



MEMOIRE DE MASTER

Pour l'obtention du diplôme de Master en Hydraulique

Option: CONCEPTION DES SYSTEMES D'ASSAINISSEMENT

THEME :

**Contribution à la prédiction des précipitations par
l'apprentissage automatique (machine learning)**

Présenté par :

BELASLA YASMINE

Devant les membres du jury

Nom et Prénoms	Grade	Qualité
SALAH BOUALEM	Professeur	Président
KAHLERRAS MALIKA	M.C.B	Examineur
BENCHIHB NEDJOUA	M.C.B	Examineur
TOUMI SAMIR	M.C.B	Promoteur

Session Novembre 2024

ملخص : يشكل تغير الظروف المناخية وتأثيرها على الموارد المائية تحديات كبيرة لإدارة البنى التحتية للمياه، وخاصة شبكات الصرف الصحي. في هذا السياق، يصبح التنبؤ بهطول الأمطار أمرًا بالغ الأهمية. تهدف هذه الدراسة إلى استخدام الشبكات العصبية من نوع "الذاكرة الطويلة والقصيرة المدى (LSTM)" للتنبؤ بكميات الأمطار على مدى 120 شهرًا. يهدف البحث إلى تحسين القدرة على التنبؤ بالأمطار الشديدة، مما يساعد في تحسين تخطيط شبكات الصرف الصحي. أظهرت النتائج كفاءة نموذج LSTM في التنبؤ بالسلاسل الزمنية الهيدرولوجية، حيث أظهر توافقًا جيدًا بين القيم الحقيقية والقيم المتوقعة.

الكلمات المفتاحية : الأمطار، التنبؤ، الشبكات العصبية LSTM، السلاسل الزمنية، الموارد المائية، شبكات الصرف الصحي

Abstract

The evolution of climate conditions and their impact on water resources present new challenges for the management of hydraulic infrastructures, especially wastewater networks. In this context, precipitation forecasting becomes crucial. This study explores the application of Long Short-Term Memory (LSTM) neural networks to predict rainfall over a period of 120 months. The goal is to enhance the anticipation of extreme rainfall events and improve the planning of sewage systems. The results demonstrate the effectiveness of the LSTM model in predicting hydrological time series with a good fit between real and predicted values

Keywords: • Precipitation, Prediction, LSTM neural networks, Time series, Water resources, Sanitation networks

Résumé

L'évolution des conditions climatiques et leur impact sur les ressources en eau imposent de nouveaux défis pour la gestion des infrastructures hydrauliques, notamment les réseaux d'assainissement. Dans ce contexte, la prévision des précipitations devient cruciale. Ce travail explore l'application des réseaux de neurones à mémoire longue et courte durée (LSTM) pour prédire les précipitations sur une période de 120 mois. L'objectif est d'améliorer l'anticipation des précipitations extrêmes et de permettre une meilleure planification des réseaux d'assainissement. Les résultats obtenus montrent l'efficacité du modèle LSTM dans la prévision des séries chronologiques hydrologiques avec un bon ajustement entre les valeurs réelles et prédites.

Mots-clés : • Précipitations, Prédiction Réseaux de neurones LSTM, Séries chronologiques, Ressources en eau , Réseaux d'assainissement

Remerciements

Tout d'abord, je remercie Allah le Tout-Puissant de m'avoir donné la santé, la force, la patience et la capacité de mener à bien ce travail.

Je remercie mon encadrante, Mr. TOUMI, dont les conseils et le soutien, du début à la fin, m'ont permis de développer mes compétences en recherche et ma compréhension du sujet.

Je tiens également à remercier les membres du jury pour avoir consacré leur temps à examiner et évaluer mon travail.

Dédicaces

Avant tout, je souhaite dédier ce modeste travail à ma chère mère, ainsi qu'à mon père, pour leurs sacrifices, leur patience, leur compréhension, leur soutien et leur amour inconditionnel qui m'ont aidé à surmonter de nombreux obstacles dans ma vie. Je leur serai éternellement reconnaissant.

À mes deux sœurs, Meriem et Widad, ainsi qu'à mon frère, Younes, je prie Allah de veiller sur vous, de vous protéger et de vous accorder une vie heureuse et épanouissante.

Introduction Générale	1
1 Synthèse bibliographique:	2
1.1 Introduction	2
1.2 Définition	2
1.3 Processus d'analyse des séries chronologiques	2
1.3.1 Collecte de données	2
1.3.2 Visualisation des données:	3
1.3.3 Prétraitement des données	3
1.3.4 Modélisation	3
1.3.5 Prévision	3
1.3.6 Validation :	3
1.4 Approches de prédiction des séries temporelles	3
1.4.1 Approches Traditionnelles	3
1.4.1.1 Modèle de régression linéaire	3
1.4.1.2 Modèle de régression non linéaire	4
1.4.2 Approches modernes	4
1.4.2.1 Approches basées sur l'apprentissage automatique (ML)	4
1.4.2.2 Approches basées sur l'apprentissage profond (DL)	5
1.4.2.3 La fonction d'activation	6
1.4.2.4 Réseaux de neurones récurrents (RNN)	8
1.4.2.5 Les variantes des réseaux de neurones récurrents	8
1.4.2.6 Fonctionnement du réseau LSTM	9
1.5 Exemples de la prévision hydrologique par les réseaux de neurones artificiels :	11
1.6 Méthodologie:	12
1.6.1 validation croisée,	12
1.7 la validation croisée des séries chronologiques	12
1.8 conclusion:	13
2 EXPÉRIMENTATION:	14
2.1 introduction:	14
2.2 Environnement de travail:	14
2.2.1 Le langage Python:	14
2.2.2 Environnement de développement:	15
2.2.3 Les bibliothèques utilisées:	15
2.3 Conclusion:	16

3	Application des réseaux LSTM pour la prévision des précipitations dans la zone de Teffessour	17
3.1	LA ZONE DE TEFFESSOUR :	17
3.2	Préparation des données:	17
3.3	la visualisation et l'analyse de jeu de donnée:	18
3.4	Résultats et discussion:	19
3.4.1	architecture de modèle:	19
3.5	Mesures de performance pour l'évaluation	20
3.5.1	Erreur quadratique moyenne (MSE)	20
3.5.2	Coefficient de détermination (R^2)	20
3.5.3	les résultats des prédictions du modèle:	21
	Conclusion Générale	24
	Bibliographie	25

1.1	Étapes d'analyse des séries chronologiques	2
1.2	relation entre ML /IA/DL	4
1.3	Architecture des réseaux de neurones	5
1.4	Un neurone réel. Un neurone artificiel	6
1.5	Présentation de la fonction d'activation ReLU et sa dérivée	7
1.6	Présentation de la fonction d'activation Tanh et sa dérivée	7
1.7	Représentation graphique de la fonction sigmoïde	8
1.8	Architecture d'un réseau de neurones récurrent (RNN) (source : Colah's Blog, 2015)	8
1.9	Architecture d'un réseau LSTM	9
1.10	Structure de l'état de la cellule LSTM	9
1.11	Porte d'oubli dans une cellule LSTM	9
1.12	Mise à jour de l'état de la cellule LSTM	10
1.13	Mise à jour de l'ancien état de la cellule	10
1.14	Sortie du réseau LSTM	10
1.15	validation croisée standard	12
1.16	la validation croisée standard et la validation croisée pour les séries temporelles	13
2.1	python	14
2.2	google colab	15
3.1	carte d'altitude de la commune de teffsour	17
3.2	les précipitations maximales	18
3.3	Décomposition de la série temporelle	19
3.4	L'architecture de modèle	19
3.5	L'évolution de la MSE pour tous les folds.	20
3.6	L'évolution de R^2 pour tous les folds.	21
3.7	Perte d'entraînement et perte de validation au cours des FOLDS	21
3.8	LA prédiction de 120 mois dans le futur	22
3.9	Comparaison entre les valeurs réelles des précipitations et les valeurs prédites	22

LIST OF TABLES

3.1	Informations sur la station MERINE	18
3.2	serie des precipitations	27

LISTE DES ABRÉVIATIONS

LSTM	: Long Short-Term Memory
DL	: Deep Learning
ML	: Machine Learning
ANN	: Artificial Neural Network (Réseau de Neurones Artificiels)
RNN	: Recurrent Neural Network (Réseau de Neurones Récurrents)
MSE	: Mean Squared Error (Erreur Quadratique Moyenne)
R^2	: Coefficient de Détermination
PMAX	: Précipitation Maximale
IA	: Intelligence Artificielle.

L'évolution des conditions climatiques et leur impact sur les ressources en eau font de la prévision des précipitations un enjeu majeur dans le domaine de la gestion des ressources hydriques et des réseaux d'assainissement. Des précipitations imprévisibles peuvent entraîner des inondations, une surcharge des infrastructures d'assainissement et des impacts négatifs sur l'environnement et la santé publique. Les techniques traditionnelles de prévision peuvent parfois se révéler insuffisantes face à la complexité et à la variabilité des phénomènes météorologiques.

C'est dans ce contexte que les modèles d'apprentissage profond, notamment les réseaux de neurones récurrents tels que les LSTM (Long Short Term Memory), émergent comme des outils prometteurs pour la prévision des séries chronologiques. Ces modèles permettent de capturer les relations temporelles complexes entre les données de précipitations, offrant ainsi des prévisions plus précises et fiables. Ce travail se concentre sur l'utilisation de ces modèles pour prédire les précipitations sur une période de 120 mois dans la région de Tefessour, en tenant compte des implications pour la gestion des réseaux d'assainissement.

