

RÉPUBLIQUE DU CAMEROUN  
Paix-Travail-Patrie  
.....  
MINISTÈRE DES POSTES ET  
TÉLÉCOMMUNICATIONS  
.....  
ÉCOLE NATIONALE SUPÉRIEURE DES  
POSTES, DES TÉLÉCOMMUNICATIONS,  
DES TECHNOLOGIES DE L'INFORMATION  
ET DE LA COMMUNICATION  
.....

REPUBLIC OF CAMEROON  
Peace-Work-Fatherland  
.....  
MINISTRY OF POSTS AND  
TELECOMMUNICATIONS  
.....  
NATIONAL ADVANCED SCHOOL OF  
POSTS, TELECOMMUNICATIONS,  
INFORMATION AND COMMUNICATION  
TECHNOLOGIES  
.....



## **MÉMOIRE DE FIN D'ÉTUDES**

### **DÉVELOPPEMENT D'UNE MÉTHODE DE PRÉDICTION MÉTÉOROLOGIQUE EN ZONE TROPICALE À L'AIDE DU MACHINE LEARNING : CAS DE YAOUNDÉ**

En vue de l'obtention du diplôme  
**D'INGÉNIEUR DE CONCEPTION DES TÉLÉCOMMUNICATIONS**

Présenté par :

**NDOUMIN SAMBA Serge**

Matricule : **15T31062**

Option : **Services Réseaux et Radio Mobiles (SRM)**

Devant un jury composé de :

**Président:** **M. Jean-Robert KALA KAMDJOUG, MC/UCAC**

**Rapporteur:** **M. TCHAGNA KOUANOU Aurelle, Dr-Ing/**

**Examinateur:** **M. Valery MONTHE, CC/UY1**

**Invité:** **M. HEUGANG N. A. Steven, Ing/INCHTECH'S Yaoundé**

**Année académique : 2019-2020**

## DÉDICACES

A mon père et à ma mère

M. NDOUMIN Maurice

Mme. EBAH ABESSOLO R. Georgette

## REMERCIEMENTS

Parvenu au terme de ce travail, c'est l'occasion d'adresser mes profonds et sincères remerciements à des personnes qui, d'un bout à l'autre et de quelque manière qu'il s'agisse, ont contribué à son élaboration.

Je tiens à remercier **M. WATCHING Félix**, Directeur de l'École Nationale Supérieure des Postes, des Télécommunications et des TICS ; mon encadreur académique **Dr-Ing. TCHAGNA KOUANOU Aurelle** pour sa disponibilité, ses encouragements et les conseils avisés dont il m'a fait profiter. Mes remerciements vont aussi à l'endroit de mon encadreur professionnel **Ing. HEUGANG NDJANDA Steven A.** pour sa disponibilité, sa pertinence, sa rigueur et ses conseils. Je tiens aussi à remercier **Messieurs les Membres du Jury** pour l'intérêt qu'ils ont manifesté à l'égard de ce travail en acceptant de l'évaluer, ainsi que tous les enseignants du cycle ingénieur des télécommunications pour la formation qu'ils nous ont donnée.

J'adresse également mes remerciements au **Directeur de INCHTECH'S SARL**, pour m'avoir permis de toucher du doigt le monde professionnel, au **Directeur de la Météorologie du Ministère des Transports** pour son accueil chaleureux dans sa Direction. J'exprime aussi ma gratitude au **CEO de ROBOTSAVE, Ing. DESSAP FOMEKONG Loïc Joffree** de m'avoir accompagné avec ses équipements ainsi qu'aux Ingénieurs **BAYANGMBE MOUNMO** et **MUHOZAM Yves Christian** pour leur écoute, leur aide et leurs conseils très avisés.

Je remercie très chaleureusement mes parents **NDOUMIN Maurice** et **EBAH ABESSOLO Rufine Georgette** sans qui rien de tout ceci n'aurait été possible : pour tous les sacrifices que vous avez faits je vous présente par ces quelques mots mon profond respect et mon amour ; mes sœurs **NSIZOA Léa Catherine**, **NDOUMIN NDOKONG Estelle**, **NDOUMIN SEN Patricia** ; mes neveux.

J'adresse un merci particulier à **NKODO Liliane Josiane Marina** pour son écoute, ses encouragements et à mes amis **KARAGAMA Aicha Myriam**, **MOULIOM BOUREIDAH**, **DJUIKOM FOKA Gwladys Ornella**, **TEDONGMOUO MAKAMTA Sandra**, **NINGAHI NDUNGLA Chancel Joël**, **LISSOM Wilfried Albert**, **DAMA YERO Valdès**, qui n'ont jamais cessé de m'encourager durant ce travail.

---

# TABLE DES MATIÈRES

DÉDICACES .....	i
REMERCIEMENTS .....	ii
TABLE DES MATIÈRES .....	iii
LISTE DES FIGURES .....	vi
LISTES DES TABLEAUX.....	viii
LISTES DES ACRONYMES .....	ix
RÉSUMÉ.....	x
ABSTRACT .....	xi
INTRODUCTION GÉNÉRALE.....	1
1. Contexte .....	2
2. Problématique.....	4
3. Objectifs .....	5
4. Plan du travail.....	5
Chapitre 1 : REVUE DE LA LITTÉRATURE .....	7
Introduction .....	8
1.1. Méthodes de prédiction météorologique .....	8
1.1.1. Méthodes traditionnelles .....	8
1.1.2. Méthode numérique.....	9
1.2. Machine Learning (Apprentissage Automatique) .....	10
1.2.1. Généralités.....	10
1.2.2. Etat de l'art de la prédiction météorologique à l'aide du Machine Learning.....	16
Conclusion.....	19
Chapitre 2 : MÉTHODOLOGIE .....	20
Introduction .....	21

## TABLE DES MATIÈRES

---

2.1.	Présentation de la solution.....	21
2.2.	Pipeline de la solution.....	21
2.2.1.	Données : collecte et exploration .....	22
2.2.2.	Prétraitement des données .....	23
2.3.	Modèle mathématique du modèle.....	26
2.3.1.	Réseau de neurones par récurrence (RNN) .....	27
2.3.2.	Long Short Term Memory (LSTM) .....	27
2.3.3.	Fonction d'optimisation .....	31
2.3.4.	Métriques d'évaluation des performances.....	31
2.4.	Outils d'implémentation .....	32
2.4.1.	Matériels physiques.....	32
2.4.2.	Architecture .....	36
2.4.3.	Environnement logiciel .....	36
	Conclusion.....	38
	<b>Chapitre 3 : RÉSULTATS ET INTERPRÉTATION .....</b>	<b>39</b>
	Introduction .....	40
3.1.	Collecte et sauvegarde de données .....	40
3.1.1.	Rappel de l'objectif .....	40
3.1.2.	Présentation du système de collecte et sauvegarde de données .....	40
3.1.3.	Discussion .....	43
3.2.	Visualisation des données.....	43
3.2.1.	Rappel de l'objectif .....	43
3.2.2.	Présentation de la plateforme de visualisation .....	44
3.1.	Modèle de prédiction .....	51
3.1.1.	Rappel de l'objectif .....	51
3.1.2.	Présentation du modèle de prédiction .....	51
3.1.3.	Discussion .....	54

## TABLE DES MATIÈRES

---

3.2. Estimation de la solution .....	59
Conclusion.....	60
CONCLUSION GÉNÉRALE .....	61
ANNEXES .....	xii
RÉFÉRENCES .....	xv

# LISTE DES FIGURES

Figure 1.1 : Approche classique (Géron, 2017) .....	10
Figure 1.2 : Approche orientée Machine Learning (Géron, 2017).....	11
Figure 1.3 : K-plus proche voisins (Sanjay, 2018).....	12
Figure 1.4 : Régression linéaire (Roman 2020) .....	13
Figure 1.5 : Régression logistique, classification binaire (Younes, 2017).....	13
Figure 1.6 : Support Vector Machine (Sweet, 2020) .....	14
Figure 1.7 : Arbre à décision.....	14
Figure 1.8 : Random forest (Abhishek, 2020).....	15
Figure 1.9 : Réseau de neurones (Bastien,2019) .....	16
Figure 2.1 : Algorithme de travail .....	22
Figure 2.2 : Visualisation des données .....	24
Figure 2.3 : Corrélation entre les données.....	24
Figure 2.4 : Structure du RNN .....	27
Figure 2.5 : Structure d'un bloc LSTM basique (Hewamalage et al. 2020).....	28
Figure 2.6 : Raspberry pi 4 B .....	32
Figure 2.7 : Arduino Uno .....	33
Figure 2.8 : Capteur DHT11 .....	34
Figure 2.9 : Anémomètre - capteur de la vitesse du vent avec RJ11 - pour station météorologique - N25FR WH1080 .....	34
Figure 2.10 : Pluviomètre .....	35
Figure 2.11 : Mini Station météo .....	35
Figure 2.12 : Architecture de notre solution .....	36
Figure 3.1 : Session d'authentification dans la station de base.....	41
Figure 3.2 : Critère d'accès à la base de données influx db.....	41
Figure 3.3 : Login du tableau de bord .....	44
Figure 3.4 : Tableau de visualisation de données mesurées en temps réel.....	45
Figure 3.5 : Graphe d'évolution de la température .....	45
Figure 3.6 : Requête pour interroger la base de données par rapport à la température actuelle .....	46
Figure 3.7: Température moyenne prédictive arrondie par défaut .....	46

---

## LISTE DES FIGURES

---

Figure 3.8 : Température minimale quotidienne de chapelle Nsimeyong .....	47
Figure 3.9 : Température maximale journalière de Chapelle Nsimeyong .....	48
Figure 3.10 : Vitesse du vent de chapelle Nsimeyong temps réel.....	48
Figure 3.11 : Direction du vent de chapelle Nsimeyong en temps réel .....	49
Figure 3.12 : Widget de l'humidité relative.....	49
Figure 3.13 : Humidité relative de la journée du 23 novembre 2020 à chapelle Nsimeyong ..	50
Figure 3.14 : Widget du niveau de précipitation .....	50
Figure 3.15 : Niveau de précipitation.....	51
Figure 3.16: Vue d'ensemble du modèle .....	51
Figure 3.17 : Résumé des paramètres du modèle de prédiction .....	52
Figure 3.18 : Perceptron multicouche .....	53
Figure 3.19 : Pertes d'apprentissage et de validation de notre modèle (01) jour .....	53
Figure 3.20 : Courbes évolutives RMSE et MAE prédiction (01) jour.....	54
Figure 3.21 : Graphe de comparaison des performances de modèles .....	55
Figure 3.22 : Comparaison des prédictions quotidiennes .....	56
Figure 3.23 : Pertes d'entraînement et de validation des prédictions du 10 <sup>ème</sup> jour.....	58
Figure 3.24 : (05) prédictions effectuées avec les prédictions à 10 jours .....	59

## LISTES DES TABLEAUX

Tableau 1.1 : Résumé des travaux antérieurs .....	19
Tableau 2.1 : Format des données fournies par le centre national américain pour la prédition environnementale .....	22
Tableau 2.2 : Données restructurées sur Jupyter Notebook .....	23
Tableau 2.3 : Tableau résultant du nettoyage des données .....	25
Tableau 2.4 : Normalisation des champs .....	26
Tableau 2.5 : Tableau de données normalisées .....	26
Tableau 2.6 : Paramètre d'entraînement .....	31
Tableau 2.7 : Structure de notre base de données .....	37
Tableau 3.1 : Données collectées par notre mini-station météo.....	42
Tableau 3.2 : Statistiques des données collectées du Novembre 2020 par notre mini-station météo .....	42
Tableau 3.3 : Tableau de comparaison des données avec celles du Centre national américain pour les prévisions météorologiques .....	43
Tableau 3.4 : Comparaison des modèles de prédition à court terme (01) jour.....	55
Tableau 3.5 : Comparaison entre notre solution et AccuWeather.....	57
Tableau 3.6 : Evaluation des performances du modèle aux intervalles de 05,10 et 20 jours... <td>57</td>	57
Tableau 3.7 : Estimation du prix de la solution.....	59

## LISTES DES ACRONYMES

**ARMA:** Auto Regression Moving Average

**CPU:** Central Processing Unit

**DeepESN:** Deep Echo State Network

**ECMWF:** European Centre Medium-Range for Weather Forecast

**HF:** High Frequency

**Km/h:** Kilomètre par heure

**KNN:** K-plus proches voisins

**LF:** Low Frequency

**LSTM:** Long Short Term Memory

**MAE:** Mean Absolute Error

**MEMS:** MicroelectroMechanical Systems

**MF:** Medium Frequency

**MINTRANS:** Ministère des Transports

**mm:** millimeter

**MSE:** Mean Square Error

**NCEP:** National Centers for Environmental Prediction

**OMM:** Organisation Météorologique Mondiale

**ONACC:** Observatoire Nationale sur les Changements Climatiques

**RAM:** Random Access Memory

**RNN:** Recurrent Neural Network

**RSME:** Root Mean Square Error

**SGD:** Stochastic Gradient Descent

**STL:** Seasonal and Trend

**SVM:** Support Vector Machine

**TPU:** TensorFlow Processing Unit

**UHF:** Ultra High Frequency

**UTC:** Universal Time Coordinated

**V:** Volt

**VHF :** Very High Frequency

## RÉSUMÉ

La variabilité du climat dans le temps a une influence considérable sur différents métiers du monde. Il est à la charge de la météorologie de fournir les informations climatiques utiles (température, humidité, niveau de précipitation, pression atmosphérique, vitesse et direction du vent, etc.) en fonction du secteur d'activités. Afin de fournir ces informations, le Ministère des Transports utilise son réseau national d'observations et la méthode de prédition dite numérique. Dans ce projet, nous avons constitué une mini-station météorologique capable de mesurer la température, l'humidité, le niveau de précipitation, la vitesse et la direction du vent, ainsi que de visualiser les données prélevées en temps réel. Le Réseau de Neurone Récurrent (RNN) avec son algorithme ‘Long Short Term Memory’ (LSTM) a été utilisé pour concevoir notre modèle de prédition de la température. Ce modèle nous permet de prédire la température moyenne du jour suivant en fonction des données des 07 précédents jours avec un ‘Root Mean Square Error’ (RMSE) égal à **0,0188**. Ceci montre l'efficacité de notre modèle, qui a été comparé à ceux utilisés par l'Observatoire Nationale sur les Changements Climatiques (ONACC) en termes de prédition.

**Mots clés :** Climat tropical, Machine Learning, LSTM, Système embarqué

# ABSTRACT

The variation of climate over time has a great influence on different occupations around the world. It is the responsibility of meteorology to provide useful climatic information (temperature, humidity, precipitation level, atmospheric pressure, wind speed and direction, etc.) depending on the sector of activity. In order to provide this information, the Ministry of Transport uses its national observation network and the so-called numerical prediction method. In this project, we put in place a mini weather station capable of measuring temperature, humidity, precipitation level, wind speed and direction, as well as visualizing the data collected in real time. The Recurrence Neural Network (RNN) with its Long Short Term Memory (LSTM) algorithm was used to design our temperature prediction model. This model permitted us to predict the next day's average temperature based on the data of the previous 7 days with a Root Mean Square Error (RMSE) equal to 0.0188. This shows the efficiency of our model, which was compared to those used by the National Observatory on Climate Change (ONACC) in terms of prediction.

**Keywords:** Tropical climate, Machine Learning, LSTM, embedded system

# INTRODUCTION GÉNÉRALE

## 1. Contexte

La quasi-totalité des métiers du monde sont météo-sensibles autrement dit sont sensibles au temps, au climat et à l'eau. Le domaine d'étude en relation avec ces éléments est appelé météorologie. Selon l'organisation météorologique mondiale (OMM) la météorologie est l'étude de l'atmosphère et de ses phénomènes (METEOTERM OMM). Son enjeu majeur est la fourniture et l'application en temps utile d'informations, de prédictions et d'avis météorologiques précis qui contribuent au bien-être de l'humanité et qui sont extrêmement utiles pour le développement socio-économique et la protection de l'environnement.

Dans la parution OMM-N°1126 en 2017, la météo est aux services des domaines comme le transport dans le but de fournir en temps utile des informations précises sur les conditions météorologiques dans les airs, en mer et sur terre (OMM, 2017). En effet, dans les airs, les prédictions météorologiques permettent de réduire les décalages et la durée des vols, la consommation du carburant, tout ceci contribue à assurer la sécurité et le confort des passagers (OMM, 2017). Pour se faire, les avis météorologiques sur le vent fort, les orages, les turbulences et le givrage sont particulièrement préconisés. En mer, les prédictions météorologiques prévoient les conditions climatiques sur la trajectoire, le lieu et l'intensité des tempêtes fortes. Ces informations peuvent également être utilisées pour enquêter sur l'état de la mer lors des catastrophes en mer, des accidents maritimes ou encore lors des déversements accidentels des hydrocarbures (OMM, 2017). Concernant les airs, les phénomènes de brouillard, de fortes précipitations, de fort vent et de chasse-neige élevé sont des phénomènes qui perturbent énormément le trafic terrestre et ferroviaire.

La propagation des ondes radio (VHF, LF, MF, HF et UHF) est fortement perturbée par l'atmosphère et ses phénomènes (la pluie, le vent fort, les couches atmosphériques). Dans la planification d'un réseau de communication sans fil, il est nécessaire de déterminer les pertes du chemin et le rayon de couverture du signal. Une atténuation est causée par le brouillard dans les bandes de l'infrarouge et optique sans négliger les ravages causés par la pluie sur les ondes millimétriques beaucoup utilisées pour accroître le débit et la bande passante (Brown et al, 1984). De façon générale la propagation des ondes courtes (1 – 300 GHz) est influencée par les phénomènes de la couche basse de l'atmosphère (pluie, brouillard et vapeur d'eau) et crée ainsi une dispersion et une absorption du signal (Kestwal et al, 2014). Parallèlement, le génie civil est aussi conditionné par les éléments climatiques (la pluie, le brouillard, les vents forts, la

## INTRODUCTION GÉNÉRALE

---

neige, tempête, etc.) causant le ralentissement de nombreux chantiers tels que la construction de station de base et la pose de la fibre optique.

Cependant l'agriculture et la santé, secteurs clés du Cameroun, ne sont pas à négliger face à la variabilité des conditions météorologiques. L'agriculture dépend essentiellement du temps et du climat ; pour améliorer les cultures et accroître la production, il est plus que recommandé de maîtriser le climat des saisons de la localité. En effet, des facteurs environnementaux influent sur l'apparition des insectes ravageurs, des maladies des plantes et des animaux. L'exemple palpable est celui de l'envahissement des insectes qui ravagent les cultures, impactant significativement sur les rendements agricoles par des faibles récoltes (OMM, 2017). Les informations climatiques apportées à l'agriculture vont permettre d'anticiper sur les périodes de sécheresse et de prendre des dispositions conséquentes pour éviter ce genre de situation.

