performances des panneaux photovoltaïques. Ces approches peuvent contribuer à une meilleure gestion de l'énergie solaire, à une maintenance proactive des installations et à une augmentation de l'efficacité globale du système. Des recherches futures devraient se concentrer sur l'exploration de nouvelles architectures de réseaux neuronaux, l'optimisation des hyperparamètres et l'intégration de données multi-source pour continuer à progresser dans ce domaine passionnant de l'énergie solaire et de l'intelligence artificielle.

## Bibliography

- [1] Géron, A. (2017). Hands-On\_Machine\_Learning\_with\_Scikit-learn and tensorflow. USA: O'Reilly Media..
- [2] Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press.
- [3] Moualek, DY - Université Abou Bakr Belkaid Tlemcen, "Deep Learning pour la classification des images", Mémoire de master, 2017.
- [4] Hamla, A. r, Chebout L, "Reconnaissance automatique de gestes manuels en langue des signes avec le Deep learning", Mémoire de master, 2021/2022.
- [5] Senani, S, "Réseaux de neurones convolutionnels pour la détection précoce de la rétinopathie diabétique", Faculté de Génie électrique et d'informatique, 10 juillet 2019.
- [6] Labiad, A, Sélection Des Mots Clés Basée Sur La Classification Et L'extraction Des Règles D'association, Juin 2017.
- [7] «[https://en.wikipedia.org/wiki/Cluster\\_analysis#/media/File:Cluster-2.svg](https://en.wikipedia.org/wiki/Cluster_analysis#/media/File:Cluster-2.svg),» [En ligne].
- [8] «<https://openclassrooms.com/fr/courses/4379436-explorez-vos-donnees-avec-des-algorithmes-non-supervises/4379443-comprenez-pourquoi-reduire-la-dimension-de-vos-donnees>,» [En ligne].
- [9] Lakhdari, S. Saidi, A, "Étude des techniques d'apprentissage semi-supervisé par regroupement", UNIVERSITÉ ABOU BEKR BELKAID TLEMCEN, 11 septembre 2017.
- [10] «Simon Bernard. Forêts aléatoires : De l'analyse des mécanismes de fonctionnement à la construction dynamique. December 2009».
- [11] «C4.5: Programs for machine learning by j. ross quinlan. Morgan kaufmann publishers, inc., 1993.».
- [12] «N. Donges. A COMPLETE GUIDE TO THE RANDOM FOREST ALGORITHM».

- [13] «Moualek, D. Y, « Deep Learning pour la classification des images », Mémoire de master, Université Abou Bakr Belkaid, Tlemcen, 2017».
- [14] «Boughaba Mohammed et Boukhris Brahim, « L`apprentissage profond (Deep Learning) pour la classification et la recherche d'images par le contenu », Mémoire de master, Université kasdi merbah, Ouargla, 2016-2017».
- [15] «Alom, M. Z., Taha, T. M., Yakopcic, C., Westberg, S., Sidike, P., Nasrin, M. S., Van Esesn, B. C., Awwal, A. A. S., and Asari, V. K. (2018). The history began from alexnet : A comprehensive survey on deep learning approaches».
- [16] «Arulkumaran, K., Deisenroth, M. P., Brundage, M., and Bharath, A. A. (2017). A brief survey of deep reinforcement learning».
- [17] «Le, X. H., Ho, H. V., Lee, G., & Jung, S. (2019). Application of Long Short-Term Memory (LSTM) neural network for flood forecasting. Water (Switzerland)».
- [18] «Meroua Khene et Sabrina Souid - Détection d'activité vocale utilisant l'apprentissage profond - 2019 – 2020 - Université de Ghardaïa».
- [19] «Neataptic.js - GRU (wagenaartje.github.io),» [En ligne].
- [20] A. Labouret-Michel Villos, "Energie solaire photovoltaïque", édition Dunod, Paris, 2006..
- [21] M.N. Mchaliikh et Ch. Hmada''Modélisation et simulation d'un système photovoltaïque en fonctionnement autonome et connecté au réseau' Mémoire de Master Université Kasdi Merbah–Ouargla 2013.
- [22] F. Kininger, "photovoltaic systems technology", university of kassel, 2003] [Mr. Bensalah Belqassim et Mr. Filali Mohammed, étude et réalisation d'un circuit de commande avec MPPT d'un hacheur pour système photovoltaïque.
- [23] M. A. Hamzaoui, « Améliorations d'une chaîne de conversion de l'énergie solaire en électricité autonome en vue d'application dans les pays en voie de développement », thèse de doctorat à l'Université de Reims Champagne-Ardenne, 2019. Français.