1.1 Introduction

L'analyse des séries chronologiques temporelles est une méthode importante utilisée pour étudier des données recueillies au fil du temps. Elle permet de comprendre comment les données évoluent, d'identifier des tendances, et de prévoir des événements futurs. Ce type d'analyse est essentiel dans de nombreux domaines, comme la météo, la finance et la gestion des ressources

1.2 Définition

L'analyse des séries chronologiques temporelles consiste à examiner des données collectées à différents moments pour en extraire des informations utiles. L'objectif est de trouver des modèles dans ces données afin de faire des prévisions sur ce qui pourrait se passer à l'avenir. Cela implique d'utiliser des techniques statistiques pour identifier des tendances et des variations dans les données au fil du temps[1]

1.3 Processus d'analyse des séries chronologiques

Les étapes du processus sentimental sont illustrées ci-dessus[15] :

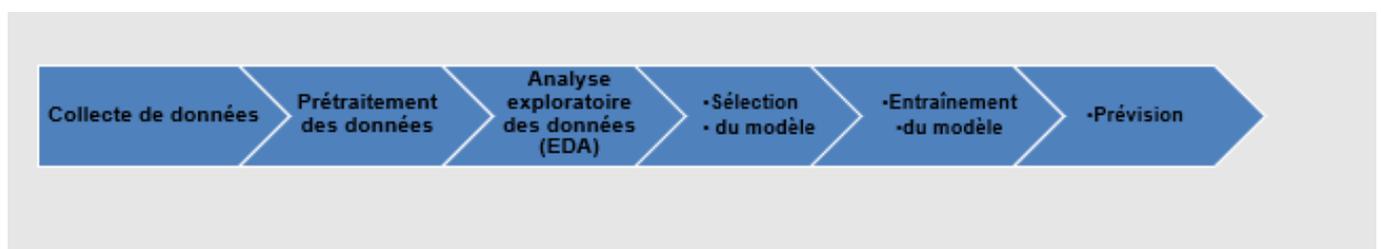


Figure 1.1: Étapes d'analyse des séries chronologiques

1.3.1 Collecte de données

Il s'agit de collecter les données souvent sous forme de chiffres mesurés à des moments précis (par exemple, les précipitations, les températures quotidiennes, les ventes mensuelles). Une fois les données extraites, elles seront ensuite préparées pour l'analyse.

1.3.2 Visualisation des données:

Tracer les données sur un graphique pour voir les tendances, les variations saisonnières et les valeurs anormales. Cela aide à comprendre comment les données évoluent.

1.3.3 Prétraitement des données

- Nettoyage : Trouver et corriger les valeurs manquantes ou incorrectes.
- Transformation : Appliquer des changements aux données si nécessaire, par exemple, pour rendre les données plus faciles à analyser

1.3.4 Modélisation

- Choix du modèle : Sélectionner un modèle adapté pour prévoir les valeurs futures, comme GRU ou les réseaux de neurones
- Estimation des paramètres : Ajuster les paramètres du modèle en utilisant des méthodes statistiques

1.3.5 Prévision

Utiliser le modèle pour faire des prédictions sur des valeurs futures basées sur les données passées.

1.3.6 Validation :

Évaluer la précision du modèle en le testant sur des données non utilisées lors de la création du modèle. [12].

1.4 Approches de prédiction des séries temporelles

1.4.1 Approches Traditionnelles

1.4.1.1 Modèle de régression linéaire

Le modèle de régression linéaire simple peut être exprimé par l'équation suivante[10] :

$$y = \beta_0 + \beta_1 x + \epsilon$$

où :

- y est la variable dépendante,
- x est la variable indépendante,
- β_0 est l'ordonnée à l'origine (intercept),
- β_1 est le coefficient de régression (pente),
- ϵ est le terme d'erreur aléatoire.

Le but est d'estimer les paramètres β_0 et β_1 en minimisant la somme des carrés des écarts entre les valeurs observées et les valeurs prédites :

$$\text{Minimiser } \sum_{i=1}^n (y_i - (\beta_0 + \beta_1 x_i))^2$$

1.4.1.2 Modèle de régression non linéaire

Contrairement à la régression linéaire, un modèle de régression non linéaire peut inclure des termes non linéaires par rapport aux paramètres. Par exemple, pour une régression polynomiale d'ordre 2, l'équation peut être écrite comme :

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \epsilon$$

où :

- y est la variable dépendante,
- x est la variable indépendante,
- $\beta_0, \beta_1, \beta_2$ sont les coefficients à estimer,
- ϵ est le terme d'erreur aléatoire.

L'objectif ici est également de minimiser la somme des carrés des écarts entre les valeurs observées et les valeurs prédites, mais la relation entre x et y est non linéaire :

$$\text{Minimiser } \sum_{i=1}^n (y_i - (\beta_0 + \beta_1 x_i + \beta_2 x_i^2))^2$$

1.4.2 Approches modernes

1.4.2.1 Approches basées sur l'apprentissage automatique (ML)

L'apprentissage machine (ML) est une branche de l'intelligence artificielle qui permet aux ordinateurs d'apprendre et de résoudre des tâches sans une programmation explicite. Il se divise en trois catégories principales : l'apprentissage supervisé, où les ordinateurs apprennent à partir de données étiquetées, l'apprentissage non supervisé, où ils découvrent des structures dans les données sans supervision et l'apprentissage par renforcement où la machine interagit avec son environnement pour apprendre de manière interactive. Ces approches sont utilisées dans une variété de domaines pour résoudre divers problèmes complexes.[12][2]

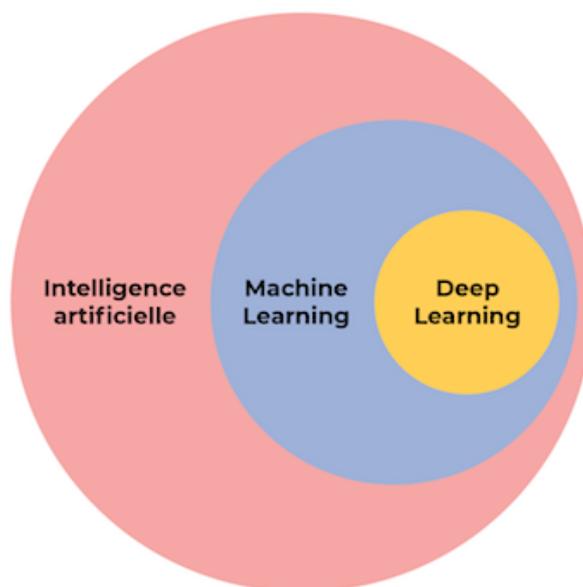


Figure 1.2: relation entre ML /IA/DL

1.4.2.2 Approches basées sur l'apprentissage profond (DL)

L'apprentissage profond est une branche de l'apprentissage machine qui utilise des réseaux de neurones artificiels multicouches pour extraire des caractéristiques complexes à partir des données. Ces réseaux passent par des couches hiérarchiques pour comprendre les données à des niveaux de complexité croissante. Cette approche est particulièrement utile pour résoudre des problèmes de traitement de données complexes, tels que la reconnaissance d'objets, la classification d'images, la prédiction, la compréhension du langage naturel, entre autres, grâce à sa capacité à apprendre des représentations de données de plus en plus abstraites.[12]

❖ Le réseau neuronal:

Dans la réalité, tous les algorithmes de Deep Learning se résument à des réseaux neuronaux. Les réseaux neuronaux, également connus sous le nom d'ANN (Artificial Neural Networks), sont des modèles de traitement de l'information qui émulent le fonctionnement d'un système nerveux biologique. Cette approche imite la manière dont le cerveau traite l'information, avec des réseaux de neurones interconnectés organisés en couches:

1Couche d'entrée : Cette couche reçoit les données brutes en entrée. Chaque neurone représente une caractéristique spécifique des données d'entrée.

2Couches cachées : Il peut y avoir une ou plusieurs couches cachées où se produit la transformation des données. Chaque neurone dans une couche cachée applique une fonction d'activation (comme ReLU, sigmoïde ou tanh) à la somme pondérée de ses entrées pour produire une sortie.

3Couche de sortie : Cette couche génère la sortie finale du réseau, en fonction de la tâche. Par exemple, pour la classification, elle peut utiliser une fonction d'activation Softmax pour produire des probabilités.

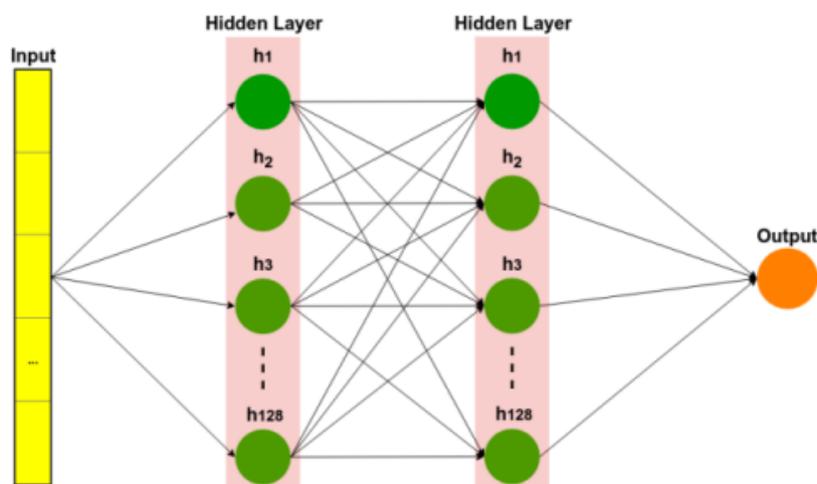


Figure 1.3: Architecture des réseaux de neurones

❖ Le neurone:

Les réseaux de neurones sont constitués de neurones artificiels qui s'inspirent des neurones réels présents dans notre cerveau. Les deux figures suivantes illustrent la représentation d'un neurone réel et d'un neurone artificiel

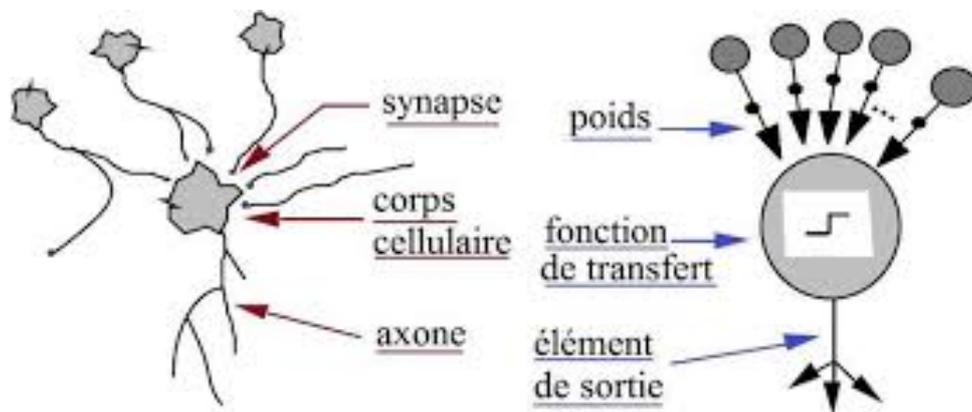


Figure 1.4: Un neurone réel. Un neurone artificiel

Dans le contexte des réseaux de neurones, les valeurs X_i représentent des données numériques, pouvant être soit des données d'entrée, soit les sorties de neurones précédents. Les poids W_i sont également des valeurs numériques, et ils représentent l'importance des entrées ou des connexions entre les neurones.