Le temps et le climat jouent un rôle capital dans la propagation des maladies infectieuses potentiellement dangereuses comme la grippe aviaire, qui sont véhiculées par les gouttelettes et les particules en suspension dans l'air (OMM, 2017). Le temps chaud et humide favorise le développement des moustiques, vecteurs de plusieurs maladies mortelles telles que le paludisme et la fièvre jaune. Le temps froid quant à lui rend particulièrement vulnérables les personnes âgées, les personnes handicapées et les nourrissons. Les systèmes d'alerte d'urgence pour ces dangers sont à établir par les spécialistes du temps et du climat en collaboration avec les services sanitaires et sociaux.

Quittant du cadre des métiers de la météorologie dans le monde en général et au Cameroun en particulier, il est important de dresser une caricature de l'acteur de la météorologie au Cameroun. En effet, selon le rapport de l'Observatoire National sur les Changements Climatiques (ONACC, 2016), nous relevons la responsabilité de la Direction Météorologique Nationale en matière de collecte de données d'observations météorologiques, de modélisation climatique et de modélisation de prévision numérique au travers en partie d'un réseau national centralisé de stations météorologiques. On dénombre à ce titre 41 stations fonctionnelles sur les 58 officiellement créées, dont 03 stations conformes aux normes recommandées par l'OMM et 38 en cours de réhabilitation progressive (ONACC, 2016). Notons ici que les données satellitaires sont utilisées pour renforcer ce réseau d'observation à l'échelle du haut de l'atmosphérique.

## INTRODUCTION GÉNÉRALE

---

L'état actuel du système d'observation au Cameroun est tel que nous avons les données météorologiques quotidiennement pour les villes à l'instar de Yaoundé. Mais, tenant compte des résultats d'une descente effectuée dans les arrondissements de la ville de Yaoundé, aucune station météorologique n'est actuellement déployée afin de collecter, de visualiser et de prédire la température moyenne ou même encore de corriger cette valeur en cas d'absence de mesure. Il est nécessaire de se pencher sur les procédés de collecte et de prédition de données météorologiques.

## 2. Problématique

La prédition météorologique se fait soit selon l'approche déterministe, soit selon l'approche probabiliste. La plus recommandée dans la littérature est celle probabiliste, utilisée pour prédire qu'un évènement spécifique va se réaliser à un instant T dans une région donnée. A cette approche est associée plusieurs méthodes de prédictions revues par Iseh en 2013 (Iseh et al, 2013) :

- a) La méthode de la permanence : elle se base principalement sur la continuité du jour présent ;
- b) La méthode climatologique : elle est fondée sur les anciennes conditions climatiques à la même période ;
- c) La méthode par analogie : elle recherche une correspondance entre les paramètres actuels et ceux d'un jour auparavant ;
- d) La méthode numérique : elle repose sur la résolution approximative de l'équation primitive dont suivent les gaz dans l'atmosphère ;
- e) La méthode par machine Learning : elle consiste à appliquer les algorithmes de machine Learning afin de trouver une relation entre les observations passées et prédire ensuite les paramètres climatiques.

On peut classer les méthodes (a-c) parmi les méthodes traditionnelles basées sur les statistiques et la continuité du temps et du climat. La méthode (d) est la plus utilisée notamment dans les grands centres d'observations comme European Centre Medium-Range for Weather Forecast (ECMWF). Enfin la méthode par le Machine Learning est présentée dans la littérature comme l'alternative de la technique de prédition numérique (Balamurugan et al 2019).

Au Cameroun, la méthode adoptée par le Ministère des Transports (MinTrans) est la méthode dite numérique. D'après Bauer en 2015, la précision de ce modèle dépend des observations en surface et des algorithmes d'assimilation (Bauer, 2015). Appliquer ce procédé

## INTRODUCTION GÉNÉRALE

---

au niveau des arrondissements de la ville de Yaoundé est un véritable challenge d'ordre technologique. Aujourd'hui, les ordinateurs de calculs de prédition numérique météorologique doivent être dotés de grosses performances. De plus, le réseau national ne permet pas d'être autant précis dans la ville de Yaoundé que dans ses arrondissements. Quant à la méthode par Machine Learning, elle tire parti de ses méthodes d'optimisation et réduit la consommation en ressource de calculs.

En tenant compte de ces paramètres, comment prédire la température moyenne dans un arrondissement de la ville de Yaoundé tout en appliquant la méthode par Machine Learning ?

### 3. Objectifs

Ce travail vise principalement à résoudre le problème du manque d'outils d'observation climatique précis au niveau d'un arrondissement de Yaoundé et va ainsi se décliner en trois (03) objectifs spécifiques :

- Collecter et sauvegarder les paramètres climatiques (température, humidité, vitesse du vent et niveau de précipitations) grâce à une mini-station et un Raspberry pi 4;
- Utiliser les outils du langage python pour le prétraitement et la visualisation des données ;
- Construire et optimiser un modèle de Machine Learning pour la prédition d'un jour de la température moyenne.

### 4. Plan du travail

Notre travail sera organisé en trois (03) chapitres :

- Dans le premier chapitre il sera question de faire la revue de littérature sur les techniques de mesures météorologiques ;
- Au deuxième chapitre nous présenterons les matériels et les méthodes que nous utiliserons pour la mise en œuvre de notre solution : le Raspberry pi, Arduino uno et les différents capteurs. Il s'agira aussi de présenter l'implémentation de la méthode de Machine Learning ainsi que le jeu de données ;
- Le troisième chapitre consistera à implémenter cette solution et à discuter des résultats qui en découlent. Notre travail se terminera par une conclusion et des perspectives à venir.

## **Chapitre 1 :**

# **REVUE DE LA LITTÉRATURE**

## Introduction

La ville de Yaoundé se caractérise par un climat tropical savane selon la classification de Köppen-Geiger (Tsalefac et al, 2015). Sa particularité est d'avoir une température moyenne annuelle supérieure à 18 °C (64,4 °F), ce qui peut influencer dans le calendrier de culture et favoriser la propagation des maladies comme le paludisme, la typhoïde voire même le choléra ou encore bouleverser la saison touristique. Dans ce chapitre, nous étudierons les techniques de prédictions météorologiques, ensuite nous présenterons le concept de Machine Learning et ses algorithmes. Nous terminerons avec l'état de l'art des prédictions météorologiques appliquant la méthode de Machine Learning.

### 1.1. Méthodes de prédition météorologique

#### 1.1.1. Méthodes traditionnelles

##### a) Méthode de la permanence

- Définition et principe

La permanence fait référence au fait que les conditions actuelles sont les prédictions pour un futur proche. Par exemple, si la température moyenne aujourd’hui est de 27 °C, alors la température moyenne de demain sera aussi 27°C (Cresswell 2009).

Cette méthode s'appuie sur la variation très lente de l'atmosphère excepté dans certaines circonstances telles que la pollution humaine.

- Problème résolu par la méthode

Cette méthode permet de déterminer la continuité des événements tels que le moment de l'arrivée d'un orage qui se déplace vers leur région (Iseh et al 2013).

- Limite

La principale difficulté est que cette méthode nécessite plus de 30 ans de données (Cresswell 2009).

##### b) Méthode climatologique

- Définition et principe

Cette méthode se base sur la climatologie. La climatologie descriptive traite de la distribution géographique ou temporelle des observations météorologiques sur une période de temps donnée. En général, on parle d'une moyenne de 30 années d'observations (Cresswell 2009).

Cette méthode repose sur l'apprentissage des 30 années du jeu données et ne permet pas seulement d'obtenir la moyenne mais aussi les écarts types, les valeurs extrêmes et aussi les valeurs seuils.

- **Problème résolu par la méthode**

Cette approche résout le problème d'absence d'informations relatives au climat de saison.

- **Limite**

Une des limites de cette méthode est qu'elle est utilisée comme méthode de contrôle de prédiction signifiant que toute méthode de prédiction valable devrait avoir une meilleure précision qu'elle (Murphy et Epstein 1989).

- c) **Méthode par analogie**

- **Définition et principe**

La prédiction par analogie est une méthode qui opère sur le principe simple de faire des prévisions en comparant les tendances météorologiques actuelles à des tendances similaires (ou analogues) du passé. De par sa définition, cette méthode repose sur les données du passé identique à l'instant  $t$  pour prédire la condition à l'instant  $t+1$ .

- **Problème résolu par la méthode**

Apporter des prédictions moins coûteuses en ressources de calcul comme la méthode de prédiction numérique.

- **Limite**

Il faudrait avoir une quantité massive de données pour pouvoir appliquer cette méthode. De plus avec le changement climatique, l'effet de serre et la pollution industrielle, les conditions climatiques ne peuvent pas rester identiques.

### 1.1.2. Méthode numérique

- a) **Définition et principe**

Linacre et Geerts en 1997 définissent la prédiction météorologique numérique comme étant un ensemble simplifié d'équations appelées équation primitive utilisée pour calculer les changements de conditions (Linacre et al, 1997).

La prédiction numérique repose sur le fait que les gaz constitués dans l'atmosphère obéissent à un certain nombre de principes physiques connus. Les données obtenues de ce modèle sont très biaisées et nécessitent un post-traitement appelé assimilation des données (Cresswell 2009, Apte 2015, Bauer et al 2015).

### b) Problème résolu par la méthode

Cette méthode permet de suivre les variations de l'atmosphère et fournit les données météorologiques en temps réel.

### c) Limites

Bien qu'utilisée dans les centres d'observations comme ECMWF, Synergie ou encore windy via son site, la méthode numérique de prédition météorologique a pour principale limite la quantité gigantesque de ressources de calcul pour la résolution de l'équation primitive (Bauer, Thorpe et al. 2015). De plus, cette méthode perd sa précision sans la technique de data assimilation qui nécessite des observations au-dessus de l'atmosphère. Face à ces contraintes de ressources de calculs ou encore du manque de vecteur d'observations (satellites), une nouvelle technique plus évoluée a vu le jour.

## 1.2. Machine Learning (Apprentissage Automatique)

### 1.2.1. Généralités

Artur Samuel définit en 1959 le Machine Learning comme un domaine d'étude qui donne aux ordinateurs l'habileté d'apprendre sans avoir été programmés au préalable (Artur Samuel, 1959).

Avant l'arrivée du Machine Learning, la résolution des problèmes faisait appel à une approche classique figure (1.1). Cette méthode consiste à ressortir du problème des règles à définir pour résoudre le problème puis les évaluer afin de relever d'éventuelles erreurs qui doivent être corrigées de manière itérative. Par exemple, considérons de classification de mails en spam, avec la méthode classique. Il est question de ressortir les spécificités de chaque mail pouvant être considérés comme spam et définir différentes règles correspondantes afin de pouvoir étiqueter un nouveau mail arrivant et respectant ces caractéristiques.

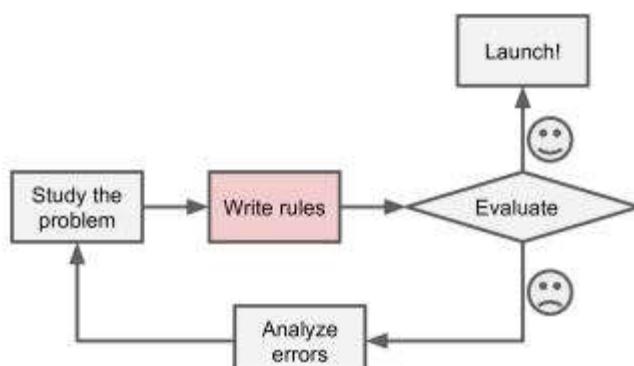


Figure 1.1 : Approche classique (Géron, 2017)

## CHAPITRE 1 : REVUE DE LA LITTÉRATURE

En contradiction avec l'approche classique précédente, les techniques de Machine Learning consiste à créer un modèle capable d'apprendre des données d'anciens mails pour reconsidérer l'exemple ci-avant et va alors trier de manière automatique (voir figure 1.2). Ce qui rendra le travail moins lourd et beaucoup plus précis.

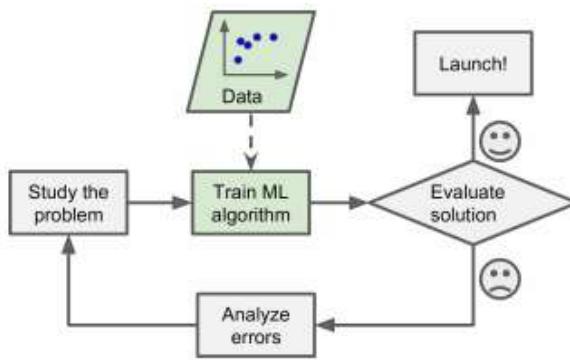


Figure 1.2 : Approche orientée Machine Learning (Géron, 2017)

Selon Géron en 2017, les systèmes de Machine Learning peuvent être regroupés en trois (03) catégories (Géron, 2017):

- La catégorie qui nécessite ou pas la supervision de l'Homme pour l'entraînement (les apprentissages supervisés, non supervisés et de renforcement) ;
- La catégorie qui apprend continuellement ou pas sur internet (l'apprentissage 'statique' ou en ligne) ;
- La catégorie qui compare simplement les nouveaux points de données aux points de données connues, ou, au contraire, détecter des modèles dans les données de formation et construire un modèle prédictif (apprentissage basé sur des exemples ou des modèles).

Dans le cadre de ce travail, nous ne présenterons que la première catégorie qui est beaucoup utilisée dans la prédiction météorologique.

Dans la résolution des problèmes à travers le Machine Learning, certains nécessitent l'intervention humaine dans la phase dite d'entraînement. Il s'agit ici de la notion d'apprentissage supervisé. C'est une approche qui consiste à étiqueter les données à la phase d'entraînement de telle manière à apprendre au système à caractériser une nouvelle donnée au vu de l'apprentissage. Cette méthode dénombre des algorithmes tels que :

### a) K-plus proches voisins (KNN)

Pour estimer la sortie associée à une nouvelle entrée  $x$ , la méthode des  $k$  plus proches voisins consiste à prendre en compte (de façon identique) les  $k$  échantillons d'apprentissage dont l'entrée est la plus proche de la nouvelle entrée  $x$ , selon une distance à définir (Cunningham and Delany 2020). L'exemple ci-dessous (voir figure 1.3) présente deux (02) classes A & B (en bleu et en rouge) et donc l'exercice consiste à catégoriser l'objet en vert soit dans A ou B en utilisant les plus proches voisins d'ordre 3, 6, etc.

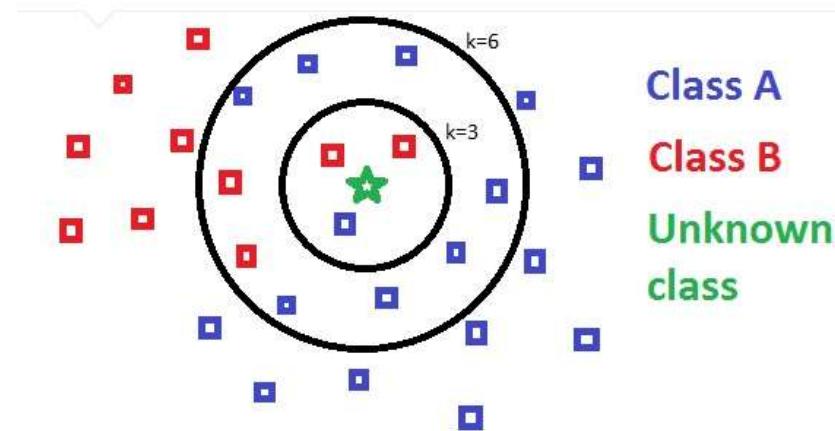


Figure 1.3 : K-plus proche voisins (Sanjay, 2018)

### b) Régression linéaire

C'est un modèle de régression qui cherche à établir une relation linéaire entre une variable, dite expliquée, et une ou plusieurs variables, dites explicatives. En général, le modèle de régression linéaire explicite la relation sous forme d'équations linéaire et la corrélation quantifie cette relation entre -1 et 1 (Yadav, 2018). La figure (1.4) illustre un tracé de droite permettant d'apprécier la relation linéaire entre deux variables;

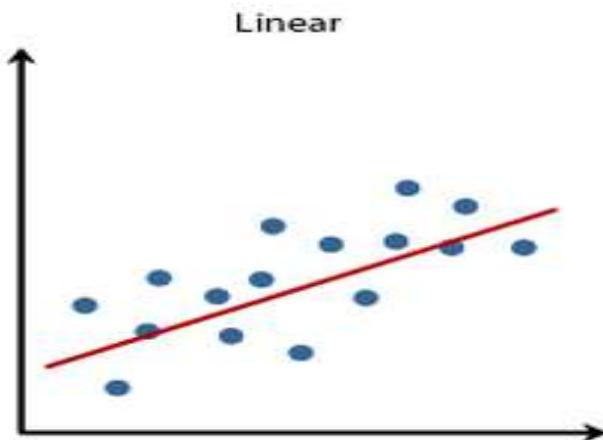


Figure 1.4 : Régression linéaire (Roman 2020)

**c) Régression Logistique**

L'idée derrière la régression logistique est d'avoir soit une sortie binaire (un évènement ou son contraire)(voir figure 1.5) ou soit une sortie multidimensionnelle dans un problème où on a une série de données avec plusieurs indicateurs auxquels on va appliquer la fonction sigmoïde pour établir des probabilités et ainsi estimer une nouvelle donnée appartenant à des sorties préétablies (Younes, 2017) ;

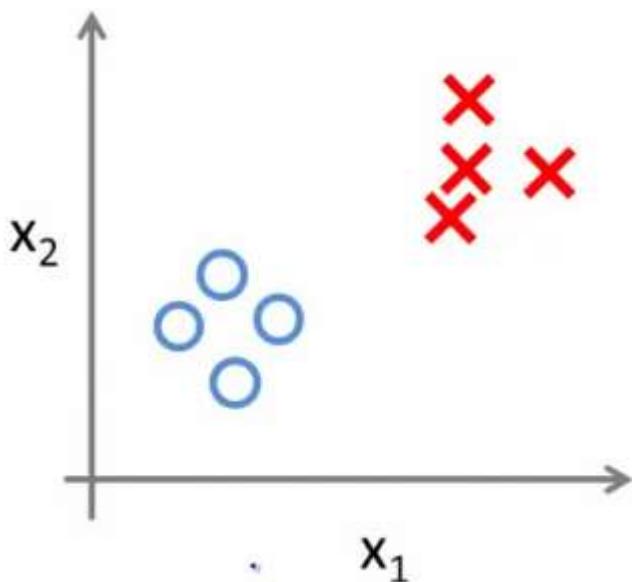


Figure 1.5 : Régression logistique, classification binaire (Younes, 2017)

**d) Support Vector Machine (SVM)**

C'est une méthode d'apprentissage très puissante et polyvalente capable d'effectuer une classification linéaire ou non linéaire, une régression, et même une détection des valeurs

aberrantes. Pour se faire, l'algorithme permet de tracer des hyperplans correspondants aux différentes propriétés de données collectées (voir figure 1.6). Ainsi, lorsqu'une nouvelle donnée est entrée, la machine va la classifier en fonction de la distance entre les valeurs de propriétés de la nouvelle donnée et les différents hyperplans correspondants aux classes existantes dans le système (Dey 2016) ;

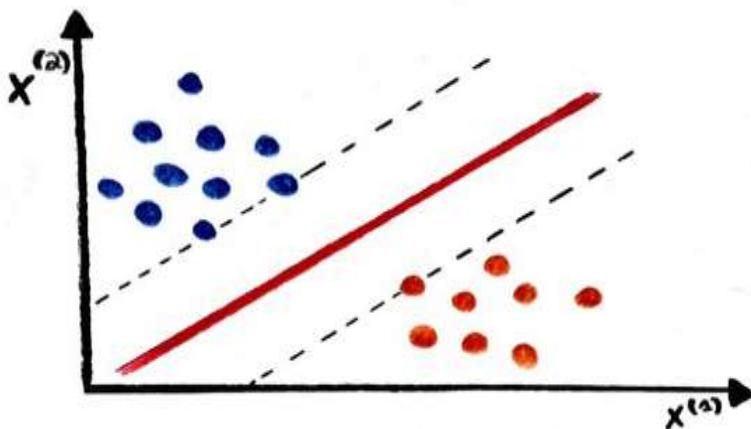


Figure 1.6 : Support Vector Machine (Sweet, 2020)

### e) L'arbre à décision

L'arbre à décision regroupe les attributs en triant en fonction de leurs valeurs. C'est principalement utilisé pour un problème de classification. Cet arbre est constitué de nœuds et de branches. Chacun des nœuds représente les attributs d'un groupe qui doit être classé (Dey 2016). La figure (1.7) présente un exemple d'arbre à décision prise en fonction de l'âge et du sexe.