- [24] S, ABOUDA « CONTRIBUTION A LA COMMANDE DES SYSTEMES PHOTOVOLTAIQUES: APPLICATION AUX SYSTEMES DE POMPAGE », thèse de doctorat à l'Université de REIMS CHAMPAGNE.
- [25] C. Lerouge, Recherche & Industrie Photovoltaïque (PV) aux Etats-Unis, Science physique, 2006.
- [26] Bensalem, S « Effets de la température sur les paramètres et caractéristiques des cellules solaire », université Farhat Abbas- Setif, mémoire de magister, 2011..
- [27] P. Cumunel, Braun.J-P, Labouret. A, 'cellules solaires', édition dunod Paris 2001..
- [28] V. Barth, « Dipyrrométhènes métallés (Co,Ni,Cu) et dipyrannilidènes: de nouveaux matériaux organiques pour la conversion photovoltaïque de l'énergie solaire », thèse de doctorat à l'Université Pierre et Marie Curie - Paris VI, 2014.
- [29] V. Guiheneuf, « Approche multi-physique du vieillissement des matériaux pour application photovoltaïque », thèse de doctorat à l'Université Paris-Est, 2017.
- [30] G. Mailys, « Réalisation de nouvelles structures de cellules solaires photovoltaïques à partir de couches minces de silicium cristallin sur substrat de silicium préparé par frittage de poudres », thèse de doctorat à INSA de Lyon, 2012..
- [31] M. P. Paranthaman, W. Wong-Ng, et R. N. Bhattacharya, Éd., Semiconductor Materials for Solar Photovoltaic Cells, vol. 218. Cham: Springer International Publishing, 2016.
- [32] Anne Labouret, Pascal Cumunel, "Cellules solaires Les bases de l'énergie photovoltaïque", Dunod, 5e édition, 2010.
- [33] A. Toulait, A. Rachid, « Modélisation et simulation sous MATALAB/SIMULINK d'un système photovoltaïque adapté par une commande MPPT », Mémoire de Master académique, Université Mouloud Mammeri, Tizi-Ouzou, 2014.
- [34] B. Laurent Adonis, « Elaboration de cellules solaires organiques à base de tétra-tert-butyl-phthalocyanine de zinc (TTB-ZnPc) ». Mémoire de Master, Université de Québec, 2012..

- [35] M. Missoum, « Contribution de l'énergie photovoltaïque dans la performance énergétique de l'habitat à haute qualité énergétique en Algérie ». Mémoire de Magister, Université Hassiba Ben Bouali, Chlef, 2011.
- [36] D. Boukhers, « Optimisation d'un système d'Energie photovoltaïque application au pompage ». Mémoire de Magister, Université Mentouri, Constantine, 2007.
- [37] M. Mehdi, « Injection de l'électricité produite par les énergies renouvelables dans le réseau électrique », Mémoire de Master académique, Université Abou Bekr Belkaïd, Tlemcen, 2010.
- [38] A. MORSLI, « Etude d'un système solaire photovoltaïque », Mémoire de Magister, ECOLE NATIONALE POLYTECHNIQUE, 2010..
- [39] A.M.Green, K. Emery, Y. Hishikawa, W.Warta, «Solar cell efficiency tables (version 50) », 2017.
- [40] Z. M'hamed and A. Cheurfa, "Etude et optimisation d'un système photovoltaïque," 2019/2020.
- [41] K. Helali, "Modélisation D'une Cellule Photovoltaïque: Etude Comparative," Mémoire de MAGISTER EN ELECTROTECHNIQUE, Tizi-Ouzou, 25 juin 2012.
- [42] M.Mouzaoui et H.Mezri, 'Modélisation et simulation d'un système photovoltaïque', Mémoire de fin d'études, Université de Bejaia, 2014.
- [43] D. Moussaoui and D. Mesbah, "Modélisation des paramètres physiques en utilisant des données issues d'un réseau de capteurs sans fils en agriculture de précision," 13 juillet 2021, Ecole Nationale Polytechnique.
- [44] «J.-O. J.-S. V.F.Rodriguez-Galiano, B.Ghimire, "An assessment of the effectiveness of a random forest classifier for land-cover classification," vol. 67, pp. 93–104, 2012.».