Le processus au sein d'un neurone artificiel peut être décrit en plusieurs étapes :

1. Les valeurs d'entrée (X_i) sont multipliées par les poids correspondants (W_i).
2. Les résultats de ces multiplications sont sommés.
3. Un biais (b) est ajouté à cette somme.
4. Le résultat est ensuite transmis à une fonction d'activation (f), ce qui introduit une composante non linéaire dans le processus.

1.4.2.3 La fonction d'activation

Après avoir effectué le produit entre ses entrées et ses poids, le neurone applique une non-linéarité sur ce résultat, processus appelé la fonction d'activation.

La fonction d'activation est un composant essentiel du réseau neuronal, déterminant si le neurone est activé ou non. Elle calcule la somme pondérée des entrées, ajoute le biais, puis applique une transformation non linéaire à la valeur obtenue. Cette sortie est ensuite transmise à la couche suivante.[11]

La non-linéarité introduite par la fonction d'activation est cruciale dans les réseaux de neurones. Sans elle, un réseau de neurones se réduirait à un modèle linéaire. Il existe plusieurs types de fonctions d'activation parmi lesquelles :

- **Fonction ReLU (Rectified Linear Unit)** : Elle renvoie 0 si l'entrée z est inférieure à 0, sinon elle renvoie z .

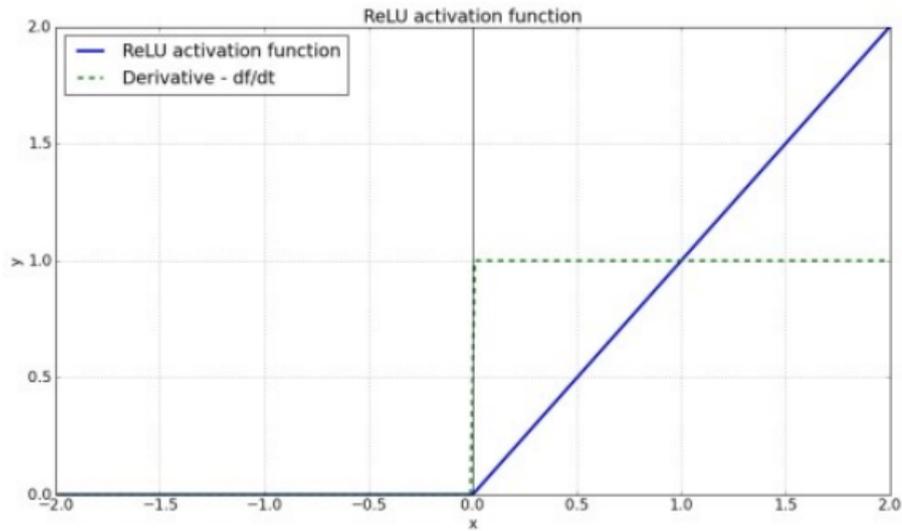


Figure 1.5: Présentation de la fonction d'activation ReLU et sa dérivée

- **Fonction tanh** : Elle transforme une valeur d'entrée en une sortie comprise entre -1 et 1, contrairement à la fonction sigmoïde qui produit des sorties entre 0 et 1. La tanh est particulièrement utile pour les données centrées autour de zéro, car elle permet d'obtenir des sorties négatives et positives, favorisant ainsi un meilleur apprentissage dans les couches profondes

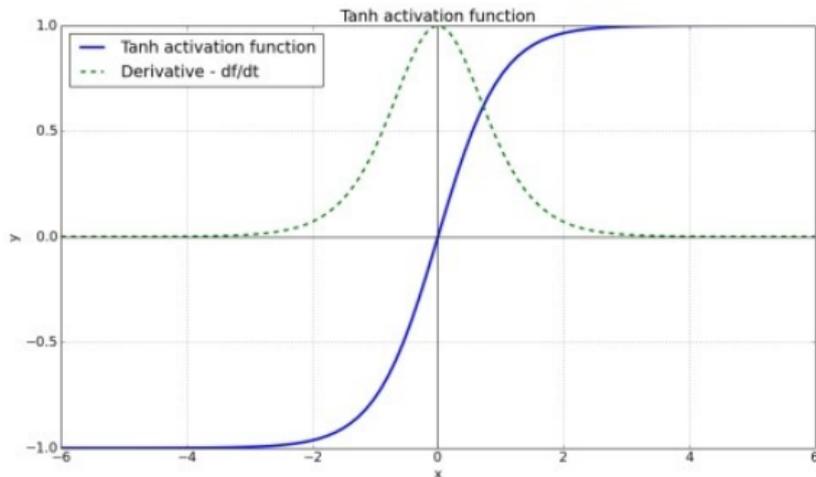


Figure 1.6: Présentation de la fonction d'activation Tanh et sa dérivée

- **Fonction sigmoïde** : Elle est bornée entre 0 et 1 et est couramment utilisée dans les tâches de classification binaire. Elle est aussi appelée fonction logistique.

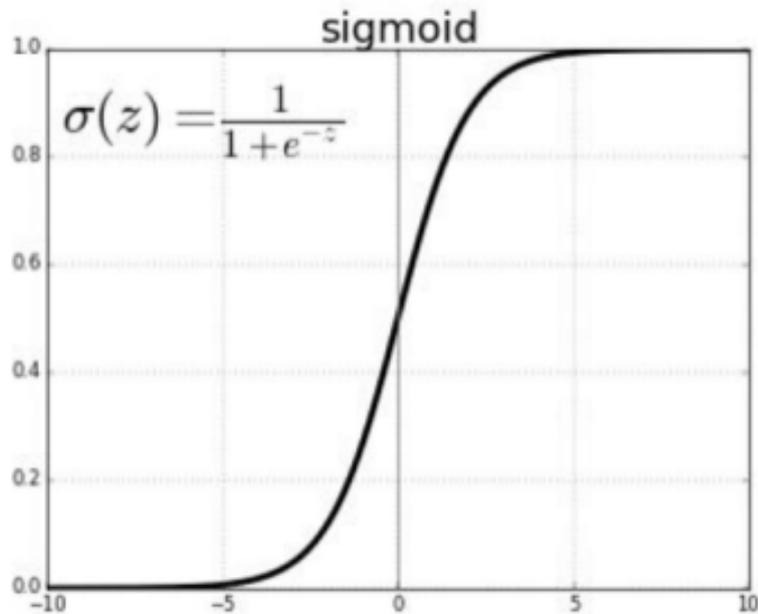


Figure 1.7: Représentation graphique de la fonction sigmoïde

1.4.2.4 Réseaux de neurones récurrents (RNN)

Les réseaux de neurones récurrents (RNN) sont une classe de réseaux de neurones qui utilisent des boucles de rétroaction pour traiter des données séquentielles en se basant sur leur mémoire interne. Contrairement aux réseaux neuronaux traditionnels, les RNN peuvent se souvenir du calcul précédent et l'utiliser pour traiter l'entrée actuelle.[3]

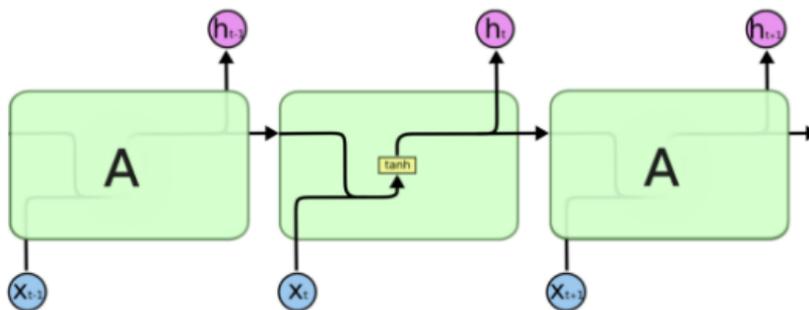


Figure 1.8: Architecture d'un réseau de neurones récurrent (RNN) (source : Colah's Blog, 2015)

1.4.2.5 Les variantes des réseaux de neurones récurrents

- Mémoire à Long Terme Courte (LSTM) :

Le réseau de neurones Long Short-Term Memory (LSTM) est un type de réseau de neurones récurrents conçu pour capturer des dépendances à long terme dans les données d'entrée grâce à des cellules de mémoire spéciales. Dans une architecture LSTM classique pour la prédiction des séries temporelles, les données d'entrée traversent une couche LSTM composée de plusieurs cellules, suivie d'une couche entièrement connectée qui sert à effectuer la prédiction.

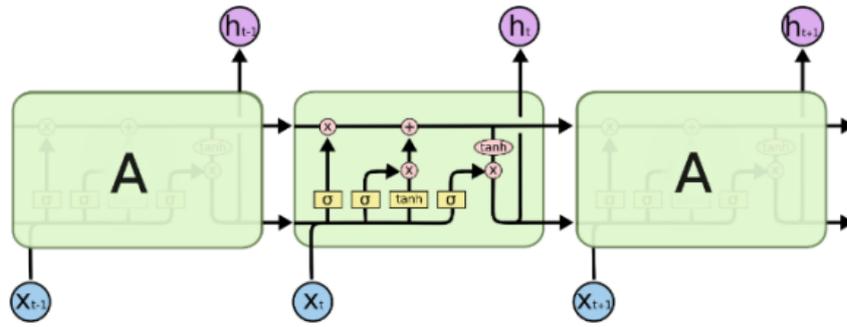


Figure 1.9: Architecture d'un réseau LSTM

1.4.2.6 Fonctionnement du réseau LSTM

Dans une cellule LSTM, des portes contrôlent le passage des informations à travers l'état de la cellule, à l'aide de couches sigmoïdes suivies de multiplications élémentaires.[12]

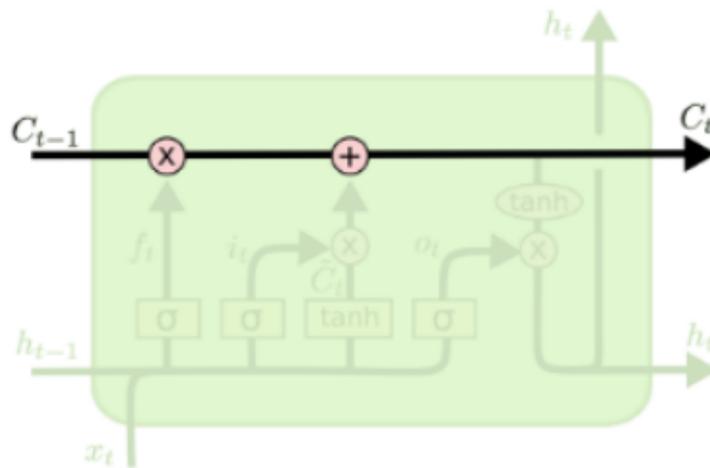


Figure 1.10: Structure de l'état de la cellule LSTM

1. **Porte d'oubli (Forget Gate)** : La première étape consiste à décider quelles informations doivent être exclues de l'état de la cellule. Cette décision est prise par une couche sigmoïde appelée "porte d'oubli".