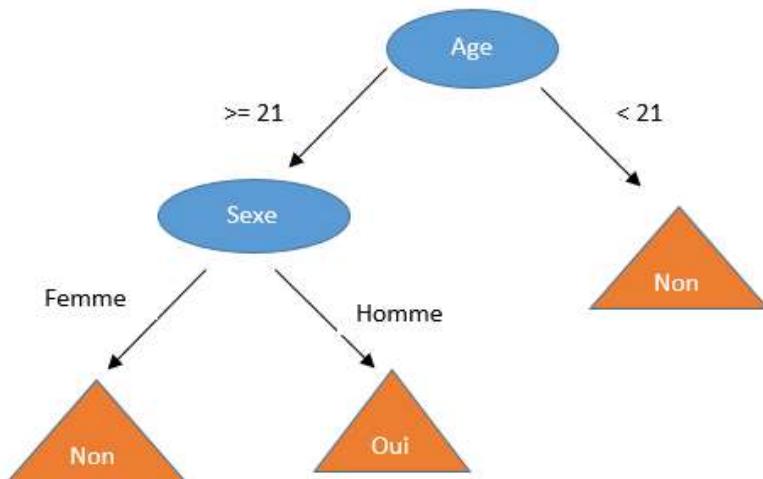


Figure 1.7 : Arbre à décision

### f) Random Forests

Appelé random forests du fait du choix aléatoire de l'arbre à décision adapté pour la résolution d'un problème. En d'autres termes, chaque nœud du random forest est un arbre à décision et l'ensemble des résultats de chacun des arbres est combiné pour ressortir un résultat final (voir figure 1.8) (Cutler et al, 2012) ;

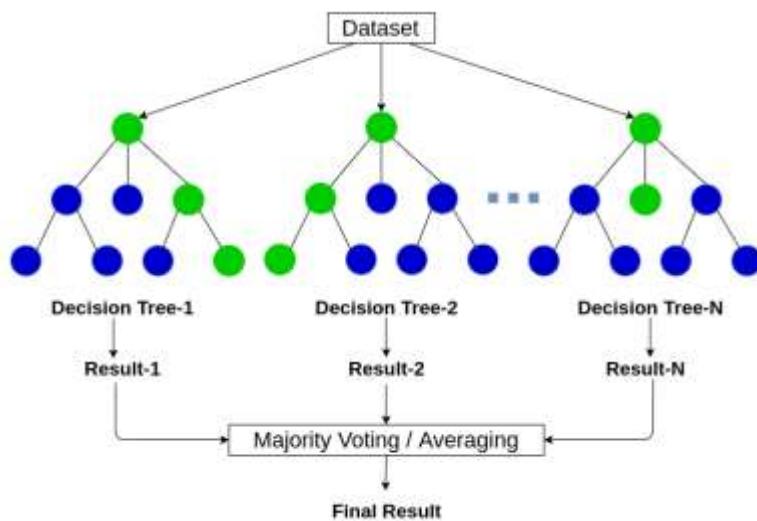


Figure 1.8 : Random forest (Abhishek, 2020)

### g) Réseau de neurones

C'est un système informatique s'inspirant du fonctionnement du cerveau humain pour apprendre. Le premier tiers reçoit les entrées d'informations brutes, un peu comme les nerfs optiques de l'être humain lorsqu'il traite des signaux visuels. Par la suite, chaque tiers reçoit les sorties d'informations du tiers précédent. On retrouve le même processus chez l'Homme, lorsque les neurones reçoivent des signaux en provenance des neurones proches du nerf optique. Le dernier tiers, quant à lui, produit les résultats du système (Shrestha and Mahmood 2019). La figure (1.9) présente une architecture d'un réseau de neurones. Celle-ci comprend quatre (04) couches :

- **La couche input** : elle prend en paramètres nos données brutes et a la même taille que celles-ci
- **Les couches hidden 1 et hidden 2** : il s'agit ici des couches intermédiaires qui prennent en entrée la sortie de la couche précédente. Dans notre cas la couche hidden 1 a pour entrée la sortie de la couche input et quant à la couche hidden 2 son entrée est la sortie de la couche hidden 1

- **La couche output :** comme les autres couches, cette couche prend ses données des couches antérieures et sa taille dépend du format de la variable à prédire.

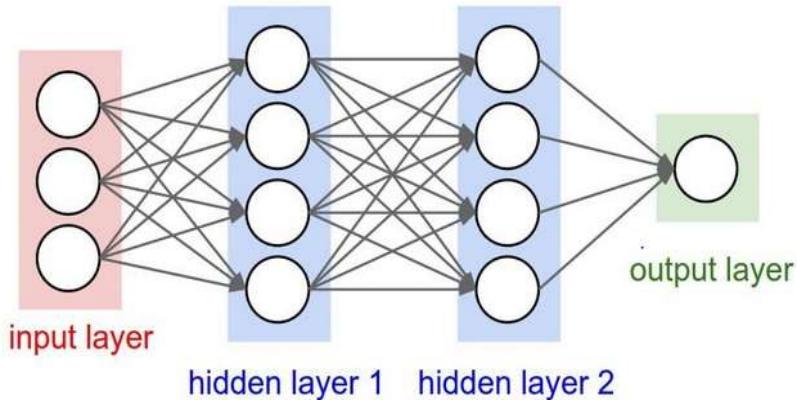


Figure 1.9 : Réseau de neurones (Bastien,2019)

Tous ces algorithmes peuvent être utilisés de façon singulière ou de façon couplée dans les activités de classification ou de prédiction par exemple.

### 1.2.2. Etat de l'art de la prédiction météorologique à l'aide du Machine Learning

Dans la partie 1.1.2. , nous avons mis en avant la méthode de prédiction météorologique numérique comme la méthode la plus utilisée en matière de prédiction des éléments du climat. Le choix du Machine Learning comme alternative de la méthode de prédiction numérique est approuvé. L'équivoque reste actuellement sur le choix d'un ou des algorithme(s) à utiliser. En 2018, Abrahamse et al proposent l'approche du réseau de neurones à l'aide du Framework TensorFlow pour prédire la température sur une échéance de 1, 3, 6 et 12 heures (Abrahamse et al, 2018). Ils justifient cette approche du Deep Learning du fait de la simplicité d'acquisition de données et du nombre important d'outils que possède le langage de programmation Python. Il est important ici de noter que le Deep Learning fait partie d'une famille plus large de méthodes d'apprentissage du Machine Learning basées sur les réseaux de neurones avec l'apprentissage par représentation. L'une des limites de ce travail est d'avoir utilisé un, voire deux paramètres (la température et les précipitations) en entrée pour prédire la température (Abrahamse et al, 2018).

Effectué la prédiction en temps réel et en fonction de tendance saisonnière de la ville de Wuhan afin d'anticiper sur les inondations, a été une étude menée par Chao en 2018 (Chao et al 2018). Basée sur les capteurs **MicroelectroMechanical Systems (MEMS)** qui collectent les

## CHAPITRE 1 : REVUE DE LA LITTÉRATURE

---

données en temps réel et les stockent dans un serveur. Ce serveur contient également les données anciennes relatives à la tendance climatique de la zone. **MEMS** ensuite décompose les données collectées en série temporelle avec l'approche ‘**seasonal and Trend**’ (**STL**) utilisant la régression logistique (Loess). L'algorithme de prédition utilisée ici est la méthode ‘**Long Short Term Memory**’ (**LSTM**), qui présente un ‘Root Mean Square Error’ (RMSE) très petit par rapport à ceux des algorithmes Random Forests et Support Vector Machine pour une prédition des cinq (05), dix (10) et quinze (15) minutes qui suivent. L'une des difficultés de cette étude est qu'elle souffre d'une erreur stochastique, ce qui entraîne la présence d'une composante résiduelle dans la décomposition en séries de temps.

En 2019, Balamurugan et al démontrent que la méthode numérique présente des limites à un moment donné (Balamurugan et al, 2019). Du fait du changement climatique en fonction du lieu où on se trouve, la prédition numérique devient ainsi obsolète et a besoin d'une méthode alternative. Ils proposent alors les algorithmes Machine Learning comme cette alternative qui, à travers la mesure des paramètres clés tels que la température, la précipitation, la pression atmosphérique, l'humidité, la vitesse du vent, etc. peuvent prédire la pluie à court terme, deux (02) jours tout au plus avec de meilleurs taux de précision. Les algorithmes présentant ici de meilleurs résultats après analyse sont les algorithmes de régression logistique et d'arbre à décision avec comme RMSE égal à 0,1126 pour tous les deux.

Ayant des traits communs en matière de climat, les procédés utilisés à Taïwan pour la prédition du paramètre pluie peuvent être également utilisés pour la ville de Yaoundé. En 2019, du fait de l'importance qu'a la pluie dans la planification ressource notamment dans l'agriculture, l'hydroélectricité et bien d'autres domaines, Yen et al proposent l'algorithme Deep Echo State Network (DeepESN) comme l'algorithme idéal pour sa prédition sur une (01) heure et ceci indépendamment du type du climat de la zone à condition d'avoir uniquement une quantité massive de données (Yen et al, 2019). Pour se faire ils recommandent l'utilisation des paramètres de pluviométrie, d'humidité et de pression pour avoir un RMSE autour de 1,69.

Dans la littérature de ces dernières années, de nombreuses recherches ont été menées dans le but d'impliquer le domaine de l'internet de l'objet notamment dans le processus d'acquisition de données météorologiques et le processus de prédition d'un ou de plusieurs éléments du climat. Nallakaruppan et al utilisent en 2019 le Raspberry pi 3 comme unité de traitement de données collectées via les capteurs de pluviométrie (pluviomètre, baromètre, température et humidité) (Nallakaruppan et al, 2019). Ces données sont stockées dans une base de données

## CHAPITRE 1 : REVUE DE LA LITTÉRATURE

---

relationnelle et qui sont ensuite exploitées par le logiciel R studio pour la construction des modèles de prédiction basés respectivement sur les algorithmes de l’arbre à décision et AutoRegression Moving Average (ARMA). De cette étude, on retient la précision de l’algorithme arbre à décision au détriment de ARMA qui dévie par rapport aux données observées.

Partant du problème de la forte consommation d’énergie des stations de météorologie et des perturbations qu’elle entraîne dans la précision des prédictions, Jakaria et al proposent en 2020 une méthode de prédiction. Celle-ci s’appuie sur des modèles de Machine Learning qui se nourrissent des données historiques des stations et fournissent les prédictions à très court terme (Jakaria et al, 2020). Parmi les modèles utilisés, seul l’algorithme ‘Support Vector Regressor’ fournit une meilleure précision comparé au ‘Ridge Regression’ qui avait une  $RMSE > 4$ . Par ailleurs, ils suggèrent la technologie de l’internet des objets pour déployer des capteurs de température et d’humidité pour améliorer les performances des prédictions.

Le tableau (1.1) présente un résumé de ces travaux antérieurs regroupés en six (06) colonnes à savoir l’année, les auteurs, l’objectif visé, paramètres utilisés, Méthodes et enfin RMSE obtenus.

## CHAPITRE 1 : REVUE DE LA LITTÉRATURE

Tableau 1.1 : Résumé des travaux antérieurs

Année	Auteurs	Objectifs visés	Paramètres utilisés	Méthodes	RMSE
2018	Abrahamse et al (Abrahamse et al, 2018)	Prédire la température sur une échéance de 1,3, 6 et 12 heures	- Température - précipitations	Deep Learning	0,0101
2019	Balamurugan et al (Balamurugan et al, 2019)	Prédire la pluie à court terme	- Température - Précipitations - Pression atmosphérique - Humidité - Vitesse du vent	Régression logistique Arbre à décision	0,1126
2019	Yen et al (Yen et al, 2019)	Prédire la pluie sur une (01) heure	- Niveau de précipitations - Humidité relative - Pression atmosphérique	Deep Echo State Network (DeepESN)	1,69
2020	Jakaria et al (Jakaria et al, 2020)	/	/	Ridge Regression	>4

## Conclusion

Dans ce chapitre, il était question pour nous de mener une étude sur les méthodes de prédiction météorologiques. De là nous retenons que la méthode numérique est actuellement utilisée dans les grands centres d'observation et de nombreuses études sont menées pour trouver une alternative à cette dernière qui est gourmande en ressources. La deuxième partie de ce chapitre présente le Machine Learning et ses algorithmes pouvant être utilisés dans la prédiction météorologique. L'implémentation de ces algorithmes dans la résolution de notre problématique fera l'objet du chapitre suivant.

## Chapitre 2 : MÉTHODOLOGIE

## Introduction

Ce chapitre porte essentiellement sur la présentation des outils nécessaires pour l'implémentation de notre solution. A cet effet, nous présenterons d'entrée de jeu notre approche de façon générale, suivra le procédé de travail pour la conception de notre modèle de prédiction. Nous terminerons par le matériel de travail.

### 2.1. Présentation de la solution

La solution que nous apportons fait appel aux domaines de Machine Learning et de l'internet des objets. Il est question pour nous de construire un modèle de prédiction grâce au Machine Learning en général et à travers un réseau de neurones en particulier. Nous avons choisi le réseau de neurones par récurrence (RNN) qui va permettre au modèle d'apprendre des anciennes données fournies par le Centre National pour la Prédition Environnementale (NCEP) aux Etats unis. Il permettra aussi de déterminer la température moyenne, ce qui nécessite les paramètres comme température maximale, température minimale, niveau de précipitation, vitesse du vent et l'humidité relative. Au sortir de cet entraînement, l'on obtiendra un modèle utilisable.

A la suite de cette partie, les technologies de l'internet des objets interviennent dans la collecte et le traitement des paramètres météorologiques en temps réel. Nous avons disposé d'une série de capteurs à savoir le DHT11, l'anémomètre et le pluviomètre qui sont reliés sur nos contrôleurs Arduino uno et Raspberry pi. Les données vont être stockées dans notre base de données non relationnelle conçue dans Influx db et pourront être visualisées en local et à distance sur un tableau de bord.

Notre solution va donc se servir des paramètres collectés par nos capteurs comme inputs à notre modèle conçu pour prédire la température moyenne.

### 2.2. Pipeline de la solution

Dans le jargon informatique, le pipeline renvoie à une séquence de composants de traitement de données. Le pipeline est très important pour les solutions orientées Machine Learning dans ce sens où il va décrire les différentes opérations et transformations appliquées à nos données. La figure (2.1) présente notre algorithme de travail pour la construction de notre modèle de prédiction météorologique. Il est subdivisé en trois (03) grandes parties à savoir la préparation des données, la construction et l'entraînement du modèle ainsi que son déploiement.

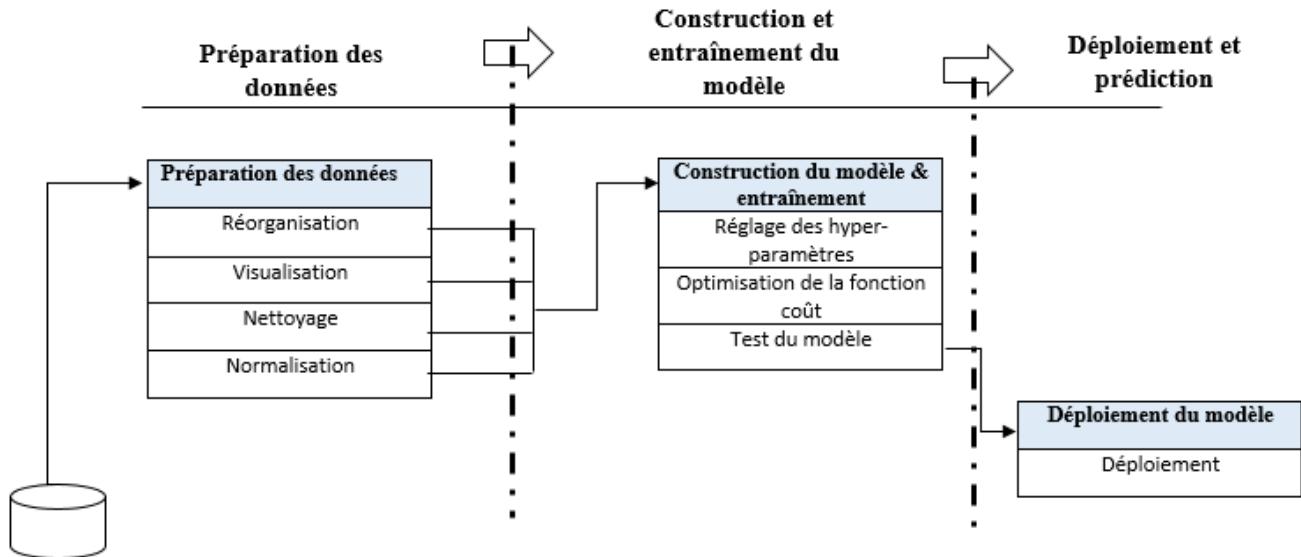


Figure 2.1 : Algorithme de travail

### 2.2.1. Données : collecte et exploration

Pour mener à bien un travail de Machine Learning en général et le nôtre en particulier, nous avons besoin d'informations assez consistantes. Le centre national américain pour la prédition environnementale (NCEP), nous a fourni trente-cinq (35) ans de données allant de 1979 à 2014 de la ville de Yaoundé. Il est important de noter que ces données sont libres de droit.

Les observations fournies par ce centre vont du 01<sup>er</sup> janvier 1979 au 31 juillet 2014. On dénombre exactement **12987** indexés et retrouve les observations suivantes :

Tableau 2.1 : Format des données fournies par le centre national américain pour la prédition environnementale

Format des données collectées
Longitude (°)
Latitude (°)
Élévation (m)
Max Température (°C)
Min Température (°C)
Précipitation (mm)
Vitesse du vent (km/h)
Humidité relative
Ensoleillement

### 2.2.2. Prétraitement des données

De nombreuses activités ont été menées pour le prétraitement de ce jeu de données.

#### a. Réorganisation des colonnes

Une fois uploadé dans notre notebook Jupyter, les colonnes du jeu de données ont été mal structurées et il a fallu les déplacer et transformer la colonne date en un format mieux organisé. Nous allons rajouter une colonne ‘Mean temperature’ pour signifier la température moyenne quotidienne (Tableau 2.2).

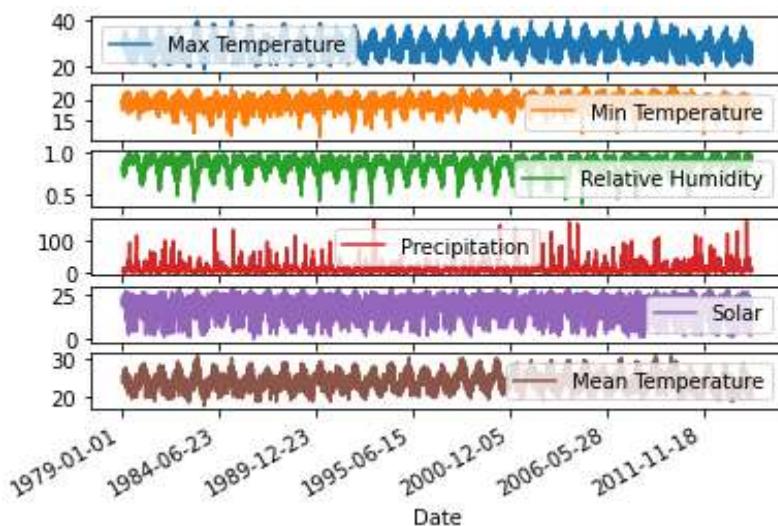
Tableau 2.2 : Données restructurées sur Jupyter Notebook

	Longitude	Latitude	Elevation	Max Temperature	Min Temperature	Precipitation	Wind	Relative Humidity	Solar	Mean Temperature
Date										
1979-01-01	11.5625	3.90286	715	32.458	18.666	0.424004	1.003633	0.729931	19.272037	25.5620
1979-01-02	11.5625	3.90286	715	28.010	19.823	0.340748	1.078598	0.837596	18.461476	23.9165
1979-01-03	11.5625	3.90286	715	28.652	19.461	0.142479	1.542989	0.822336	20.823017	24.0565
1979-01-04	11.5625	3.90286	715	29.302	18.942	0.017166	1.497211	0.814112	24.300898	24.1220
1979-01-05	11.5625	3.90286	715	30.138	17.106	0.120163	1.205390	0.814844	21.780956	23.6220

#### b. Visualisation des données

La visualisation de nos données a relevé que :

- Les paramètres météorologiques sont stationnaires au cours du temps (voir figure 2.2). En effet, les courbes évolutives de nos paramètres tracés avec la librairie **matplotlib** montrent le caractère invariant de ceux-ci.



## CHAPITRE 2 : MÉTHODOLOGIE

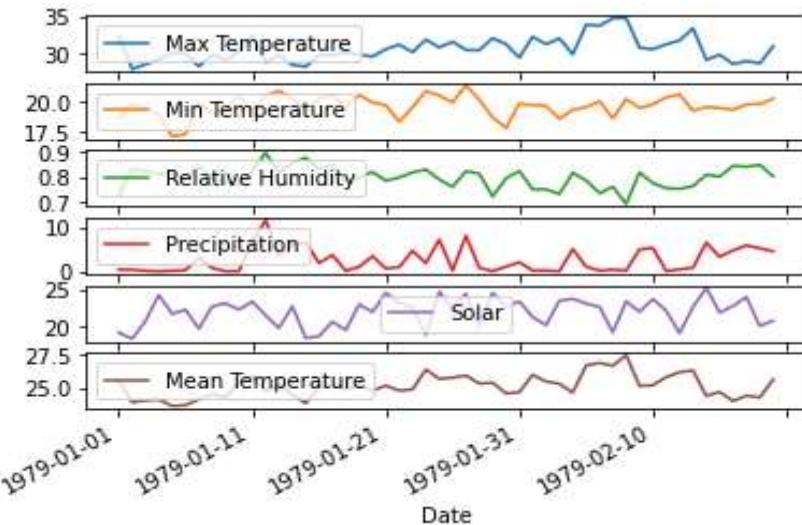


Figure 2.2 : Visualisation des données

- La corrélation entre les différentes propriétés de nos données voir figure (2.3). Après examen du paramètre température moyenne, nous constatons une forte corrélation avec l'ensoleillement, une corrélation moyenne avec la précipitation et l'intensité du vent et pas de corrélation avec l'humidité relative. L'équation (1) a été utilisée pour effectuer le calcul.

$$\rho(x, y) = \frac{\sum_{i=1}^N (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^N (x_i - \bar{x})^2} \cdot \sqrt{\sum_{i=1}^N (y_i - \bar{y})^2}} = \frac{COV(X,Y)}{\sqrt{V(X)} \cdot \sqrt{V(Y)}} \quad (1)$$

Où  $\bar{x}$  et  $\bar{y}$  sont les moyennes respectives de x et y et  $V(X)$ ,  $V(Y)$  leurs variances respectives.

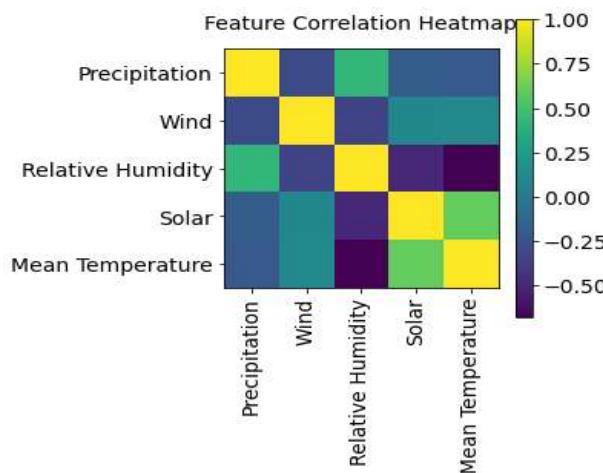


Figure 2.3 : Corrélation entre les données

### c. Nettoyage des données

Le nettoyage des données va consister à enlever les propriétés longitude, latitude et élévation qui n'influent pas sur la température moyenne d'une part et d'autre part de vérifier si notre jeu de données est complet (Tableau 2.3).

Tableau 2.3 : Tableau résultant du nettoyage des données

Date	Max Temperature	Min Temperature	Precipitation	Wind	Relative Humidity	Solar	Mean Temperature
1979-01-01	32.458	18.666	0.424004	1.003633	0.729931	19.272037	25.5620
1979-01-02	28.010	19.823	0.340748	1.078598	0.837596	18.461476	23.9165
1979-01-03	28.652	19.461	0.142479	1.542989	0.822336	20.823017	24.0565
1979-01-04	29.302	18.942	0.017166	1.497211	0.814112	24.300898	24.1220
1979-01-05	30.138	17.106	0.120163	1.205390	0.814844	21.780956	23.6220

### d. Normalisation des données

La normalisation de nos données va consister à rendre nos données uniformes dans l'intervalle 0 et 1 à l'aide d'une fonction préconçue par le framework Sklearn. Cette fonction suit l'équation :

$$X_{std} = \frac{X - X.\min(axis=0)}{X.\max(axis=0) - X.\min(axis=0)} \quad (2)$$

$$X_{scaled} = X_{std} * (max - min) + min \quad (3)$$

Où *min* et *max* sont les extrêmes de la caractéristique *X*, *axis = 0* renvoie aux index de notre *dataframe*

Par ailleurs nous allons également transformer nos propriétés nettoyées en chiffre (Tableau 2.5):

## CHAPITRE 2 : MÉTHODOLOGIE

---

Tableau 2.4 : Normalisation des champs

Champ	Correspondance
0	Max Temperature
1	Min Temperature
2	Precipitation
3	Wind
4	Relative Humidity
5	Solar
6	Mean Temperature

Tableau 2.5 : Tableau de données normalisées

	0	1	2	3	4	5	6	Date
0	0.630789	0.638749	0.002667	0.245146	0.586741	0.671724	0.594862	1979-01-01
1	0.430393	0.731301	0.002143	0.277425	0.760313	0.643472	0.470359	1979-01-02
2	0.459317	0.702344	0.000896	0.477387	0.735712	0.725783	0.480952	1979-01-03
3	0.488602	0.660827	0.000108	0.457676	0.722452	0.847005	0.485908	1979-01-04
4	0.526266	0.513959	0.000756	0.332021	0.723633	0.759172	0.448076	1979-01-05

### 2.3. Modèle mathématique du modèle

Dans le Machine Learning, la prédiction météorologique entre dans la catégorie de problèmes dits série temporelle. Une série temporelle renvoie à un paramètre qui est fonction du temps. Dans la visualisation de nos données, nous avons observé ce caractère stationnaire de la température moyenne.

Il n'existe pas dans la littérature une relation linéaire entre la température moyenne et les autres propriétés que nous avons dans notre jeu de données. Pour pallier à cela nous avons adopté un réseau de neurones par récurrence non pas traditionnel mais évolué dénommé LSTM. Il a la capacité de traiter et de prédire une séquence d'une série temporelle en tenant compte des événements passés dans le temps. A cela nous avons ajouté une couche bidirectionnelle LSTM qui va apprendre à la fois des événements passés et des événements futurs.

### 2.3.1. Réseau de neurones par récurrence (RNN)

Il s'agit d'un réseau de neurones avec des boucles à l'intérieur qui entraînent la persistance de l'information. La structure d'un RNN est illustrée sur la figure (2.4)

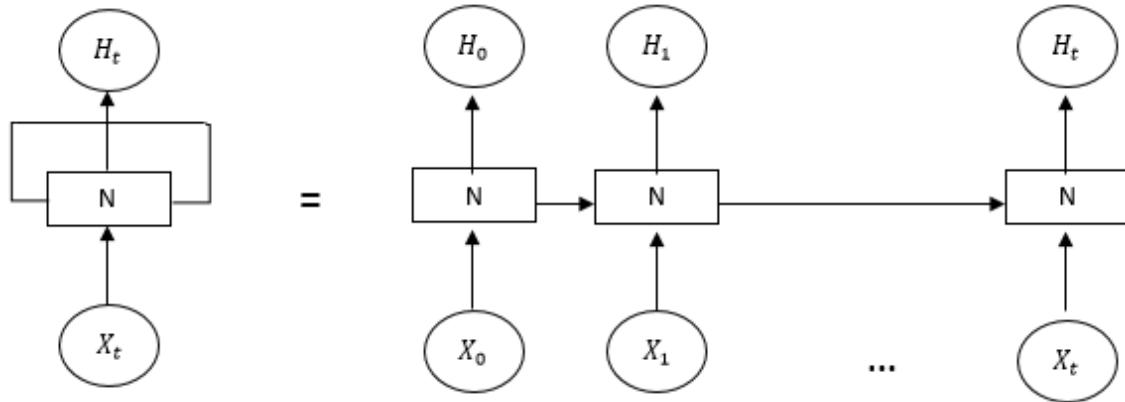


Figure 2.4: Structure du RNN

Ayant une séquence d'entrée ( $x_1, x_2, \dots, x_n$ ) un RNN standard calcule une séquence de sortie ( $y_1, y_2, \dots, y_n$ ) par itération en suivant l'équation suivante (Yu et al. 2015, Zaytar et al. 2016) :

$$h_t = \sigma(W^{hx} x_t + W^{hh} h_{t-1} + b_h) \quad (4)$$

$$y_t = W^{yh} h_t + b_o \quad (5)$$

$W^{hx}, W^{hh}, W^{yh}$  sont les poids et  $\sigma$  est Fonction Sigmoïde

### 2.3.2. Long Short Term Memory (LSTM)

#### a) LSTM principe de base

LSTM est un type de réseau de neurones par récurrence capable de tirer parti de la dépendance sur le long terme dans une séquence tout en atténuant les problèmes de perte de gradient (Hewamalage, Bergmeir et al. 2020). Comparée à la cellule RNN basique, la cellule LSTM a deux (02) composants à son état. L'état caché ('hidden') correspond à la mémoire à court-terme et l'état interne ('internal') correspond à la mémoire à long terme. La structure d'un bloc LSTM basique est illustrée dans la figure (2.5)

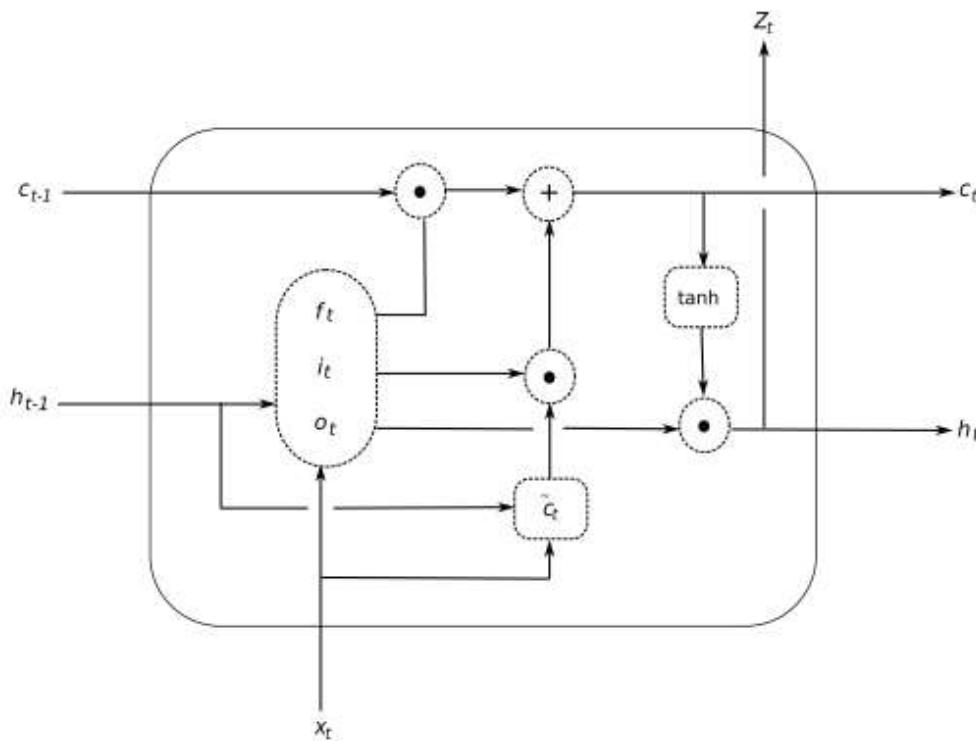


Figure 2.5 : Structure d'un bloc LSTM basique (Hewamalage et al. 2020)

$$i_t = \sigma(W_i \cdot h_{t-1} + V_i \cdot x_t + b_i) \quad (6)$$

$$o_t = \sigma(W_o \cdot h_{t-1} + V_o \cdot x_t + b_o) \quad (7)$$

$$f_t = \sigma(W_f \cdot h_{t-1} + V_f \cdot x_t + b_f) \quad (8)$$

$$\hat{C}_t = \tanh(W_c \cdot h_{t-1} + V_c \cdot x_t + b_c) \quad (9)$$

$$C_t = i_t \odot \hat{C}_t + f_t \odot C_{t-1} \quad (10)$$

$$h_t = o_t \odot \tanh(C_t) \quad (11)$$

$$z_t = h_t \quad (12)$$

Dans les équations 6-12,  $h_t \in \mathbb{R}^d$  est un vecteur qui dénote l'état caché ('hidden') de la cellule où  $d$  est la dimension de la cellule.

$C_t \in \mathbb{R}^d$  est l'état de la cellule et  $\hat{C}_t \in \mathbb{R}^d$  est l'état de la cellule candidate à l'étape  $t$  qui capture les informations importantes qui doivent être conservées jusqu'à l'avenir.  $W_i, W_o, W_f$  et  $W_c \in \mathbb{R}^{d \times d}$  représentent respectivement les matrices poids des portes d'entrée, de sortie, de rejet (oubli) et de l'état de la cellule. De même  $V_i, V_o, V_f, V_c \in \mathbb{R}^{d \times d}$  et  $b_i, b_o, b_f, b_c \in \mathbb{R}^d$

## CHAPITRE 2 : MÉTHODOLOGIE

---

désignent les matrices de poids correspondant à l'entrée de courant et les vecteurs de biais respectifs.  $i_t$ ,  $o_t$  et  $f_t \in \mathbb{R}^d$  sont les vecteurs portes d'entrée, de sortie et de rejet (ou d'oubli).

Dans l'équation 8, les portes d'entrée et d'oubli déterminent ensemble la part des informations passées à conserver dans l'état actuel de la cellule et la part du contexte actuel à propager vers le pas de temps futur.  $\odot$  désigne la multiplication par élément encore appelé le produit de Hudmard.

La valeur 0 dans la porte d'oubli  $f_t$  indique que rien ne doit être reporté de l'état cellulaire précédent. En d'autres termes, l'état cellulaire précédent doit être complètement oublié dans l'état actuel de la cellule. Selon cet argument, la valeur 1 implique que l'état précédent de la cellule devrait être entièrement conservé. La même notion s'applique aux deux autres portes, à savoir la porte d'entrée  $i_t$  et la porte de sortie  $o_t$ . (Hewamalage, Bergmeir et al. 2020)

Une valeur entre les deux extrêmes de 0 et 1 pour les deux portes d'entrée et d'oubli peut contrôler soigneusement la valeur de l'état actuel de la cellule en utilisant uniquement les informations importantes de l'état précédent de la cellule et l'état actuel de la cellule candidate. Pour l'état de la cellule candidate, la fonction d'activation est une fonction hyperbolique ou une fonction tangente qui produit des valeurs dans la plage [-1, 1]. Une différence importante de la cellule LSTM par rapport à la cellule RNN simple est que sa sortie  $z_t$  est égale à l'état caché  $f_t$  (Hewamalage, Bergmeir et al. 2020).

### b) Cellule LSTM

La figure (2.9) illustre l'état mémoire de la cellule LSTM et les équations 12-16 régissent son fonctionnement (Yu, Ramanarayanan et al. 2015).

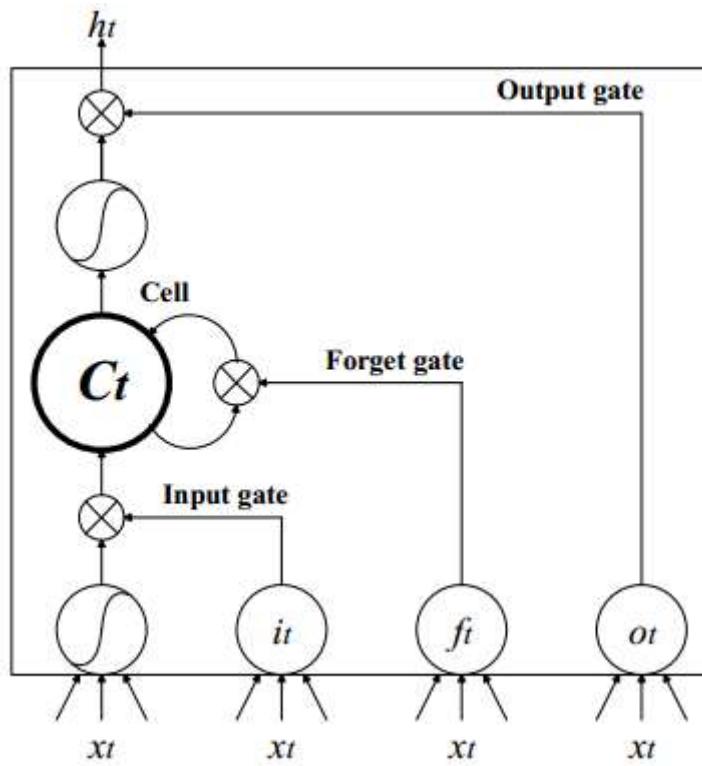


Figure 2.9 : Cellule d'une cellule LSTM (Yu, Ramanarayanan et al. 2015)

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \quad (13)$$

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f) \quad (14)$$

$$c_t = f_t c_{t-1} + i_t \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \quad (15)$$

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_t + b_o) \quad (16)$$

$$h_t = o_t \tanh(c_t) \quad (17)$$

$c$  est le vecteur d'activation de la cellule.