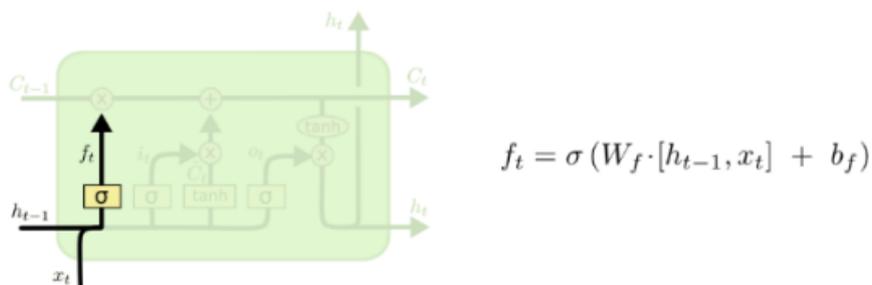


Figure 1.11: Porte d'oubli dans une cellule LSTM

- h_{t-1} : La sortie à l'instant $t - 1$
- x_t : L'entrée actuelle à l'instant t
- b_f : Le vecteur de biais
- W_f : La matrice de poids

σ : La fonction sigmoïde

2. **Mise à jour de l'état de la cellule** : La prochaine étape consiste à décider quelles nouvelles informations doivent être ajoutées à l'état de la cellule. Cela se fait en deux étapes :

- **Porte d'entrée (Input Gate)** : Nous décidons quelles informations doivent être stockées dans l'état de la cellule à l'aide d'une couche sigmoïde.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$$

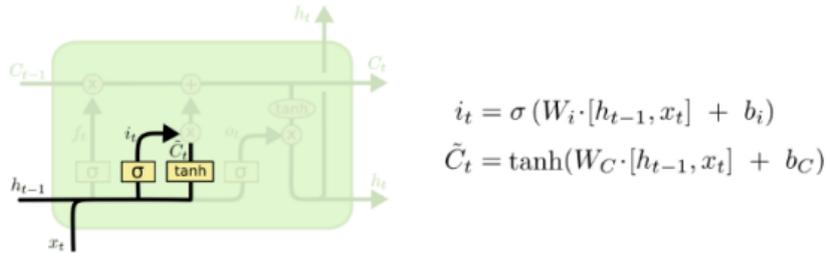


Figure 1.12: Mise à jour de l'état de la cellule LSTM

\tanh : La fonction d'activation tangente hyperbolique

C_t : La valeur candidate de l'état de la cellule

Les nouvelles informations sont calculées et ajoutées à l'ancien état pour former le nouvel état de la cellule.

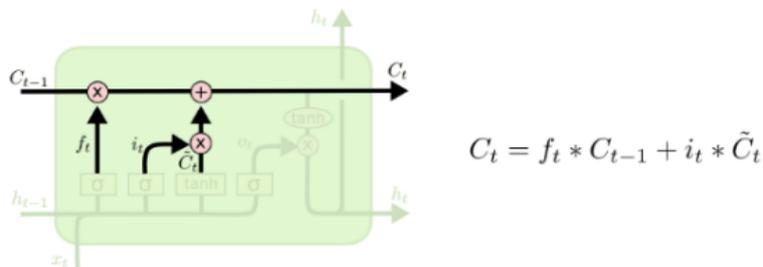


Figure 1.13: Mise à jour de l'ancien état de la cellule

3. **Calcul de la sortie** : Enfin, à partir de l'état actuel de la cellule, la sortie souhaitée est calculée.

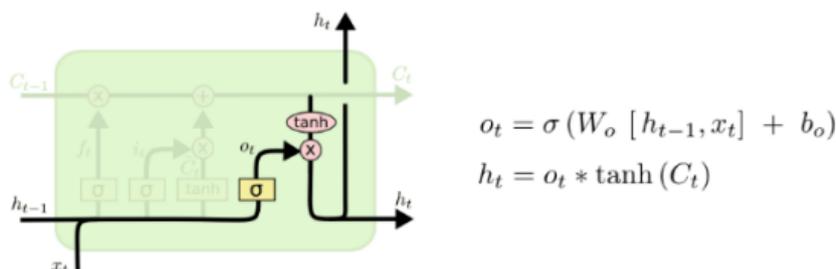


Figure 1.14: Sortie du réseau LSTM

h_t : La sortie obtenue

1.5 Exemples de la prévision hydrologique par les réseaux de neurones artificiels :

Les applications des modèles connexionnistes déjà réalisées dans le domaine hydrométéorologique concernent : la classification des données hydrologiques, la prévision des débits des rivières (crues et étiages), l'évaluation et la prévision de la qualité de l'eau, la prévision de la consommation d'eau, l'estimation des précipitations, la prévision des apports naturels aux réservoirs d'irrigation ou de production hydroélectrique. À la fin de l'année 1997, une trentaine d'articles portant sur l'application des RNA à la gestion des ressources en eau ont été publiés[16].

Une des premières applications fut celle de la prévision de la demande en eau de la ville de Camberra en Australie. Le modèle neuronal utilisé est un PMC avec 5 variables d'entrées, 7 neurones sur la couche cachée, et 1 neurone de sortie d'où la notation RNA (5,7, 1).[17]

L'apprentissage est effectué à l'aide de l'ARP standard. Les résultats obtenus sont comparés à ceux d'un modèle de régression linéaire. L'auteur remarque que la tolérance du modèle neuronal aux lacunes et aux erreurs liées aux données est excellente par rapport à celle du modèle régressif. Il en conclut que l'optimisation des procédures d'apprentissage et de configuration des RNA constitue un domaine de recherche attractif et d'avenir pour la modélisation hydrologique. Ensuite, d'autres applications à la prévision de la consommation d'eau ont mené à des résultats jugés satisfaisants (Cubero, 1991; Zhang et al., 1994)[18].

L'objectif de Zhang et al.,(1994) était de prévoir avec une grande précision la demande quotidienne d'eau d'une grande ville. Le RNA (5, 5, 1) utilisé contient cinq variables d'entrées qui sont : les prévisions météorologiques (température maximale journalière, précipitations, et climat), le jour de la semaine et la consommation du jour précédent. Les résultats obtenus en termes d'erreur relative moyenne indiquent que les prévisions du modèle neuronal sont plus précises que celles obtenues respectivement par un modèle de régression multiple, un modèle ARIMA, et un modèle de filtre de Kalman[19].

Dans la perspective d'augmenter la capacité d'adaptabilité du modèle neuronal aux changements brusques de conditions, Muster et al., (1994) ont couplé le modèle neuronal à un système de logique floue. Le modèle hybride qui en résulte est appelé « neuro-fuzzy system » et a été appliqué pour la prévision des prises d'eau au fleuve Ruhr en Allemagne. Les résultats obtenus par le modèle hybride sont peu différents de ceux obtenus par trois autres modèles : un modèle de régression multiple, un PMC avec l'ARP standard et un système de logique floue. Les auteurs expliquent la faible capacité d'adaptabilité du modèle « Neuro-Fuzzy » par l'insuffisance de données[20].

Par la suite, la moitié des applications hydrologiques concernent les prévisions des débits, crues et étiages des rivières. Karunanithi et al., (1994) ont appliqué le réseau en cascade, « cascade network », utilisant l'algorithme dit de « cascade-corrélation » (Fahlman et Lebiere, 1990) pour la prévision des débits de la rivière Huron dans l'état du Michigan. L'algorithme de « cascade-corrélation » a l'avantage de modifier la topologie du modèle lors de l'apprentissage. Les données hydrologiques utilisées sont les débits journaliers observés sur 13 ans (1960–1972)[21].

1.6 Méthodologie:

L'analyse des séries chronologiques représente une approche fondamentale dans le domaine des statistiques et de l'apprentissage automatique, visant à comprendre et à prédire les modèles de données qui évoluent au fil du temps. Compte tenu des caractéristiques uniques des données de séries chronologiques, notamment les tendances, la saisonnalité et l'autocorrélation, les techniques traditionnelles de validation croisée ne parviennent souvent pas à fournir des estimations de performances précises et fiables.

Pour relever ces défis, la validation croisée des séries chronologiques apparaît comme une méthodologie essentielle, adaptée pour respecter l'ordre temporel inhérent aux données. Cet essai explore les subtilités de la validation croisée des séries chronologiques, en soulignant sa signification, sa méthodologie, ses variations et ses considérations pratiques.

1.6.1 validation croisée,

L'objectif principal de la validation croisée des séries chronologiques est d'évaluer les performances prédictives d'un modèle d'une manière qui reflète son application future. Ceci est crucial dans divers domaines, tels que la finance, la météorologie et l'épidémiologie, où les décisions reposent sur des prévisions.[9]

Les méthodes traditionnelles de validation croisée, qui partitionnent les données de manière aléatoire, peuvent perturber la séquence temporelle, conduisant à des estimations de performances trop optimistes et à des modèles qui échouent dans les dynamiques temporelles réelles. La validation croisée des séries chronologiques préserve l'ordre chronologique, garantissant que les prédictions sont toujours basées sur des informations passées, fournissant ainsi une évaluation plus réaliste des capacités prédictives d'un modèle

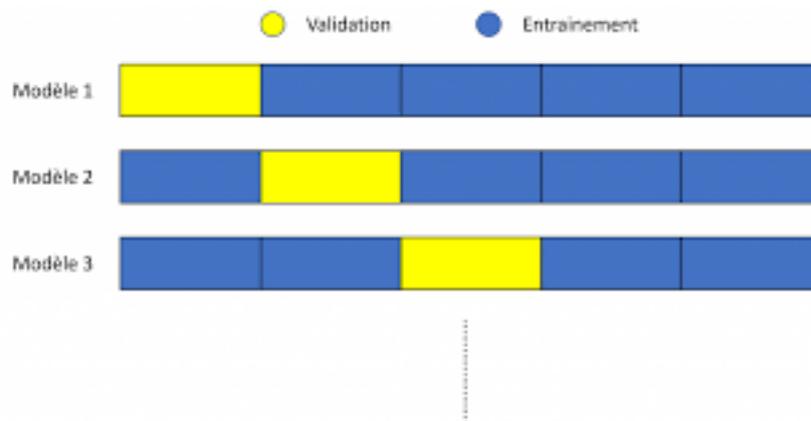


Figure 1.15: validation croisée standard

1.7 la validation croisée des séries chronologiques

L'essence de la validation croisée des séries chronologiques réside dans son approche de division séquentielle de l'ensemble de données. Contrairement au partitionnement aléatoire, il étend systématiquement l'ensemble de données d'entraînement pour inclure des observations plus récentes, tandis que l'ensemble de test comprend des observations qui suivent immédiatement celles de l'ensemble d'entraînement. Cette procédure est répétée de manière itérative, avançant à chaque fois le point de coupure entre les ensembles de formation et de test. Cette approche garantit que le modèle est validé sur différentes périodes, capturant diverses dynamiques temporelles et changements structurels potentiels dans les données.[9]

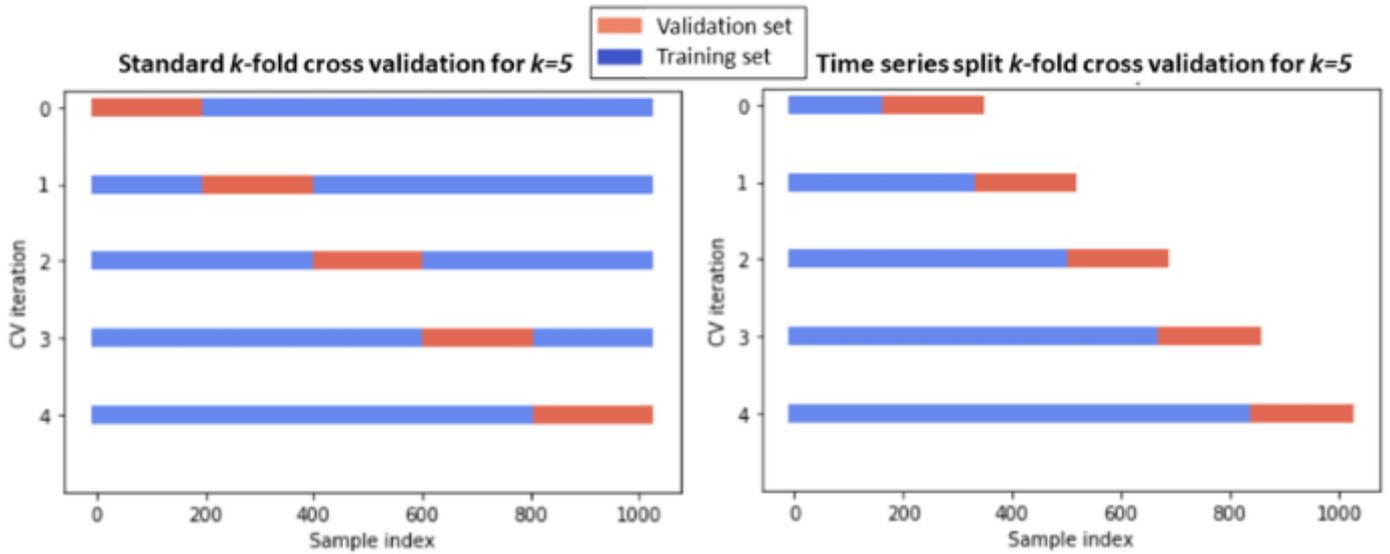


Figure 1.16: la validation croisée standard et la validation croisée pour les séries temporelles

1.8 conclusion:

Dans ce chapitre, nous avons réalisé une revue de la littérature sur l'analyse des séries chronologiques, en particulier sur l'utilisation des modèles LSTM (Long Short Term Memory). Les résultats de notre analyse montrent que les LSTM se distinguent par leur capacité à capturer efficacement les dépendances à long terme dans les données temporelles, ce qui les rend particulièrement adaptés à la modélisation des séries chronologiques complexes, comme les précipitations.

En comparaison avec les approches traditionnelles, telles que les modèles de régression linéaire et non linéaire, les LSTM offrent une meilleure flexibilité et une capacité supérieure à gérer les non-linéarités et les dépendances temporelles. Cela en fait une solution privilégiée pour les prévisions de séries temporelles, notamment dans des contextes où les données présentent des variations irrégulières ou des patterns de long terme.

Ces avancées soulignent l'importance croissante des LSTM dans l'analyse des séries chronologiques et ouvrent la voie à des recherches futures pour affiner davantage leur performance et leur applicabilité à différents domaines.

2.1 introduction:

Dans ce chapitre, nous présenterons les expérimentations menées pour évaluer les performances des modèles et analyser les résultats obtenus. Nous commencerons par décrire les outils et bibliothèques utilisés pour l'implémentation, en détaillant les choix technologiques qui ont soutenu le développement de notre modèle, notamment les environnements de programmation, les bibliothèques d'apprentissage automatique dans le but d'appliquer le modèle à notre zone d'étude.

2.2 Environnement de travail:

Plusieurs outils ont été développés et sont mis à la disposition de chercheurs et d'étudiants comme nous pour simplifier la mise en place et l'utilisation des architectures du deep learning. Dans le cadre de notre travail, nous avons fait usage de :

2.2.1 Le langage Python:

Python est un langage de programmation interprète, de haut niveau à usage général,[4] largement utilisé en science des données, facile d'utilisation et dont les fonctionnalités sont très étendues grâce à la richesse des librairies et modules pouvant être importés. Il est simple d'utilisation mais très puissant avec des fonctionnalités avancées, utilisé par plusieurs entreprises ainsi que des domaines de recherche partout dans le monde. Python est également doté d'une large gamme de librairies, y compris pour le traitement graphique, numérique, analyse des données, apprentissage automatique et profond



Figure 2.1: python

2.2.2 Environnement de développement:

Google Colaboratory (Colab): Google Colaboratory est une plateforme opensource basée sur les notebooks où vous pouvez écrire et exécuter des codes Python. Colab ne nécessite aucune configuration pour démarrer, ce qui en fait un choix[5] pratique pour les projets collaboratifs et l'expérimentation rapide



Figure 2.2: google colab

2.2.3 Les bibliothèques utilisées:

Pandas : Défini comme une bibliothèque open-source qui fournit des manipulations de données haute performance en Python. Développée par Wes McKinney, pandas est dérivée du mot "Panel Data" qui signifie une économétrie à partir de données multidimensionnelles. L'analyse des données nécessite de nombreux traitements,[6] tels que la restructuration, le nettoyage ou la fusion, etc. pandas effectuera ces traitements d'une manière rapide et efficace

TensorFlow : Développé par Google Brain Team le 9 novembre 2015, TensorFlow est un Framework (bibliothèque open-source gratuite) populaire de l'apprentissage automatique et de l'apprentissage profond. TensorFlow est basé entièrement sur le langage de programmation Python et est utilisé pour le calcul numérique et le flux de données, [7]ce qui rend l'apprentissage automatique plus rapide et plus facile. TensorFlow fonctionne sur plusieurs processeurs ou GPU et est utilisé principalement dans l'apprentissage et l'exécution des réseaux neuronaux profonds pour l'analyse des séries chronologiques, la classification d'image, le traitement du langage naturel, la détection de vidéos, etc.

Keras : Écrite en Python, Keras est une bibliothèque open-source de haut niveau pour les réseaux de neurones, capable de fonctionner avec des backends tels que Theano, TensorFlow ou CNTK (des bibliothèques Python). Elle a été développée par un ingénieur de Google, François Chollet. Keras est conviviale et extensible, visant à faciliter l'expérimentation avec les réseaux de neurones profonds. Elle ne supporte pas seulement les réseaux convolutifs et les réseaux de neurones récurrents individuellement, mais aussi leur combinaison[8]. De grandes entreprises telles que Microsoft, Google, NVIDIA et Amazon ont activement contribué au développement de Keras. Il a une bonne interaction avec l'industrie et est utilisé dans le développement d'entreprises populaires telles que Netflix, Uber, Google, Expedia, etc.

2.3 Conclusion:

Ce chapitre a permis de présenter les différentes expérimentations menées pour évaluer nos modèles de prédiction des précipitations maximales. En détaillant les outils et les bibliothèques utilisés, comme Python, Google Colaboratory, Pandas, TensorFlow et Keras, nous avons mis en lumière les technologies qui facilitent la mise en œuvre d'architectures de deep learning. L'importance des choix d'hyperparamètres pour l'optimisation des modèles a également été soulignée, en soulignant leur impact direct sur la performance des résultats. Ces éléments constituent une base solide pour l'application de nos modèles à la zone d'étude, en assurant une approche rigoureuse et adaptée à nos objectifs de recherche.

CHAPTER 3

APPLICATION DES RÉSEAUX LSTM POUR LA PRÉVISION DES PRÉCIPITATIONS DANS LA ZONE DE TEFFESSOUR

3.1 LA ZONE DE TEFFESSOUR :

une altitude variant entre 700 et 1400 mètres. Elle fait partie du grand bassin versant de Chatt Charguiet. À l'ouest, notre zone d'étude est bordée par l'oued Tawrira et entourée de montagnes. La zone est remplie de sédiments anciens, déposés par la rivière Mekerra.