### c) LSTM bidirectionnel

L'architecture basique du LSTM s'appuie sur l'état de la cellule antérieure. Par contre le bidirectionnel LSTM effectue le traitement dans les deux sens avec deux couches cachées séparées, qui sont ensuite transmises à la même couche de sortie (Yu, Ramanarayanan et al. 2015). Les équations 18-20 décrivent le fonctionnement.

$$\vec{h}_t = H(W_{x\vec{h}}x_t + W_{\vec{h}\vec{h}}\vec{h}_{t+1} + b_{\vec{h}}) \quad (18)$$

$$\overleftarrow{h}_t = H(W_{\overleftarrow{x}\overleftarrow{h}}\overleftarrow{x}_t + W_{\overleftarrow{h}\overleftarrow{h}}\overleftarrow{h}_{t+1} + b_{\overleftarrow{h}}) \quad (19)$$

$$\mathbf{y}_t = \mathbf{W}_{\vec{h}\mathbf{y}} \overrightarrow{\mathbf{h}_t} + \mathbf{W}_{\bar{h}\mathbf{y}} \overleftarrow{\mathbf{h}_y} + \mathbf{b}_y \quad (20)$$

**H** : Fonction d'activation

$\overrightarrow{\mathbf{h}_t}$  : Séquence cachée avant

$\overleftarrow{\mathbf{h}_t}$  : Séquence cachée arrière

**b** : biais associé

### 2.3.3. Fonction d'optimisation

La description de LSTM dans la section 2.3.2. nous relève qu'il atténue la perte du gradient. Parti de cette information nous avons choisi le gradient descendant stochastique (SGD) pour optimiser notre algorithme. L'équation (21) donne la formule du SGD.

$$\boldsymbol{\theta} = \boldsymbol{\theta} - \eta \cdot \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}; \mathbf{x}^{(i)}; \mathbf{y}^{(i)}) \quad (21)$$

$\eta$  : Taux d'apprentissage

Pour optimiser notre modèle nous avons utilisé 75 epochs, 256 batch-size et un taux d'apprentissage de 0.0006.

Tableau 2.6 : Paramètre d'entraînement

Epochs	Batch-Size	Taux d'apprentissage
75	256	0.0006

### 2.3.4. Métriques d'évaluation des performances

La métrique renvoie à la fonction de coût que nous chercherons à optimiser durant le processus d'entraînement. Grâce à Chai et al (2014) nous avons choisi (Chai et al, 2014) :

- **Le carré moyen d'erreur** : il peut s'écrire comme la somme de la variance de l'estimateur et du biais au carré de l'estimateur. Sa formule est à l'équation (22).

$$MSE = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 \quad (22)$$

$n$  : Taille de l'échantillon

$Y_i$  : Valeur observée

$\hat{Y}_i$  : Valeur prédictive

- **L'erreur moyenne absolue** : elle représente la moyenne de la moitié de la distribution normale. Sa formule est à l'équation (23).

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |Y_i - X_i| \quad (23)$$

*n* : Taille de l'échantillon

*Y<sub>i</sub>* : Prédiction

*X<sub>i</sub>*: Valeur vraie

- **L'erreur quadratique moyenne** : il s'agit de la racine carrée de la fonction carré moyen d'erreur. Sa formule est à l'équation (24).

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2} \quad (24)$$

## 2.4. Outils d'implémentation

### 2.4.1. Matériels physiques

#### a. Raspberry pi

Dans notre solution, le Raspberry pi joue un rôle de station de base. C'est dans ce dernier que vont être installées et déployées les plateformes (open source) pour notre base de données, pour notre interface visualisation et pour notre prédicteur. Le choix du Raspberry pi s'est fait sur la base du coût, de la flexibilité des outils de développement et de ses caractéristiques. Nous avons utilisé le Raspberry pi 4 B avec 32 Gb de stockage et 4 Gb de ‘Random Access Memory’ (RAM) (voir figure 2.6).

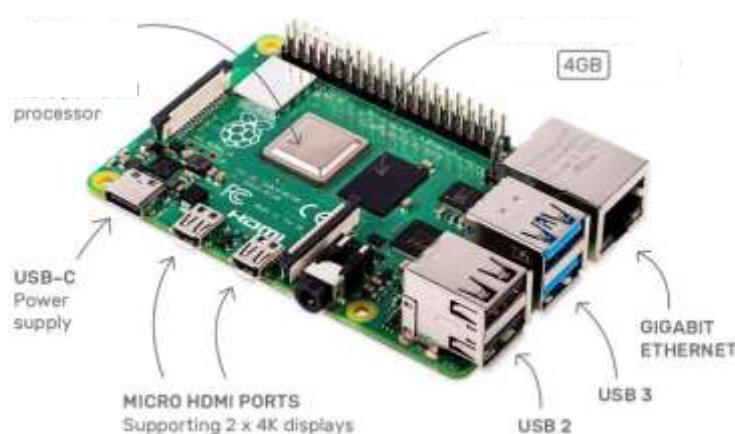


Figure 2.6 : Raspberry pi 4 B

### b. Arduino Uno

Arduino Uno une carte de microcontrôleur à source ouverte basée sur un Microchip CPU – 8 bits (Central Processing Unit) et va agir en intermédiaire entre nos capteurs et notre station de base qui est le Raspberry pi. En effet, nos capteurs ont des sorties analogiques et comparé au Raspberry, l'Arduino possède des pins analogiques (voir figure 2.7) qui permettront de mesurer la tension (0- 5 V) des capteurs et par la suite numériser (0-255).

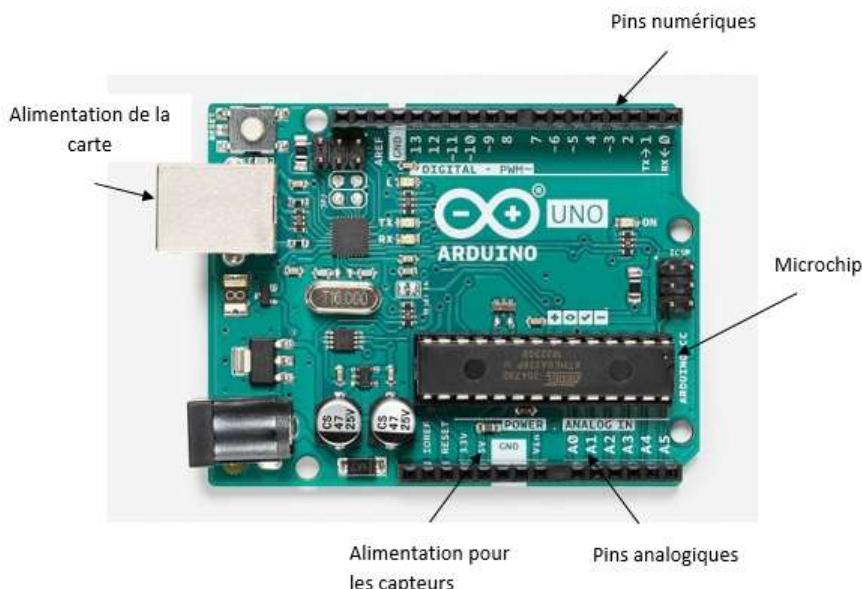


Figure 2.7 : Arduino Uno

### c. Capteurs

- DHT11** : c'est un capteur numérique qui mesure la température et l'humidité relative de l'air tout autour de lui. Il présente trois (03) pins (voir figure 2.8) à savoir un pin signal pour porter l'information et deux (02) pins pour l'alimentation (la tension et la masse).

## CHAPITRE 2 : MÉTHODOLOGIE

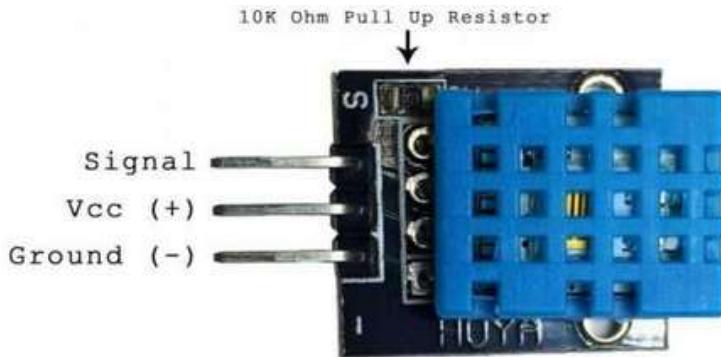


Figure 2.8 : Capteur DHT11

- **Anémomètre :** Ce capteur permet de relever la direction et la vitesse du vent (voir figure 2.9). La direction du vent est mesurée grâce à un potentiomètre. C'est une résistance qui varie en fonction de la direction. Cette partie de l'anémomètre est connectée sur l'un des pins analogiques de l'arduino. Quant à la mesure de l'intensité du vent, elle est détectée par l'ouverture et la fermeture d'un interrupteur à lames. Un petit aimant placé sur les ventouses passe sur ce commutateur une fois par tour. La sortie de ce coupe-vent est connectée à une entrée numérique de l'Arduino. Le nombre de tours est compté pendant une période de temps qui est ensuite convertie en vitesse du vent. La durée utilisée est de 2,5 secondes.



Figure 2.9 : Anémomètre - capteur de la vitesse du vent avec RJ11 - pour station météorologique - N25FR WH1080

- **Pluviomètre :** Le capteur de pluie apparaît à l'Arduino comme un interrupteur et est câblé de la même manière qu'un interrupteur à l'Arduino. Il possède une sortie soit normalement ouverte (NO) soit normalement fermée (NC). Lorsqu'aucune pluie n'est détectée, l'entrée numérique du microcontrôleur (Arduino) est portée à 5V par la

résistance de 10K. Si la pluie est détectée, la sortie NO sera fermée et l'entrée numérique sera mise à la terre. Pour détecter l'ouverture et la fermeture du relais lorsque de la pluie est détectée, nous devons utiliser une entrée numérique qui prend en charge les interruptions. Pour résumer lorsque la pluie est détectée, ça crée une interruption dans le code et comptabilise ainsi le nombre de sauts de la goutte et quantifie la pluie tombée. La figure (2.10) présente le pluviomètre.



Figure 2.10 : Pluviomètre

- **Station météo :** il ne s'agit de rien d'autres que l'ensemble des capteurs ci-haut greffé sur une tige verticale qui sera fixée à l'extérieur pour l'anémomètre et le pluviomètre. Les autres capteurs sont à l'intérieur avec l'Arduino et le Raspberry pi (voir figure 2.11).



Figure 2.11 : Mini Station météo

- **Caractéristiques de l'ordinateur utilisé** : il s'agit d'une machine virtuelle de Google Colaboratory avec 16 Gb de ‘Random Access Memory’ (RAM) et un ‘TensorFlow Processing Unit’ (TPU).

#### 2.4.2. Architecture

D'une vue d'ensemble, notre solution peut être subdivisée en deux blocs (voir figure 2.12) :

- **La station de mesure** : ce bloc permet de mesurer le niveau de précipitation, la direction et l'intensité du vent à l'aide des équipements de mesures ci-haut. Les données collectées sont analogiques et donc elles sont traitées par l'Arduino ;
- **La station de base** : Ce bloc va permettre d'une part d'effectuer les mesures de température et d'humidité (DHT11) et d'autre part de supporter Influx dB et Grafana. Cette partie est le cœur de notre solution.

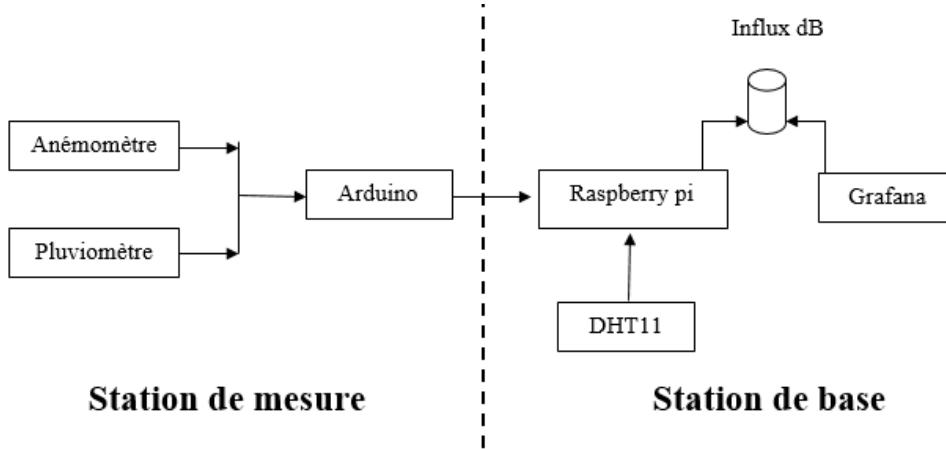


Figure 2.12 : Architecture de notre solution

#### 2.4.3. Environnement logiciel

##### a. Python

Parmi tous les langages de programmation, Python est le langage le plus flexible et dit de très haut niveau grâce à sa panoplie de librairies. Dans notre solution il est utilisé sur le Raspberry pi déjà pour la collecte des données depuis l'Arduino, pour la sauvegarde de ces paramètres dans la base de données et enfin pour la prédiction.

##### b. Arduino IDE

C'est un environnement de programmation de microcontrôleurs. Basé sur le langage C, Arduino IDE permet d'écrire les instructions d'Arduino et monitorer les valeurs mesurées par les capteurs via un port série.

### c. Raspbian

C'est le système d'exploitation supporté par le Raspberry pi. Il est basé sur Debian, une distribution de Linux qui est constitué des logiciels libres et open-source. Nous allons utiliser la nouvelle version de Raspbian appelée Buster qui a pour avantage de supporter le nouveau processeur du Raspberry pi 4 B.

### d. Influx dB

Il s'agit d'un système de bases de données fondé sur le concept 'time séries'. Ce qui signifie que chaque donnée contient une colonne dite 'timestamp' qui contient date et heure de l'enregistrement de la donnée. Grâce à ce concept, Influx dB devient l'une des meilleures bases de données pour le monitoring des paramètres ou des évènements. Dans notre solution il est question des paramètres climatiques température actuelle, la température prédictive, humidité, précipitation, vitesse et direction du vent.

La structure générale d'une table d'Influx db est la suivante :

```
<measurement>[,<tag-key>=<tag-value>...] <field-key>=<field-value>[,<field2-key>=<field2-value>...] [unix-nano-timestamp]
```

Ou dans notre cas :

Tableau 2.7 : Structure de notre base de données

Syntaxe générale	Notre solution
Measurement	Data
Tag	Yaoundé
Field_1	Temperature C
Field_2	Humidity %
Field_3	RainFalloneHour
Field_4	WindSpeedMax
Field_5	WindDirection
Field_6	forecast
Field_7	time

### e. Grafana

C'est une plateforme open-source de visualisation et d'analyse de métriques. Notre choix d'utiliser Grafana repose sur sa rapidité et sa compatibilité avec le Raspberry pi sans oublier

son adhésion avec notre base de données Influx dB. Dans notre cas, les métriques sont la température, l'humidité, le niveau de précipitations, la direction et la vitesse du vent.

### f. Jupyter Notebook

C'est une application web open-source qui permet d'exécuter différents codes sous le noyau Python. Comme activités pouvant être exécutées nous avons : le nettoyage et la transformation de données, la simulation numérique, la modélisation statistique, la visualisation de données, l'apprentissage machine, et bien plus encore.

## Conclusion

Tout le long de ce chapitre, nous avons décrit le processus d'exécution de notre travail. Ceci commence par la collecte de données et se termine par la construction d'un modèle de prédiction. Par la suite nous avons présenté les méthodes d'évaluation et la fonction d'optimisation. La deuxième partie de ce chapitre présente l'étude conceptuelle de l'outil que nous avons développé. Les résultats obtenus sur cette partie feront l'objet du chapitre suivant.

## **Chapitre 3 : RÉSULTATS ET INTERPRÉTATION**

## Introduction

Après description de notre méthodologie pour résoudre le problème posé, nous présenterons dans ce chapitre nos différents résultats obtenus. A l'entame, ce chapitre traitera de la structure de nos données minutieusement collectées. Par la suite, nous dévoilerons le tableau de visualisation des données. Pour terminer nous exposerons les résultats obtenus de notre modèle de prédiction suivi d'une évaluation de son coût.

### 3.1. Collecte et sauvegarde de données

#### 3.1.1. Rappel de l'objectif

Face à notre problème, notre premier objectif spécifique est celui de développer une méthode pour la collecte et la sauvegarde de paramètres climatiques. Ces paramètres ici sont la température, l'humidité relative, le niveau des précipitations, la direction et la vitesse du vent.

#### 3.1.2. Présentation du système de collecte et sauvegarde de données

Pour parvenir à cet objectif nous avons utilisé les outils évoqués dans la partie **2.4**. Pour rappel, il s'agit d'un travail effectué par notre station de mesure et par notre station de base. La collecte de données s'est faite tous les jours à partir du 10 novembre 2020, 24 h/ 24 au lieu-dit «chapelle Nsimeyong». La sauvegarde de ces données a été effectuée dans Influx db dans un intervalle de temps d'une (01) minute.

Avant de présenter nos données, nous nous attarderons sur le volet sécurité. Du fait de sa sensibilité, nous lui avons accordé une grande considération. Notre sécurité s'élève à deux niveaux :

- **La sécurité de la station de base**

La figure (3.1) illustre la session d'authentification dans notre station de base qui est le Raspberry Pi. Pour accéder à son interface, nous avons utilisé l'adresse statique de notre station **192.168.100.112**, adresse statique que nous lui avons attribuée.