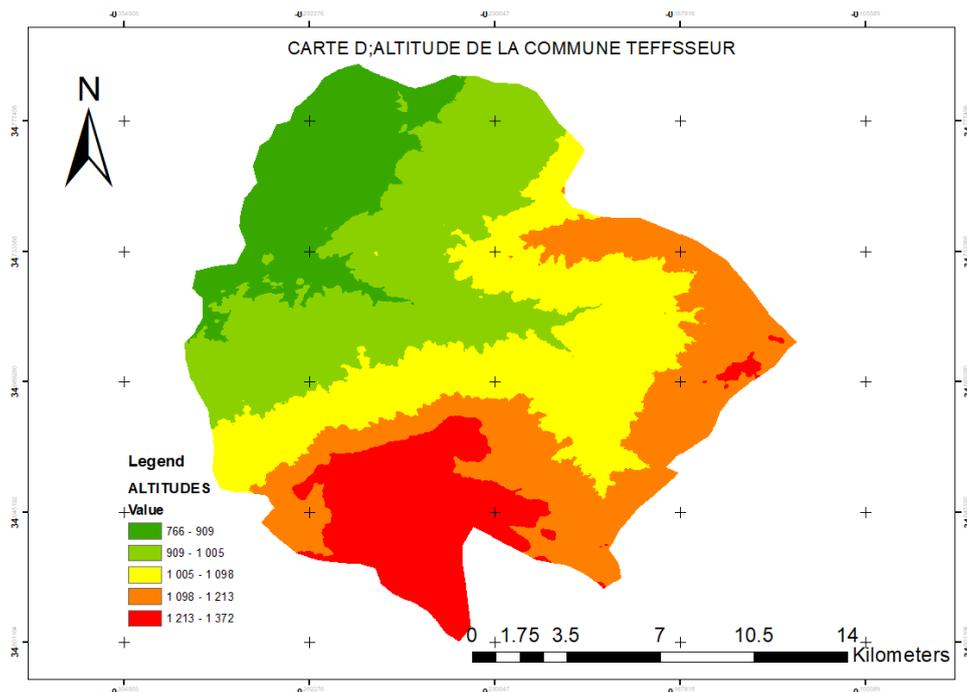


Figure 3.1: carte d'altitude de la commune de teffessour

3.2 Préparation des données:

on dispose d'une série de données comportant les totaux des précipitations mensuelles journaliers pour 46 années [15] d'observation. Les valeurs de cette série sont reportées dans ANNEX :

La station de	Code	X(km)	Y(km)	Altitude (m)	Période d'observation
MERINE	110501	211.6	169.5	970	1978-2020

Table 3.1: Informations sur la station MERINE

Les données ont été normalisées à l'aide de la méthode `StandardScaler` de la bibliothèque `sklearn`, où les données sont centrées autour d'une moyenne de 0 et d'un écart-type de 1, afin d'éviter tout effet de saturation pouvant être causé par l'utilisation des fonctions d'activation comme la sigmoïde. Après cette étape de normalisation, les données ont été divisées en six sous-ensembles pour une validation croisée spécifique aux séries temporelles, en respectant l'ordre chronologique dans chaque partition.

Le modèle LSTM utilisé est de type `Vanilla`, avec deux couches cachées. Le nombre de neurones (N) dans ces couches ainsi que la fonction d'activation ont été optimisés via une procédure d'essai-erreur pendant la phase de calibration. Chaque sous-ensemble (folder) a été entraîné 100 fois afin de garantir une meilleure généralisation du modèle. Cette approche permet de réduire les effets des variations aléatoires et de stabiliser les performances globales du modèle lors de l'évaluation sur les séries temporelles.

3.3 la visualisation et l'analyse de jeu de donnée:

Ce graphique montre la série complète des précipitations maximales en fonction du temps (1979-2020). On remarque plusieurs pics significatifs, notamment autour des années 1989, 1994, 2009 et 2014, où des précipitations maximales particulièrement élevées ont été enregistrées. La série montre beaucoup de variabilité, avec des périodes de précipitations beaucoup plus calmes suivies de pics marqués, suggérant un comportement irrégulier avec des événements extrêmes ponctuels. Les fluctuations semblent augmenter et diminuer périodiquement, ce qui pourrait indiquer une certaine cyclicité à long terme.

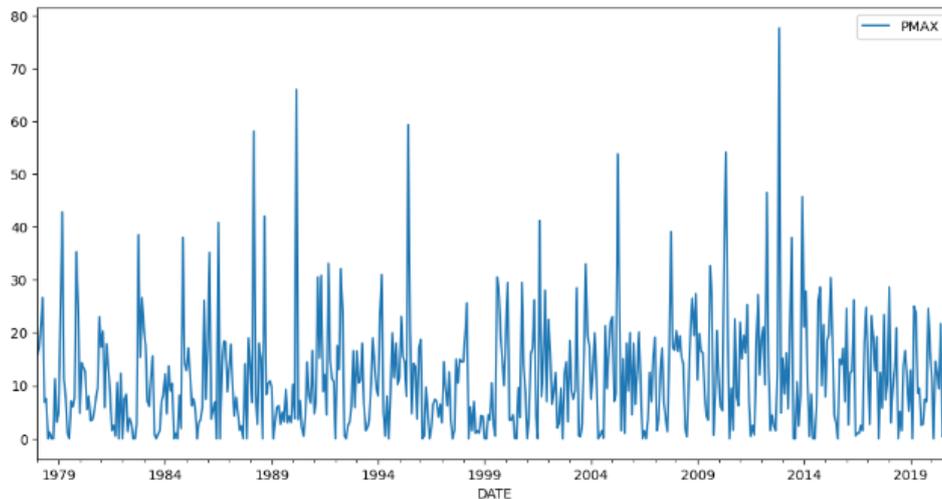


Figure 3.2: les précipitations maximales

Trend (Tendance) : Cette courbe montre la tendance générale de la série au fil du temps. On observe des fluctuations avec une tendance légèrement décroissante ou stagnante dans les dernières années.

Seasonal (Saisonnalité) : Ce graphique montre un motif répétitif régulier, ce qui signifie qu'il existe un comportement saisonnier dans les précipitations. Les vagues saisonnières semblent très régulières sur l'ensemble de la période, suggérant une forte composante saisonnière dans les précipitations.

Resid (Résidus) : Les résidus sont les valeurs qui restent après avoir enlevé la tendance et la composante saisonnière. Ici, les points montrent qu'il y a une certaine variabilité aléatoire, avec des valeurs résiduelles qui montrent que certains événements extrêmes ne sont pas capturés par la saisonnalité ou la tendance.

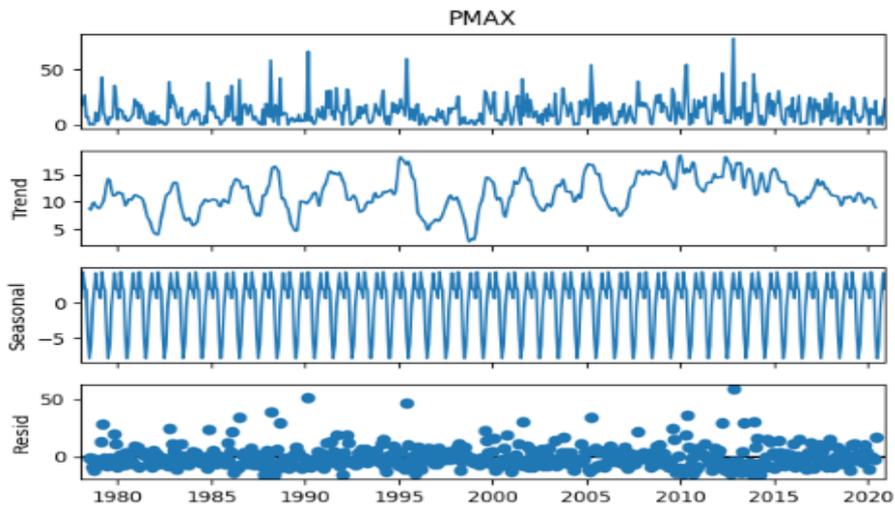


Figure 3.3: Décomposition de la série temporelle

3.4 Résultats et discussion:

3.4.1 architecture de modèle:

Le modèle LSTM utilisé pour la prédiction des précipitations est composé de deux couches LSTM successives, chacune accompagnée d'une couche de régularisation Dropout pour limiter le surapprentissage. La première couche LSTM contient 50 unités de mémoire et retourne une séquence à travers un vecteur de dimension 50, ce qui permet au modèle de capturer des relations temporelles complexes dans les données. La deuxième couche LSTM, également avec 50 unités, ne renvoie pas de séquence, mais uniquement le dernier état caché, ce qui synthétise l'information sur toute la séquence d'entrée. Les couches Dropout, intercalées après chaque couche LSTM, désactivent de manière aléatoire 30% des unités pour prévenir le surajustement. Enfin, une couche Dense entièrement connectée avec une seule unité de sortie est utilisée pour prédire la valeur des précipitations maximales. En tout, le modèle comporte 30,651 paramètres, tous optimisés durant l'entraînement, pour capturer au mieux les dynamiques des précipitations dans les séries temporelles.

Layer (type)	Output Shape	Param #
lstm_80 (LSTM)	(None, 1, 50)	10,400
dropout_80 (Dropout)	(None, 1, 50)	0
lstm_81 (LSTM)	(None, 50)	20,200
dropout_81 (Dropout)	(None, 50)	0
dense_40 (Dense)	(None, 1)	51

Total params: 30,651 (119.73 KB)
 Trainable params: 30,651 (119.73 KB)
 Non-trainable params: 0 (0.00 B)
 Epoch 1/10

Figure 3.4: L'architecture de modèle

3.5 Mesures de performance pour l'évaluation

Évaluer un modèle nous aide à savoir s'il fonctionne bien. C'est très important pour créer un modèle d'apprentissage automatique qui marche bien. Pour voir à quel point il marche, on utilise différentes mesures, selon le type de problème à résoudre.

3.5.1 Erreur quadratique moyenne (MSE)

L'erreur quadratique moyenne mesure la moyenne des carrés des écarts entre les valeurs réelles et les valeurs prédites. Plus la MSE est basse, meilleur est le modèle.[14]

Formule :

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_{\text{réel}} - y_{\text{prévu}})^2 \quad (3.5.1)$$

où :

- $y_{\text{réel}}$ est la valeur réelle,
- $y_{\text{prévu}}$ est la valeur prédite,
- n est le nombre de points de données.

3.5.2 Coefficient de détermination (R^2)

Le R^2 mesure la proportion de la variance dans les données qui est expliquée par le modèle. Un R^2 proche de 1 indique que le modèle explique bien la variabilité des données.[14]

Formule :

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_{\text{réel}} - y_{\text{prévu}})^2}{\sum_{i=1}^n (y_{\text{réel}} - \bar{y})^2} \quad (3.5.2)$$

où :

- \bar{y} est la moyenne des valeurs réelles.
- $y_{\text{réel}}$ est la valeur réelle,
- $y_{\text{prévu}}$ est la valeur prédite,
- n est le nombre de points de données.

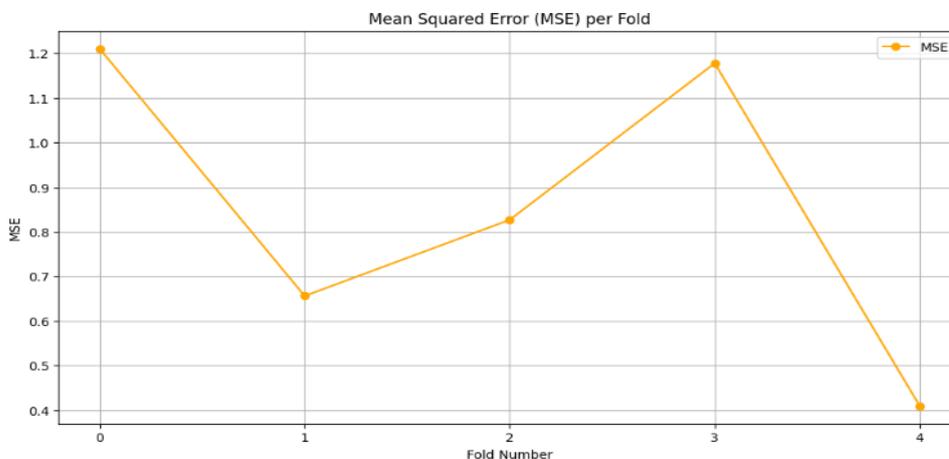


Figure 3.5: L'évolution de la MSE pour tous les folds.

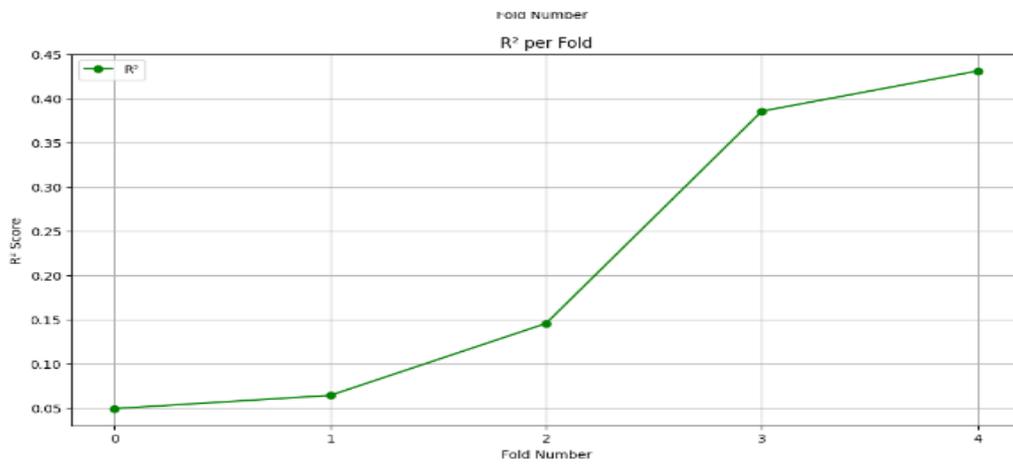


Figure 3.6: L'évolution de R^2 pour tous les folds.

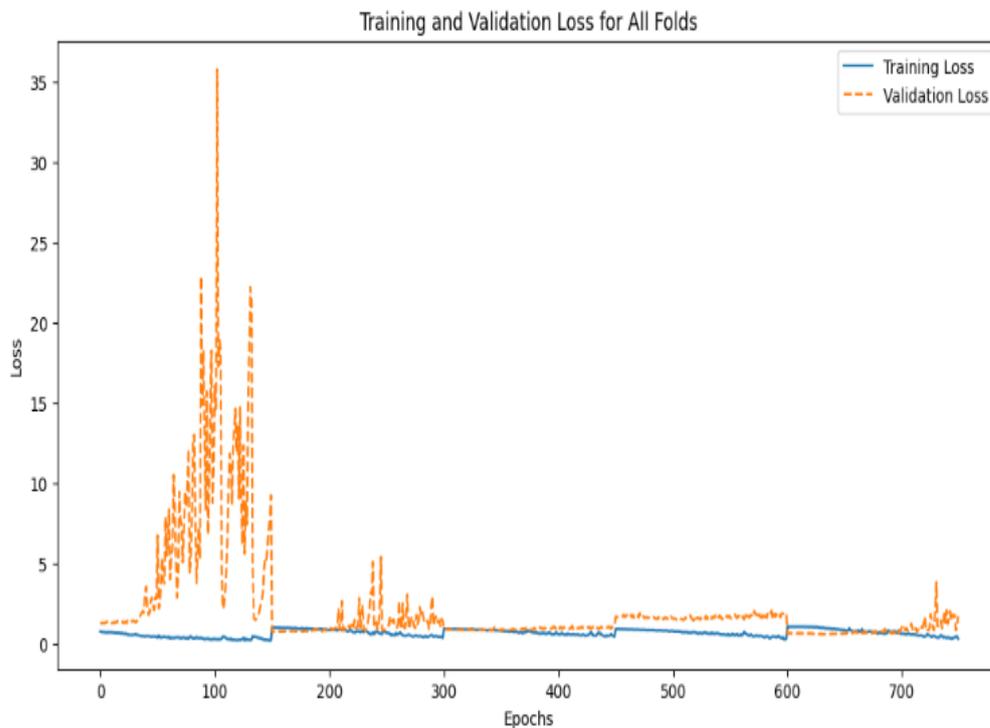


Figure 3.7: Perte d'entraînement et perte de validation au cours des FOLDS

3.5.3 les résultats des prédictions du modèle:

Le graphique ci-dessus montre les résultats des prédictions du modèle LSTM pour les précipitations maximales mensuelles (PMA) sur une période de 120 mois dans le futur, à partir des données historiques. La courbe en bleu représente les valeurs réelles observées des précipitations sur la période historique (allant de 1970 à environ 2020), tandis que la courbe en orange illustre les prédictions effectuées pour la période future, couvrant les années de 2020 à 2030.

On peut observer que les valeurs réelles des précipitations présentent une grande variabilité, avec des pics importants représentant des événements de précipitations maximales élevées

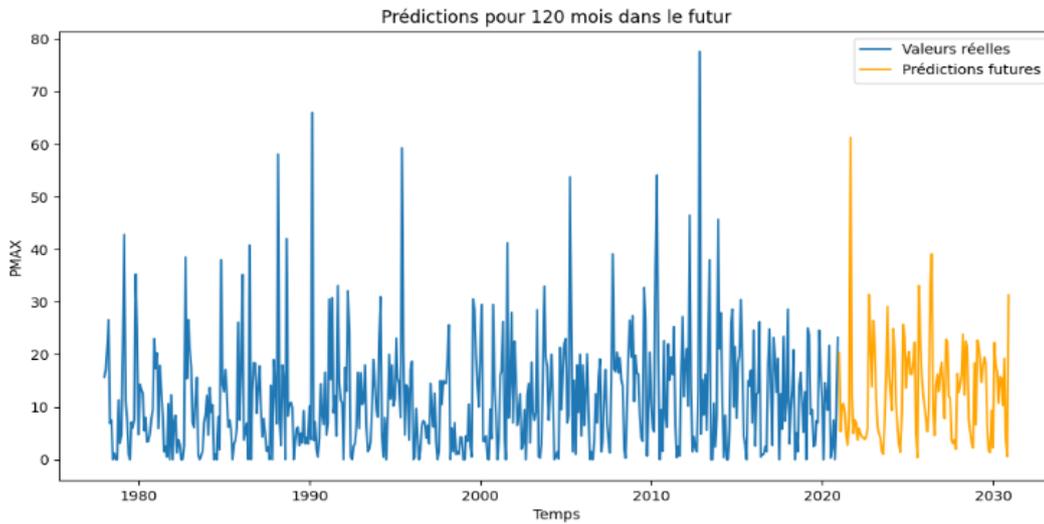


Figure 3.8: LA prédiction de 120 mois dans le futur

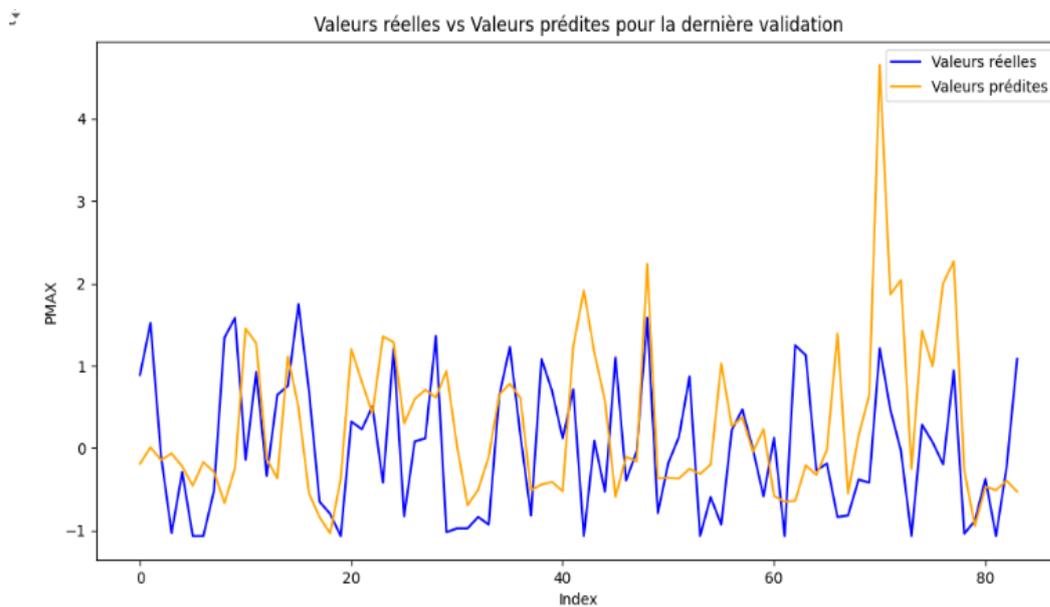


Figure 3.9: Comparaison entre les valeurs réelles des précipitations et les valeurs prédites

Conclusion

Cette étude démontre l'efficacité des modèles LSTM dans la prévision des précipitations dans la zone de Teffessour, tout en intégrant une validation croisée spécifique aux séries temporelles. Grâce à l'analyse approfondie des données historiques, incluant la normalisation et la décomposition des séries temporelles, nous avons réussi à capturer les tendances, la saisonnalité et les fluctuations aléatoires inhérentes aux précipitations.

L'architecture du modèle, composée de deux couches LSTM avec régularisation par Dropout, a permis de traiter efficacement les relations temporelles complexes tout en limitant le surapprentissage. L'utilisation de la validation croisée a été cruciale pour garantir la généralisation du modèle, en préservant l'ordre chronologique des données lors de la partition. Les mesures de performance, telles que l'erreur quadratique moyenne (MSE) et le coefficient de détermination (R^2), ont démontré que le modèle pouvait prédire avec précision les valeurs de précipitations maximales sur des périodes allant jusqu'à 120 mois dans le futur.