## CHAPITRE 3 : RÉSULTATS ET INTERPRÉTATIONS

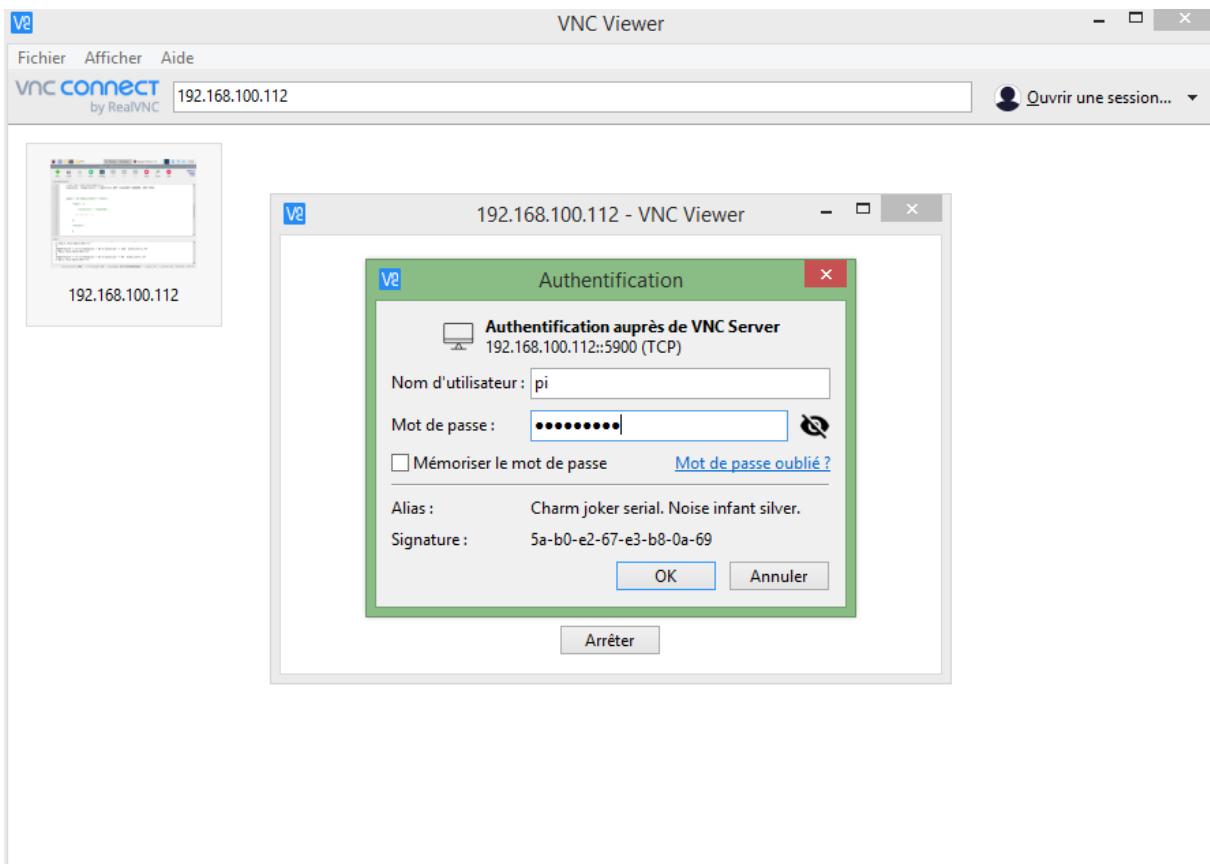


Figure 3.1 : Session d'authentification dans la station de base

- **La sécurité de notre base de données**

Pour accéder aux données, nous avons besoin d'un identifiant et d'un mot de passe. Par défaut nous avons mis **admin** pour les deux champs.

La figure (3.2) illustre le procédé pour accéder à nos données.

A screenshot of a terminal window with the title bar "pi@raspberrypi: ~". The window contains a command-line interface for InfluxDB. The user has run the command "influx -username 'admin' -password 'admin'" and connected to the database. The session shows the following output:

```
pi@raspberrypi:~ $ influx -username 'admin' -password 'admin'
Connected to http://localhost:8086 version 1.8.3
InfluxDB shell version: 1.8.3
> use station1
Using database station1
> 
```

Figure 3.2 : Critère d'accès à la base de données influx db

## CHAPITRE 3 : RÉSULTATS ET INTERPRÉTATIONS

---

Notre base de données est appelée **station1** qui contient la table **Data** comportant sept (07) champs :

- Température en degrés Celsius (°C) ;
- Humidité relative en pourcentage (%) ;
- Vitesse du vent en kilomètre par heure (km/h) ;
- Direction du vent en degré par rapport au Nord magnétique (°) ;
- Niveau de précipitation (mm)
- Localisation de la station qui est Yaoundé ;
- L'heure de la prise de mesure par rapport à l'heure universelle autrement dit ‘Universal Time Coordinated’ (UTC).

Le tableau (3.1) présente un aperçu du format de nos données avec les différents champs décrits ci-haut. Nous avons exporté ces données dans un format csv facilement exploitable.

Tableau 3.1 : Données collectées par notre mini-station météo

	Humidity %	Location	RainFalloneHour	Temperature C	WindDirection	WindSpeedMax	time
0	82.0	Yaounde	0.0	25.0	225	0.0	2020-11-14T00:00:58.595970747Z
1	82.0	Yaounde	0.0	25.0	225	0.0	2020-11-14T00:02:19.305819717Z
2	82.0	Yaounde	0.0	25.0	0	0.0	2020-11-14T00:03:30.832329223Z
3	82.0	Yaounde	0.0	25.0	225	0.0	2020-11-14T00:04:41.597082623Z
4	NaN	Yaounde	0.0	NaN	0	0.0	2020-11-14T00:05:53.129950685Z

Le tableau (3.2) nous apporte une analyse statistique.

Tableau 3.2 : Statistiques des données collectées du Novembre 2020 par notre mini-station météo

	Humidity %	RainFalloneHour	Temperature C	WindDirection	WindSpeedMax
count	9120.000000	10654.000000	9120.000000	10654.000000	10654.000000
mean	73.851425	0.002393	25.830921	154.192791	0.526953
std	6.039491	0.034858	1.147872	47.288248	0.733921
min	55.000000	0.000000	24.000000	0.000000	0.000000
25%	70.000000	0.000000	25.000000	135.000000	0.000000
50%	75.000000	0.000000	26.000000	135.000000	0.000000
75%	78.000000	0.000000	27.000000	180.000000	0.890000
max	85.000000	0.510000	29.000000	315.000000	4.020000

On observe ici :

- La température minimale est de 24 °C
- La température maximale est de 29 °C
- Le niveau de précipitation est 0 mm

### 3.1.3. Discussion

De la description faite ci-avant on peut dégager que :

- Notre solution collecte les paramètres climatiques (température, humidité relative, le niveau de précipitation, la vitesse et la direction du vent) en temps réel ;
- Notre solution permet de sauvegarder les paramètres climatiques prélevés dans une zone pendant une longue durée ;
- Notre base de données est sécurisée et nécessite des identifiants ;
- Les données prélevées peuvent être exploitables après traitement dans d'autres domaines notamment l'agriculture, la santé, le génie civil, etc.
- Les données peuvent être accessibles à distance grâce à l'usage de l'internet des objets.

Le tableau (3.3) présente une comparaison entre nos données mesurées à l'aide de notre station météorologiques et le jeu de données obtenus sur le site <https://globalweather.tamu.edu/>

Tableau 3.3 : Tableau de comparaison des données avec celles du Centre national américain pour les prévisions météorologiques

Critères	Données mesurées	Données de NCEP
Intégrité	Oui	Pas moyen de vérifier
Type de collecte	Prise en surface	Prise au-dessus de l'atmosphère
Fréquence de collecte	Donnée à la minute	Donnée journalière
Qualité de données	Besoin de post-traitement	Donnée déjà traitée

## 3.2. Visualisation des données

### 3.2.1. Rappel de l'objectif

Cet objectif spécifique doit permettre la visualisation des paramètres climatiques à travers un tableau de bord.

### 3.2.2. Présentation de la plateforme de visualisation

Comme décrit dans la partie **2.4**, Grafana a été utilisé pour visualiser les données dans un réseau local via notre adresse statique **192.168.100.112:3000**.

La figure (3.3) présente la page d'accueil une fois après avoir saisi l'adresse dans un navigateur. Nous avons besoin de s'identifier avant d'accéder au tableau de bord. Les identifiants par défaut sont **admin** pour le login et **admin** pour le mot de passe.

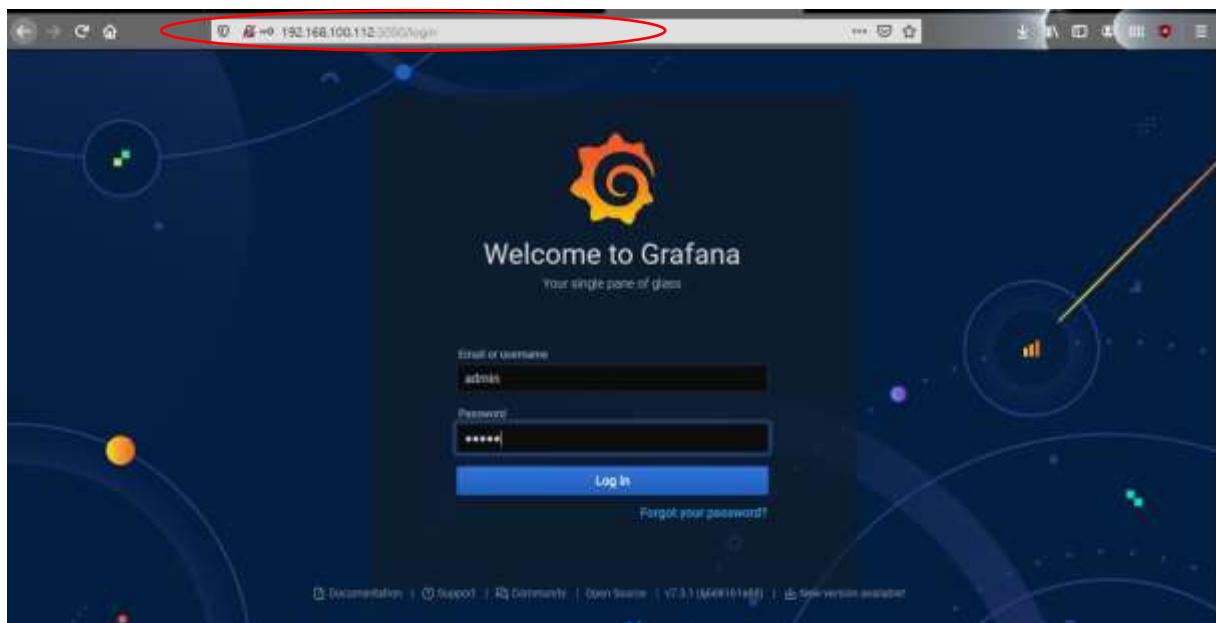


Figure 3.3 : Login du tableau de bord

Cette mesure de sécurité a pour but de préserver la confidentialité de nos données. Par conséquent on a un contrôle d'accès de base mis en place.

## CHAPITRE 3 : RÉSULTATS ET INTERPRÉTATIONS

Une fois l'authentification effectuée, nous tombons sur le tableau de bord (figure 3.4).



Figure 3.4 : Tableau de visualisation de données mesurées en temps réel

Ce tableau s'intitule 'Weather Station' faisant référence à la station de météo. Il présente six (08) widgets qui s'actualisent chaque minute. Parmi les widgets nous avons :

- **Widget 1** : un graphe de la température en degré Celsius (figure 3.5)



Figure 3.5 : Graphe d'évolution de la température

Ce graphe présente la variation de la température du **17/11/2020** à 00h00 (UTC) au **23/11/2020** à 12h00 (UTC). On observe une température minimale de 24°C et une température maximale 29°C.

## CHAPITRE 3 : RÉSULTATS ET INTERPRÉTATIONS

Ces données sont consultées depuis notre base de données Influx db. Pour obtenir ce graphe nous avons exécuté la requête décrite sur la figure (3.6).



Figure 3.6 : Requête pour interroger la base de données par rapport à la température actuelle

L’interprétation structurelle de la requête à la figure (3.6) est :

**SELECT LAST (« Temperature C ») FROM DATA GROUP BY TIME (1m)**

Où “Temperature c” ici est une colonne ou encore un champ (field\_1) de notre table Data

Cela nous permet de récupérer la dernière valeur de ce champ.

- **Widget 2 : Température prédictive**

La figure (3.7) illustre la visualisation de la température moyenne prédictive de chapelle Nsiméyong chaque heure. Cela correspond au field\_6 (forecast) dans notre base de données. Ce widget s’actualise chaque minute.

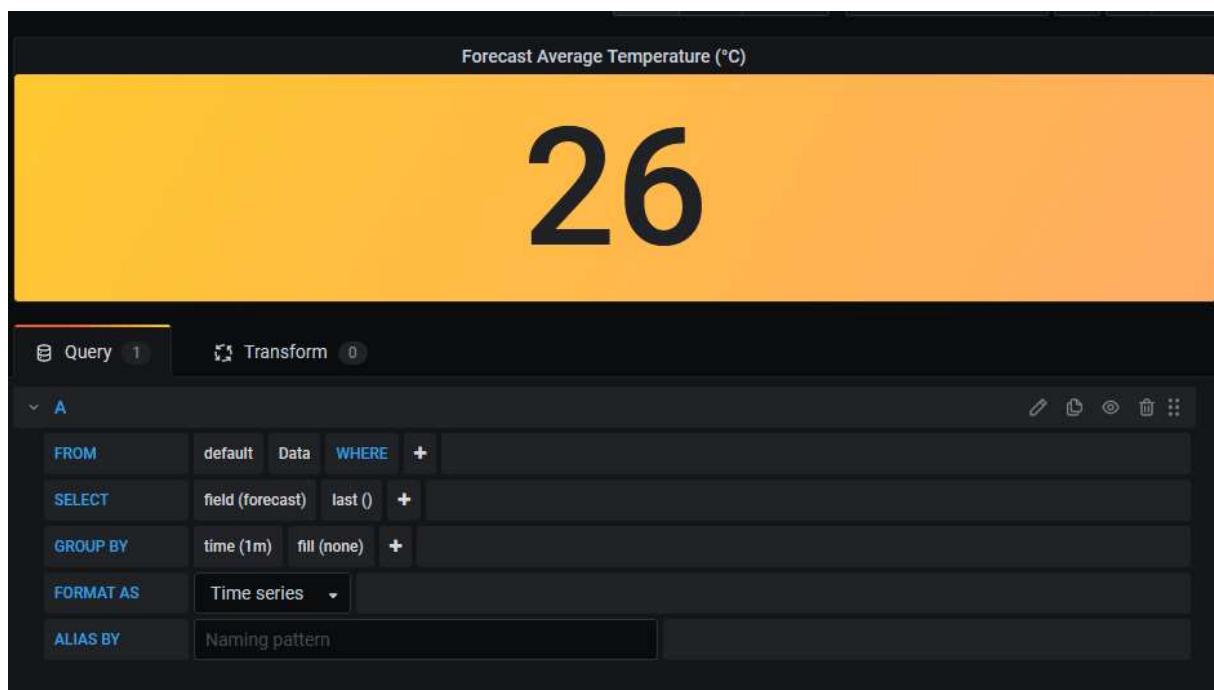


Figure 3.7: Température moyenne prédictive arrondi par défaut

## CHAPITRE 3 : RÉSULTATS ET INTERPRÉTATIONS

---

L’interprétation structurelle de la requête est :

**SELECT LAST (« forecast ») FROM DATA GROUP BY TIME (1m)**

- **Widget 3** : Température minimale en chiffre arabe

La figure (3.8) illustre la température minimale quotidienne de chapelle Nsimeyong

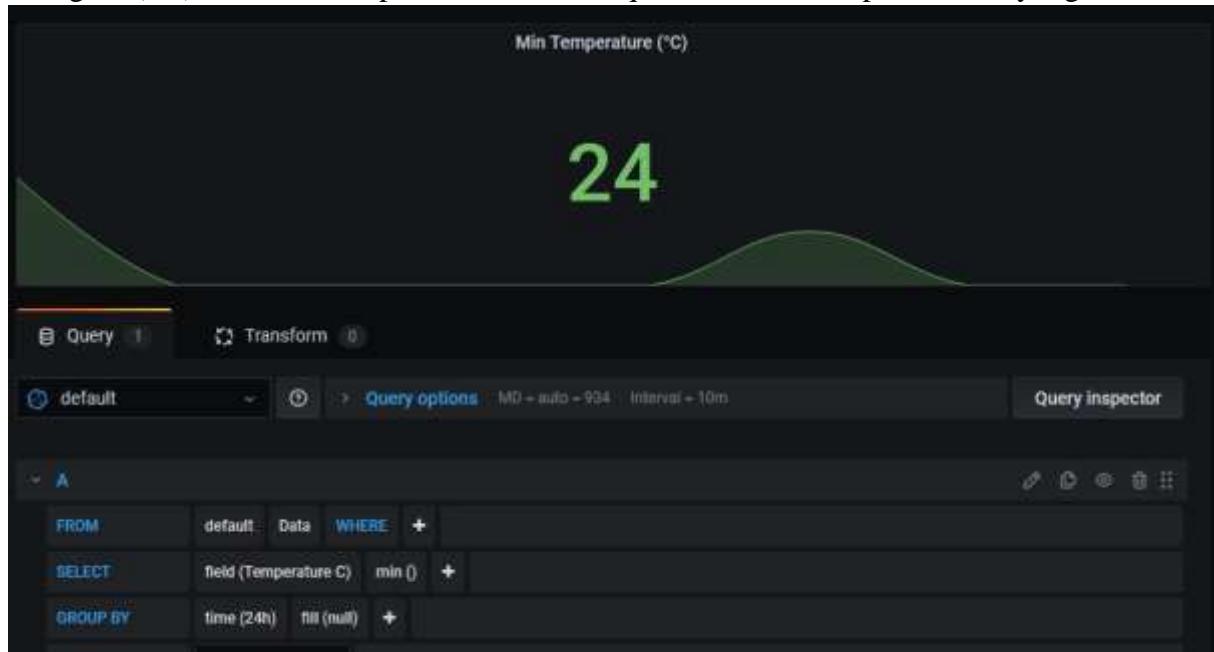


Figure 3.8 : Température minimale quotidienne de chapelle Nsimeyong

L’interprétation structurelle de la requête est :

**SELECT MIN (« Temperature C ») FROM DATA GROUP BY TIME (24h)**

- **Widget 4** : Température maximale en chiffre arabe

La figure (3.9) présente la visualisation de la température maximale de la journée.

## CHAPITRE 3 : RÉSULTATS ET INTERPRÉTATIONS



Figure 3.9 : Température maximale journalière de Chapelle Nsiméyong

L’interprétation structurelle de la requête est :

**SELECT MAX (« Temperature C ») FROM DATA GROUP BY TIME (24h)**

- **Widget 5 : la vitesse du vent (km/h)**



Figure 3.10 : Vitesse du vent de chapelle Nsiméyong temps réel

Ce graphe permet d’apprécier la variation de la vitesse du vent en temps réel (voir figure 3.10)

L’interprétation structurelle de la requête est :

### SELECT LAST (« WindSpeedMax ») FROM DATA GROUP BY TIME (1m)

- **Widget 6** : La direction du vent par rapport au nord magnétique (voir figure 3.11)

Ce paramètre n'a pas été utilisé dans notre modèle de prédition mais grâce à notre anémomètre nous pouvons prélever cette information. Par conséquent, nous pouvons la sauvegarder et la visualiser.