Les résultats des prédictions indiquent une variabilité significative des précipitations, avec des événements extrêmes bien capturés. Ces informations sont essentielles pour une meilleure gestion des ressources

en eau et une planification efficace dans la région. En somme, cette étude illustre l'importance des modèles LSTM et de la validation croisée pour améliorer la précision des prévisions dans le domaine hydrologique.

CONCLUSION GÉNÉRALE

Ce travail a examiné l'application des modèles LSTM (Long Short Term Memory) pour la prévision des précipitations sur une période de 120 mois, avec un accent particulier sur la zone de Teffessour. Dans la première partie, nous avons discuté des fondements théoriques des réseaux de neurones récurrents, en mettant en avant leur capacité à capturer les dépendances temporelles à long terme dans les séries chronologiques. L'analyse des travaux antérieurs a révélé que les modèles LSTM surpassent souvent les méthodes traditionnelles dans la prévision des précipitations, en raison de leur structure capable de gérer des séquences de données complexes.

Dans la seconde partie, nous avons détaillé notre méthodologie, qui inclut la préparation et le traitement des données de la station d'Ighil Emda, ainsi que l'utilisation des outils Python, Google Colaboratory, et des bibliothèques TensorFlow et Keras. L'optimisation des hyperparamètres a été une étape cruciale pour améliorer les performances de nos prévisions, permettant d'atteindre des résultats significatifs pour une période de prévision de 120 mois.

En conclusion, cette étude met en lumière l'efficacité des modèles LSTM pour la prévision des précipitations sur le long terme, offrant des perspectives prometteuses pour la gestion de réseau d'assainissement et l'adaptation aux changements climatiques. Les résultats obtenus soulignent l'importance de poursuivre les recherches dans ce domaine, afin de perfectionner les modèles et d'étendre leur application à d'autres contextes et climatiques.

- [1] Gridin, I. (2021). Time Series Forecasting using Deep Learning: Combining PyTorch, RNN, TCN, and Deep Neural Network Models to Provide Production-Ready Prediction Solutions (English Edition). BPB Publications..
- [2] Atto, A. M., Bovolo, F., Bruzzone, L. (2024). Détection de changements et analyse des séries temporelles d'images 2: Méthodes supervisées. ISTE Group.
- [3] Haykal, V. (2019). Modélisation des séries temporelles par apprentissage profond (Doctoral dissertation, Tours).
- [4] Python. <https://www.python.org>
- [5] Google Colab. <https://colab.research.google.com>
- [6] Pandas. <https://pandas.pydata.org>
- [7] TensorFlow. <https://www.javatpoint.com/tensorflow-introductio>
- [8] Keras. <https://www.javatpoint.com/keras>
- [9] Bergmeir, C., Benítez, J. M. (2012). On the use of cross-validation for time series predictor evaluation. Information Sciences, 191, 192-213.
- [10] Guyader, A. (2011). Régression linéaire. Université Rennes, 2, 60-61.
- [11] Farhadi, F. (2017). Learning activation functions in deep neural networks. Ecole Polytechnique, Montreal (Canada).
- [12] Sherstinsky, A. (2020). Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network. Physica D: Nonlinear Phenomena, 404, 132306.
- [13] ANRH

- [14] El Badaoui, H., Abdallaoui, A., Chabaa, S. (2013). Etude des effets des algorithmes d'apprentissage et des fonctions de transfert sur la performance des modèles statistiques neuronaux: Application dans le domaine météorologique. *International Journal of Engineering Research and Development*, 9(6), 15-26.
- [15] Gouriéroux, C., Monfort, A. (1995). *Séries temporelles et modèles dynamiques*. FeniXX.
- [16] COULIBALY, Paulin, ANCTIL, François, et BOBÉE, Bernard. Prévion hydrologique par réseaux de neurones artificiels: état de l'art. *Canadian Journal of civil engineering*, 1999, vol. 26, no 3, p. 293-304.
- [17] BELOUZ, Khaled. Modélisation de l'évapotranspiration de référence et du déficit hydrique par les réseaux de neurones artificiels à différent pas de temps. 2009. Thèse de doctorat. 2009.
- [18] HINTON, Geoffrey, et al. Réseau neuronal artificiel.
- [19] MA, Lina, BAJIC, Vladimir B., et ZHANG, Zhang. On the classification of long non-coding RNAs. *RNA biology*, 2013, vol. 10, no 6, p. 924-933.
- [20] NAUCK, Detlef et KRUSE, Rudolf. Neuro-fuzzy systems. In : *Handbook of Fuzzy Computation*. CRC Press, 2020. p. 319-D2. 10: 2.
- [21] KOFFI, Yao Blaise. Etude du calage, de la validation et des performances des réseaux de neurones formels à partir des données hydro-climatiques du bassin versant du Bandama blanc en Côte d'Ivoire. Abidjan,(Côte d'Ivoire), Mémoire Online, 2007, p. P60.

Annexes

Précipitations Mensuelles et Maximum Journalier (1978-2020)													
Année	Sept	Oct	Nov	Déc	Janv	Fév	Mars	Avr	Mai	Juin	Juil	Août	Pmaxj
1978	0	0	11.3	3.1	15.7	17.2	21.2	26.6	6.9	7.5	0	1.2	26.6
1979	6	7.7	35.3	24.8	4.8	22.6	42.8	11.2	7.7	1	0	7.1	42.8
1980	5	7.7	9.5	23	4.8	14.3	13.4	12.6	5.5	8	3.4	3.5	23
1981	0.5	10.6	0	12.3	17.3	20.3	5.9	17.9	13.5	8.5	1.5	2.5	20.3
1982	2.7	38.5	15.4	26.6	0	7.4	8.4	1.3	3.8	2.9	0	0	38.5
1983	0.8	1.5	7	8.4	21	17.6	7	6.1	11.7	15.6	0.8	0	21
1984	8.2	1.9	38	14.2	12.2	4.7	13.7	9	10.4	0	1	0	38
1985	3.6	6	26.1	7.5	12.9	17.1	12.6	6.2	7.5	5.6	0	3	26.1
1986	13.5	18.4	18.3	8.8	17.3	35.2	3.7	5.3	6.9	0	40.8	0	40.8
1987	0	14.1	0	19	13.5	17.8	9.8	4.3	7.8	4.9	1.6	2.4	19
1988	42	8.3	10.4	10.9	13.5	6.8	58.1	8.3	2.7	18	15.1	0	58.1
1989	9.3	3	4.2	3	9.8	0	3.2	5.9	6.2	2.7	5	3.2	9.8
1990	14.4	8.3	6.7	16.6	10.2	3.8	66	3.6	7.2	2	0.5	4.1	66
1991	33.1	14.7	11.3	10.8	4.7	7	30.5	15.2	30.8	8.9	12.1	4.5	33.1
1992	3.1	6.4	16.6	5.9	0	17.5	13	32.1	24.1	0.5	0	2.5	32.1
1993	10.3	19	14	9.5	16.5	10.5	11	18	6.5	1.5	2	3.5	19
1994	20	11.5	18	10.2	8.1	22.9	31	4.8	0.5	8	0	8.5	31
1995	14.2	13.6	3.7	17	11.6	23.1	15.5	14.4	8	59.3	20.4	4.7	59.3
1996	7.4	7	4.1	6.5	18.7	0	0.5	9.7	5	0	2	6.5	18.7
1997	15	10.5	15	14.5	3	14.5	9.2	6.2	12.6	3.4	0	1.5	15
1998	1.5	1	4.3	4.2	14.5	19.5	25.6	0	6	1.5	7	1	25.6
1999	28.5	19.7	14.5	10	0	0	4.5	3.5	10.5	2.5	0.5	30.5	30.5
2000	4	29.5	13.8	8.5	20.5	29.5	3.5	3.5	4.5	0.1	0	9.6	29.5
2001	7.9	14.3	28	7	0	4	16	17	26.2	4	0	41.2	41.2
2002	0	12.2	14.5	3.2	22.5	16	6.5	9	12.5	2	3	9.5	22.5
2003	16.5	33	19.5	17.6	18.5	9	7.5	9	28.5	0.6	0.3	2.9	33
2004	21.3	9.5	17	22.1	7.5	12.5	20	12	0	0.5	1.5	0.1	22.1
2005	18	9	20	4.5	23	7	8.5	53.8	24.5	1.5	15.1	1	53.8
2006	3.5	12.5	7	15.5	18	6.5	12.5	20.1	9.5	0	1.5	0	20.1
2007	15.5	39.1	17.2	16.7	19.1	1.5	3.5	12.4	17.1	6.9	3.9	1.3	39.1
2008	20.6	26.5	19.5	27.4	20.4	16.5	19.4	15.3	14	2	0.3	10.2	27.4
2009	27.2	0.7	7.8	20.4	11.1	19.8	16.5	16.2	8.2	4.2	3.5	32.7	32.7
2010	1.6	22.6	8	6.2	10.8	5.9	5.3	36.8	54.1	19.1	0	9.5	54.1
2011	0.7	16.5	27.2	12	22	15.1	19.5	16.2	25.3	6.6	0.4	2.5	27.2
2012	1.5	13.4	77.6	4.9	18.7	21.1	10.2	46.5	13.6	1.5	4.4	2.3	77.6
2013	10.7	2.4	8.3	45.7	15.2	8.5	16.2	5.6	20.5	38	0	0	45.7
2014	26	28.6	10	21.5	21.1	27.9	10.5	0.4	8.4	0	0	5.9	28.6
2015	15	14	17	7	7.9	18.5	19.7	30.4	19	4.5	2.9	0	30.4
2016	2.5	1.5	18.4	24.8	24.6	2.6	12.4	12.8	26.2	0.5	1	1	26.2
2017	5.8	23.4	7.3	11.1	13.2	2.7	23.2	19	12.8	19.2	0	12.5	23.4
2018	13.9	16.6	11.4	5.2	28.6	3	9.6	13	20.9	0	5.1	1.5	28.6
2019	7.4	7	24.6	16.6	12.9	0	25	23.7	8.6	9.5	2.5	2.7	25
2020	7.5	0	9	23.2	11.2	0	14.6	12.3	9.4	21.7	0.3	2	23.2

Table 3.2: serie des precipitations