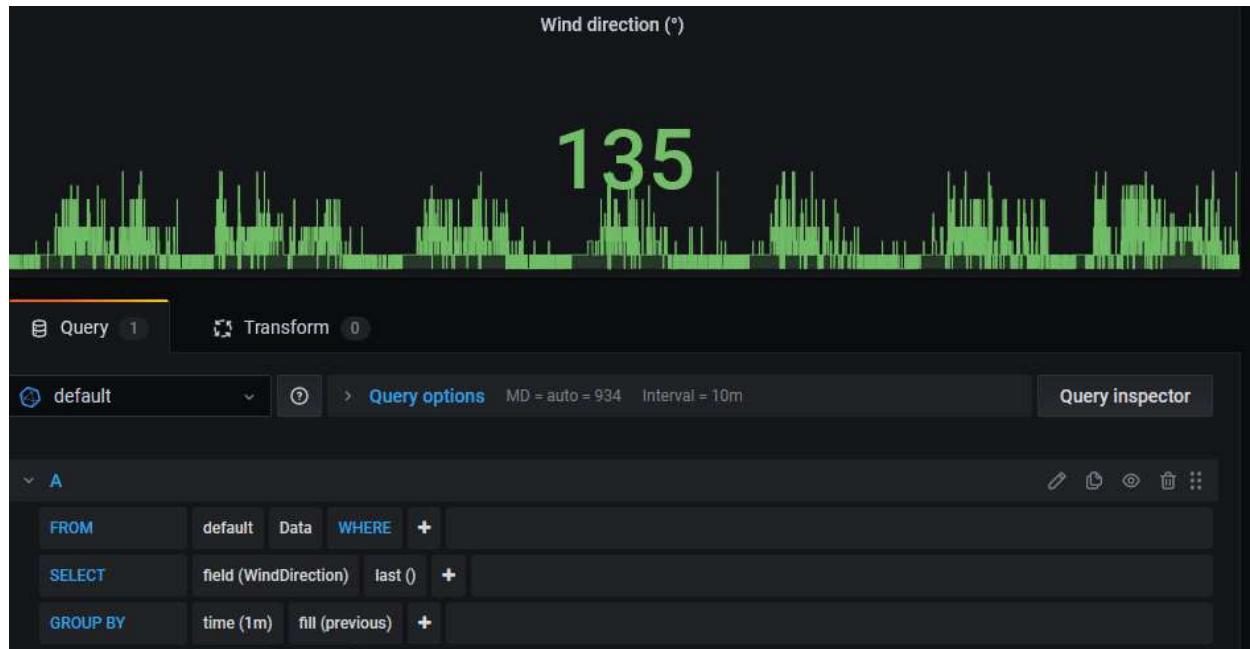


Figure 3.11 : Direction du vent de chapelle Nsimeyong en temps réel

L'interprétation structurelle de la requête est :

### SELECT LAST (« WindDirection ») FROM Data GROUP BY time (1m)

- **Widget 7** : L'humidité relative (%) voir figure (3.12)



Figure 3.12 : Widget de l'humidité relative

## CHAPITRE 3 : RÉSULTATS ET INTERPRÉTATIONS

Nous pouvons tracer la courbe d'observation de l'humidité relative grâce aux données collectées. Ceci est illustré par la figure (3.13) :

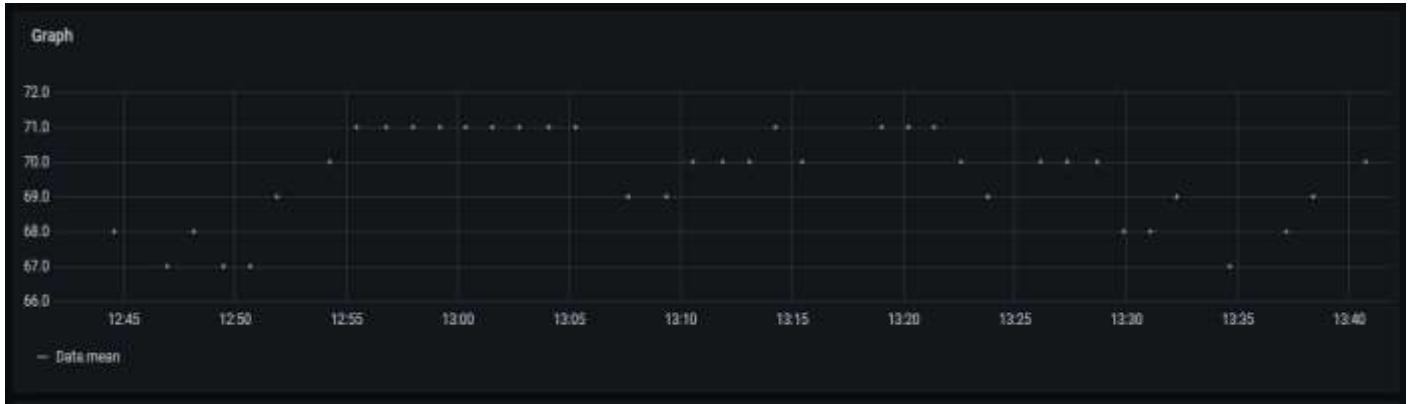


Figure 3.13 : Humidité relative de la journée du 23 novembre 2020 à chapelle Nsiméyong

La requête utilisée est la suivante :

```
SELECT LAST (<> Humidity %>) FROM Data GROUP BY time (1m)
```

- **Widget 8 :** Le niveau de précipitation (mm) figure (3.14)



Figure 3.14 : Widget du niveau de précipitation

Nous avons démarré la collecte des données de pluviométrie lorsque la saison pluvieuse de la ville de Yaoundé était terminée, ce qui explique que ce graphe soit nulle à tout instant (Figure 3.15).



Figure 3.15 : Niveau de précipitation

La requête utilisée est la suivante :

```
SELECT LAST (« RainFalloneHour ») FROM Data GROUP BY time (1m)
```

### 3.1. Modèle de prédition

#### 3.1.1. Rappel de l'objectif

Il s'agit du cœur de notre travail qui consiste à construire et à optimiser un modèle de prédition météorologique pour la ville de Yaoundé.

#### 3.1.2. Présentation du modèle de prédition

Notre modèle peut être considéré comme une boite noire comme le montre la figure (3.16) qui prend en entrée un vecteur X et donne en zone sortie un Y.

X est un vecteur de dimension 7x5 où 7 représente les 7 jours qui précédent le jour à prédire et 5 renvoie de manière respectivement à la température maximale, température minimale et les moyennes des précipitations, de la vitesse du vent et de l'humidité relative associés aux 7 jours en question.

Y renvoie à la température moyenne prédite du jour suivant.

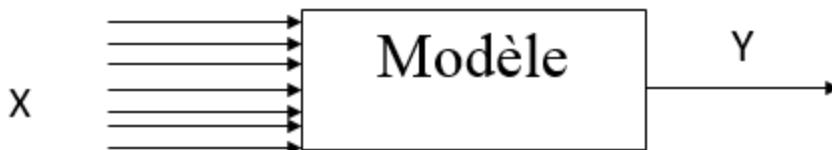


Figure 3.16: Vue d'ensemble du modèle

Le concept de boite noire évoqué plus haut n'est qu'une vue d'ensemble de notre solution. En réalité, il est question d'un réseau de neurones à quatre (04) couches :

## CHAPITRE 3 : RÉSULTATS ET INTERPRÉTATIONS

---

- La couche d'entrée
- La couche Hidden1 : la couche bidirectionnelle LSTM avec 2048 unités
- La couche Hidden2 : la couche LTSM avec 64 unités
- La couche de sortie : la couche dense avec 01 unité

La figure (3.17) présente un réseau sommaire de notre modèle

```
Model: "functional_1"
```

Layer (type)	Output Shape	Param #
<hr/>		
input_1 (InputLayer)	[None, 7, 5]	0
bidirectional (Bidirectional (None, 7, 2048)		8437760
lstm_1 (LSTM)	(None, 64)	540928
dense (Dense)	(None, 1)	65
<hr/>		
Total params:	8,978,753	
Trainable params:	8,978,753	
Non-trainable params:	0	

---

Figure 3.17 : Résumé des paramètres du modèle de prédiction

Notre modèle a besoin de 8,978,753 de neurones pour fournir la prédiction moyenne.

Pour entraîner ce modèle nous avons utilisé une machine virtuelle de Google Colaboratory avec les caractéristiques de 16 Gb de RAM et ‘TensorFlow Processing Unit’ (TPU).

La figure (3.18) illustre le réseau de neurones que nous avons construit selon les couches indiquées ci-haut.

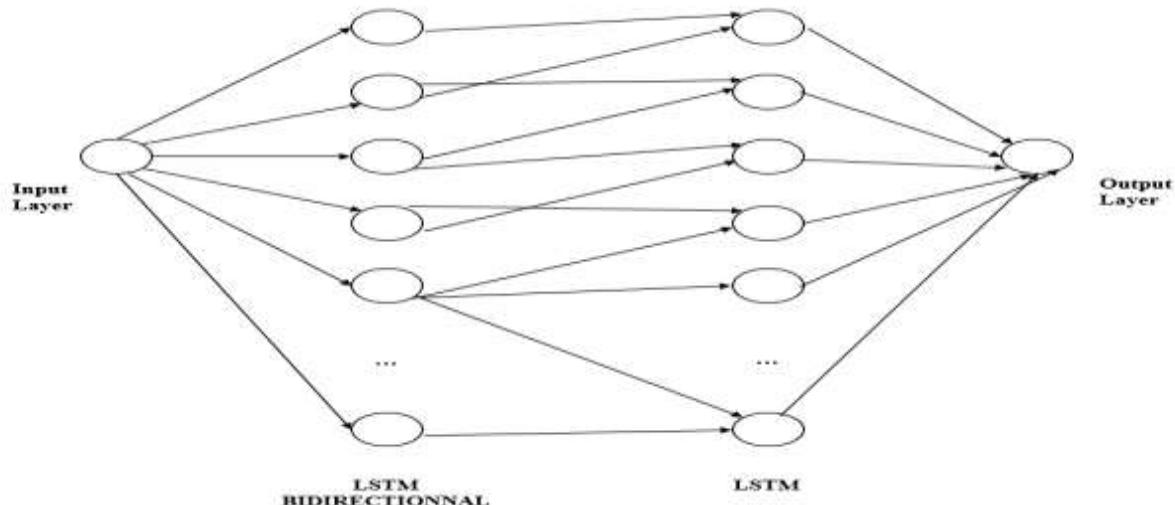


Figure 3.18 : Perceptron multicouche

Les résultats suivants sont simulés pour une prédiction à court terme, dont un intervalle d'un jour :

La figure (3.19) décrit les pertes d'apprentissage et de validation de notre modèle en fonction du nombre d'itérations (Epoch).

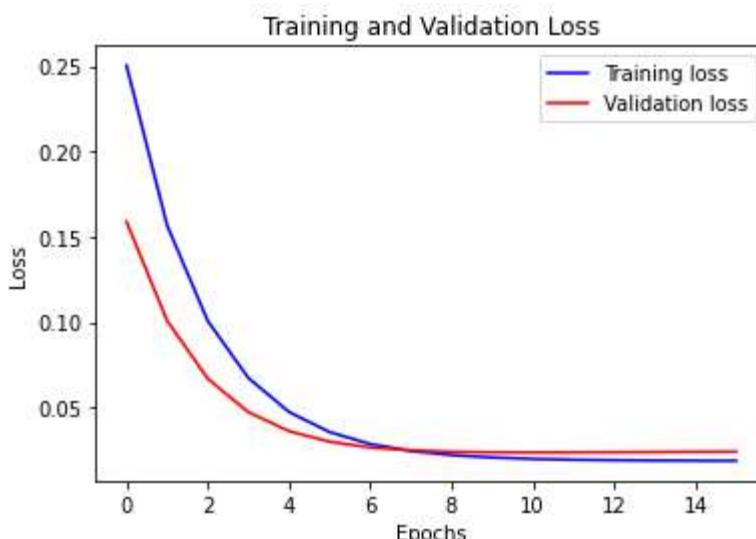


Figure 3.19 : Pertes d'apprentissage et de validation de notre modèle (01) jour

L'on relève ici :

- L'axe des abscisses renvoie au MSE tel décrit dans l'équation 22 ;
- Au bout de 14 itérations (Epochs) notre système cesse d'apprendre ;
- A partir de la 6<sup>eme</sup> itération (Epochs) les pertes de données d'entraînement et les pertes de données test convergent ;

- Notre système a un MSE < **0.05** soit donc < **5%**

Nous avons mené une comparaison entre les différentes métriques. La figure (3.20) présente les courbes évolutives des métriques RMSE et MAE soient l'équation 24 et l'équation 23.

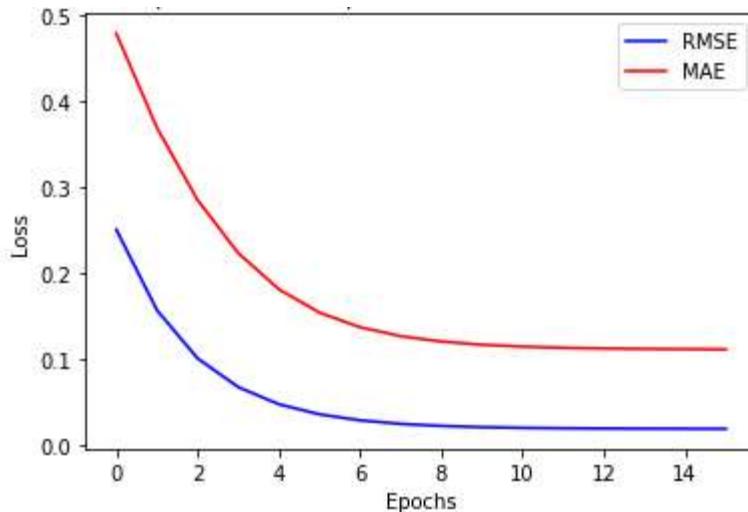


Figure 3.20 : Courbes évolutives RMSE et MAE prédition (01) jour

Notons que plus le RMSE et MAE sont proches de zéro plus le modèle est précis.

On observe que :

- RMSE converge plus vite que MAE
- RMSE = MSE = 0.0188
- MAE = 0.1111

### 3.1.3. Discussion

L'analyse comparative de performances entre notre modèle et ceux de notre revue de la littérature est contenue dans le tableau (3.4). Nous n'allons qu'apprécier les modèles qu'à travers le RMSE calculé sur la base de nos données. Pour une meilleure appréciation faudrait tester ces différents modèles avec des données génériques (BenchMark) de un et de deux prédire le même paramètre climatique. Qu'à cela ne tient, nous notons que notre modèle présente un taux d'erreur quadratique moyenne assez bas;

## CHAPITRE 3 : RÉSULTATS ET INTERPRÉTATIONS

---

Tableau 3.4 : Comparaison des modèles de prédiction à court terme (01) jour

	Notre modèle	Modèle de Balamurugan	Modèle de Yen
RMSE	0,0188	0,1126	1,69

Pour mieux apprécier ce résultat, la figure (3.21) présente un histogramme de comparaison de modèles. On observe que notre modèle a le plus petit RMSE des (03) modèles.

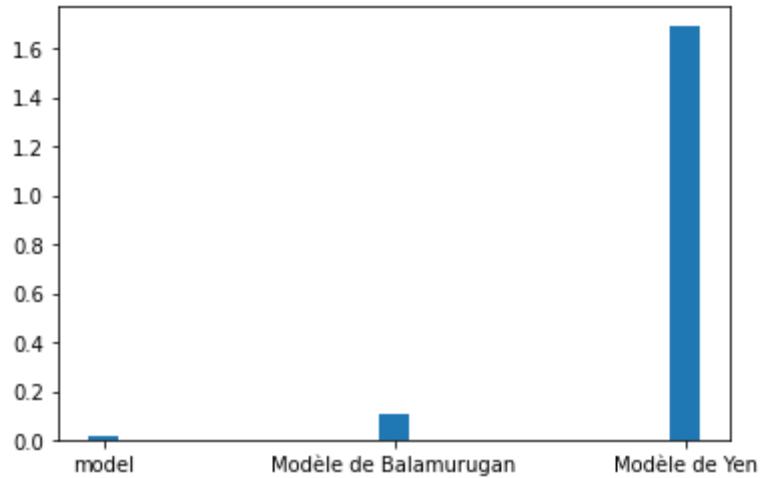


Figure 3.21 : Graphe de comparaison des performances de modèles

Nous avons mené une analyse des résultats de prédictions du 28 novembre 2020 au 05 décembre 2020 (soit sur une durée d'une semaine). Cette analyse compare nos prédictions avec celles de l'ONACC et celles de l'institution américaine spécialisée dans les prévisions météorologiques (AccuWeather).

## CHAPITRE 3 : RÉSULTATS ET INTERPRÉTATIONS

---

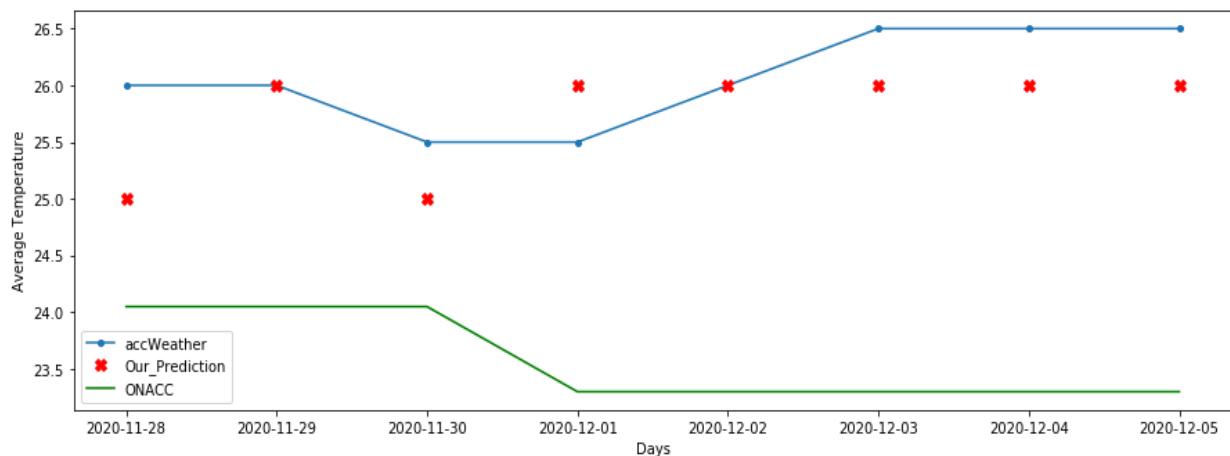


Figure 3.22 : Comparaison des prédictions quotidiennes

Les données utilisées sont dans les annexes. Il est important de noter que les prédictions faites par l'ONACC du 28 novembre 2020 au 30 décembre 2020 se basent sur les observations du 11 au 20 novembre 2020 et des données historiques de 1979 à 2018. De même que les prédictions du 01<sup>er</sup> au 05 décembre 2020 sont basées sur les observations du 21 au 30 novembre 2020 et des mêmes données historiques.

La figure (3.22) révèle que :

- Notre modèle de prédiction se rapproche plus des prédictions effectuées par ‘AccuWeather’ que celles de l’ONACC ;
- Les prédictions de l’ONACC sont moins précises que notre modèle ;
- On remarque également que notre modèle ne produit pas de valeurs décimales de température moyenne car toutes les valeurs sont arrondies par défaut ;
- En tenant compte des arrondis par défaut de la température moyenne de ‘AccuWeather’, 75 % de correspondances comme l’indique le tableau (3.5).

## CHAPITRE 3 : RÉSULTATS ET INTERPRÉTATIONS

---

Tableau 3.5 : Comparaison entre notre solution et AccuWeather

Jours	Nos prédictions (°C)	Prédictions d'AccuWeather (°C)	Prédictions d'AccuWeather à valeurs arrondies (°C)
28-11-2020	25	26	26
29-11-2020	26	26	26
30-11-2020	25	25,5	25
01-12-2020	26	25,5	25
02-12-2020	26	26	26
03-12-2020	26	26,5	26
04-12-2020	26	26,5	26
05-12-2020	26	26,5	26

L'évaluation des performances de notre modèle pour les prédictions aux intervalles de 05, 10 et 15 jours, nous donne les résultats du tableau (3.6).

Tableau 3.6 : Evaluation des performances du modèle aux intervalles de 05,10 et 20 jours

Jours	05	10	20
RMSE	0,0188	0,0203	0,02146

L'on observe que les performances diminuent à partir du 10<sup>ème</sup> jour où elles sont à 0.0203 de précision et au 20<sup>ème</sup> jour à 0,02146.

Nous apprécions la qualité de précision sur la prédiction du 10<sup>ème</sup> jour. La figure (3.23) présente les pertes d'entraînement et de validation et la figure (3.24) présente (05) prédictions effectuées avec ce modèle.

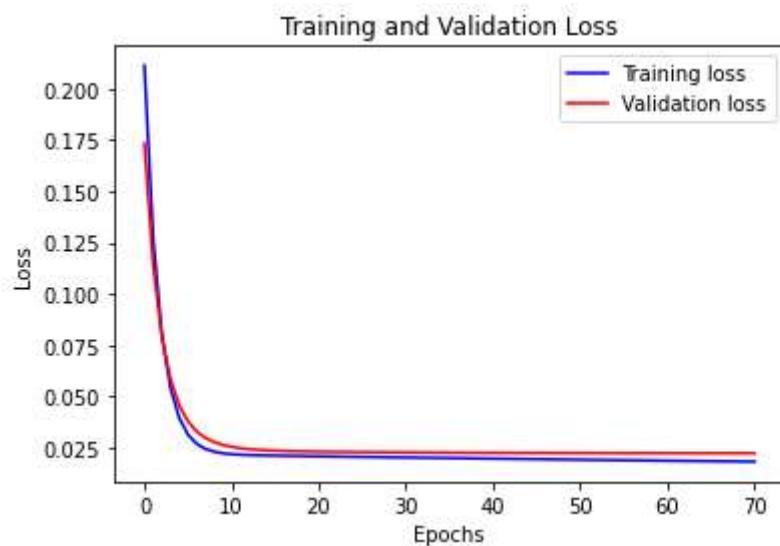
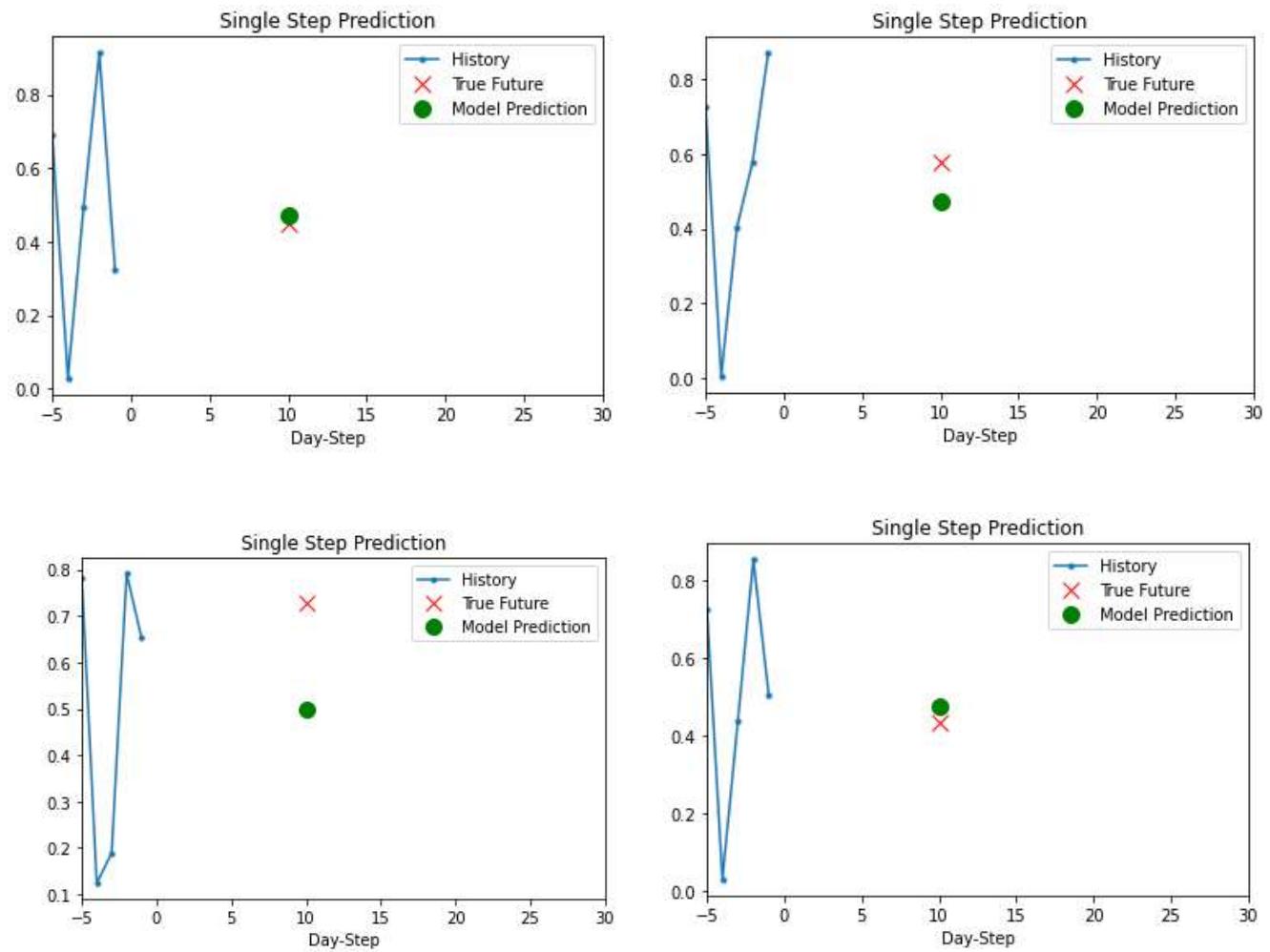


Figure 3.23 : Pertes d'entraînement et de validation des prédictions du 10<sup>ème</sup> jour



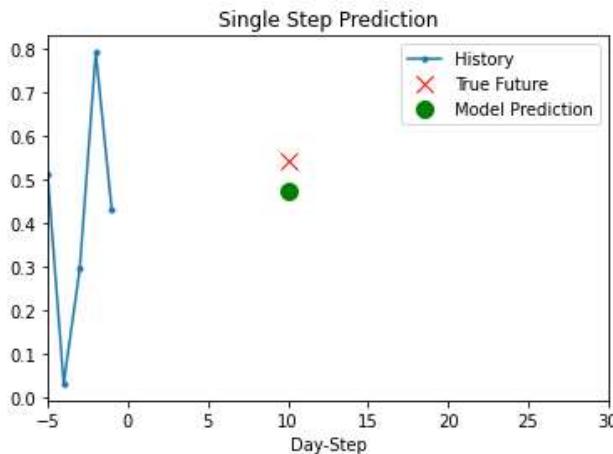


Figure 3.24 : (05) prédictions effectuées avec les prédictions à 10 jours

L'on observe que parmi les 05 prédictions :

- nous avons 01 prédition dont l'erreur est significative ( $> 0.25$ ) ;
- 03 autres prédictions avec des erreurs moins significatives ( $< 0.2$ ) ;
- une seule prédition correcte.

Toutefois ces résultats, le RMSE du modèle 10<sup>ème</sup> égale à 0.0203 reste toujours performants.

### 3.2. Estimation de la solution

Le déploiement d'une telle solution requiert tout d'abord l'acquisition des outils présentés au chapitre 2. Nous ajoutons à cela des frais liés à la conception.

Le tableau (3.7) contient l'ensemble des coûts.

Tableau 3.7 : Estimation du prix de la solution

Désignation	Prix unitaire (Fcfa)	Délai	Prix (Fcfa)
Location de la station météo avec ses différents capteurs	100 000 / mois	12 mois	<b>1 200 000</b>
Licence d'utilisation	500 000 / an	1 an	<b>500 000</b>
Implémentation	25 000	/	<b>25 000</b>
<b>TOTAL</b>			<b>1 725 000</b>

La première année, il faudra débourser **1 725 000** Fcfa pour s'offrir notre solution puis l'année suivante, on est dispensé de frais d'implémentation.

## Conclusion

Dans ce chapitre nous avons présenté l'ensemble des fonctionnalités de notre solution intelligente de prédiction de la température moyenne. Pour l'évaluation des performances, nous avons de prime abord comparé la qualité de nos données à ceux du jeu de données de départ et par la suite comparé le taux de précision des prédictions avec ceux de la littérature. Enfin nous avons estimé un prix à notre solution.

## **CONCLUSION GÉNÉRALE**

## CONCLUSION GÉNÉRALE

---

Notre travail était focalisé sur l’élaboration d’un processus de collecte et de prédition météorologique basé sur les technologies de l’internet des objets et de Machine Learning afin d’optimiser la précision des données prédictes. Nous nous sommes donnés pour mission de concevoir un outil intelligent capable de faire des prédictions d’un jour de la température moyenne. Il était également question de doter cet outil d’une base de données de stockage et d’un tableau de visualisation en temps réel.

Pour mener à bien notre objectif, nous avons tout d’abord présenté les méthodes actuelles de prédictions météorologiques et leurs principes. Ensuite, nous avons défini le concept de Machine Learning et présenté quelques algorithmes supervisés, non supervisés et le réseau de neurones.

En se nourrissant d’anciens documents (articles et livres) portant sur des thématiques similaires, nous avons développé un pipeline de travail. Il nous a permis de modéliser un flux de travail bien élaboré et adapté à l’utilisation du réseau de neurones par récurrence et de son algorithme LSTM.

Pour mettre en pratique nos recherches, nous avons eu besoin d’une station de mesure (Arduino et des capteurs analogiques) et d’une station de base (Raspberry pi 4 et le capteur numérique DHT11). De là nous avons développé en langage Python une application qui permet de collecter et de stocker les données de la station de base, de les visualiser en temps réel et enfin d’effectuer des prévisions d’un jour de la température moyenne.

Cependant, nous avons rencontré la difficulté d’obtenir un jeu de données de départ assez récent. Malgré cela, nous avons obtenu un RMSE égal à 0,0188 soit inférieur à 5 %. Quelques suggestions pour les travaux de recherches futurs peuvent être données. C’est ainsi qu’on pourra envisager prédire le niveau de précipitation pendant les saisons pluvieuses ou encore évaluer le risque de pluie.

# ANNEXES

## Données de comparaisons

### VI. Evaluation des prévisions climatiques de la décennie du 21 au 30 novembre 2020

Zones Agro-écologiques	Soudano-Sahélienne		Hautes Savanes Guinéennes	Forestière à pluviométrie Bimodale			Hauts Plateaux		Forestière à pluviométrie Monomodale	
	Régions	Ext-Nord	Nord	Adamaoua	Est	Centre	Sud	Ouest	Nord-Ouest	Sud-Ouest
<b>Températures minimales</b>										
Moyenne Historique de 1979 à 2018 (°C)	19	19,4	17	19,4	18,6	20,3	15,6	15,2	20,6	21
Prévision des tendances	↘	↘	↘	↘	↘	↘	↘	↘	↘	↘
Taux de réussite des prévisions (%)	79	78	80	80	78	79	78	80	79	78
<b>Températures maximales</b>										
Moyenne historique de 1979 à 2018 (°C)	35,1	35,4	32,93	28	28	28,66	27,46	28	28,35	28,42
Prévision des tendances	↗	↗	≈	≈	≈	≈	↗	↗	↗	↗
Taux de réussite des prévisions (%)	77	81	79	78	80	78	80	75	79	78
<b>Précipitations</b>										
Moyenne Historique de 1979 à 2018 (mm)	23-53	54-85	86-176	177-179	180-202	203-234	234-241	203-238	233-262	182-204
Prévision des tendances	≈	≈	≈	≈	≈	≈	≈	≈	≈	≈
Taux de réussite des prévisions (%)	77	91	98	100	100	100	98,6	100	100	100

Autour de la moyenne ≈ ; ↘ = Diminution ; ↗ = Augmentation

### VI. Evaluation des prévisions climatiques de la décennie du 21 au 30 novembre 2020

Zones Agro-écologiques	Soudano-Sahélienne		Hautes Savanes Guinéennes	Forestière à pluviométrie Bimodale			Hauts Plateaux		Forestière à pluviométrie Monomodale	
	Régions	Ext-Nord	Nord	Adamaoua	Est	Centre	Sud	Ouest	Nord-Ouest	Sud-Ouest
<b>Températures minimales</b>										
Moyenne Historique de 1979 à 2018 (°C)	19	19,4	17	19,4	18,6	20,3	15,6	15,2	20,6	21
Prévision des tendances	↘	↘	↘	↘	↘	↘	↘	↘	↘	↘
Taux de réussite des prévisions (%)	79	78	80	80	78	79	78	80	79	78
<b>Températures maximales</b>										
Moyenne historique de 1979 à 2018 (°C)	35,1	35,4	32,93	28	28	28,66	27,46	28	28,35	28,42
Prévision des tendances	↗	↗	≈	≈	≈	≈	↗	↗	↗	↗
Taux de réussite des prévisions (%)	77	81	79	78	80	78	80	75	79	78
<b>Précipitations</b>										
Moyenne Historique de 1979 à 2018 (mm)	23-53	54-85	86-176	177-179	180-202	203-234	234-241	203-238	233-262	182-204
Prévision des tendances	≈	≈	≈	≈	≈	≈	≈	≈	≈	≈
Taux de réussite des prévisions (%)	77	91	98	100	100	100	98,6	100	100	100

Autour de la moyenne ≈ ; ↘ = Diminution ; ↗ = Augmentation

## ANNEXES

AccuWeather Yaoundé, Centre 29%							
novembre	2020	QUOTIDIEN →					
D	L	M	M	J	V	S	S
1	2	3	4	5	6	7	
27° 20°	28° 20°	30° 21°	29° 20°	30° 19°	30° 19°	28° 21°	
8	9	10	11	12	13	14	
28° 21°	29° 19°	29° 20°	30° 19°	31° 20°	30° 20°	29° 21°	
15	16	17	18	19	20	21	
30° 21°	29° 21°	30° 21°	32° 19°	31° 20°	32° 21°	31° 21°	
22	23	24	25	26	27	28	
31° 21°	30° 21°	30° 21°	31° 21°	31° 21°	31° 21°	31° 21°	
29	30	1	2	3	4	5	
32° 20°	31° 20°	31° 20°	31° 21°	31° 22°	32° 21°	31° 22°	

### Programme du Raspberry pour la prédiction

```
1 import argparse
2 import pandas as pd
3 import numpy as np
4 from sklearn.preprocessing import MinMaxScaler
5 from influxdb import DataFrameClient
6 from keras.models import load_model
7 from influxdb import InfluxDBClient
8 import time
9
10
11 #connexion bd
12 client_2 = InfluxDBClient(host='localhost', port=8086, username='admin', password='admin')
13 client_2.switch_database('station1')
14
15 #chargement du modèle
16 print("Importation du modèle")
17 model = load_model('/home/pi/PFE/my_model')
18 #model.summary()
19
20 def main(host='localhost', port=8086):
21
22     user = 'admin'
23     password = 'admin'
24     dbname = 'station1'
```

## ANNEXES

```
while (True):
    #requete influxdb
    min_temp = client.query('SELECT MIN("Temperature C") FROM "Data" WHERE time > now() - 60m')
    max_temp = client.query('SELECT MAX("Temperature C") FROM "Data" WHERE time > now() - 60m')
    rainfallday = client.query('SELECT MEAN("RainFallDay") FROM "Data" WHERE time > now() - 60m')
    wind = client.query('SELECT MEAN("wind") FROM "Data" WHERE time > now() - 60m')
    mean_hum = client.query('SELECT MEAN("Humidity %") FROM "Data" WHERE time > now() - 60m')
    temperature=client.query('SELECT "Temperature C" FROM "Data"')

    #print(temperature)

    # création dataframe
    df=max_temp['Data']
    df['min']=min_temp['Data']
    df['precipitation']=rainfallday['Data']
    df['wind']=wind['Data']
    df['humidity']=mean_hum['Data']*0.01
    print(df)
```

## Exemple de prédiction

```

max   min   precipitation   wind   humidity
2020-12-01 00:00:00+00:00 28.0 26.0          0.0 0.678166 0.597904
2020-12-02 00:00:00+00:00 28.0 24.0          0.0 0.859474 0.685277
2020-12-03 00:00:00+00:00 28.0 24.0          0.0 0.597933 0.679163
2020-12-04 00:00:00+00:00 28.0 24.0          0.0 0.611577 0.688978
2020-12-05 00:00:00+00:00 27.0 25.0          0.0 0.453124 0.680066
2020-12-06 00:00:00+00:00 27.0 25.0          0.0 0.149303 0.741070
2020-12-07 00:00:00+00:00 27.0 25.0          0.0 0.251508 0.750855
[[1.      1.      0.      0.74469804 0.      ],
 [1.      0.      0.      1.      0.57124695],
 [1.      0.      0.      0.63172089 0.5312779],
 [1.      0.      0.      0.6509332 0.59544452],
 [0.      0.5     0.      0.42781389 0.53718097],
 [0.      0.5     0.      0.      0.93602947],
 [0.      0.5     0.      0.14391594 1.      ],
 [[26.217306]]]

```

## RÉFÉRENCES

Abrahamsen, E. B., et al. (2018). "Machine learning in python for weather forecast based on freely available weather data."

Balamurugan, M. et R. J. W. N. Manojkumar (2019). "Study of short term rain forecasting using machine learning based approach." 1-6.

Bauer, P., et al. (2015). "The quiet revolution of numerical weather prediction." **525**(7567): 47-55.

Bell, J. (2020). Machine learning: hands-on for developers and technical professionals, John Wiley & Sons.

Brown, R. J. A. (1984). "Absorption and scattering of light by small particles." **31**(1): 3.

Chai, T. et R. R. J. G. m. d. Draxler (2014). "Root mean square error (RMSE) or mean absolute error (MAE)?—Arguments against avoiding RMSE in the literature." **7**(3): 1247-1250.

Cresswell, M. J. T. G. J. (2009). "Empirical Methods in Short-term Climate Prediction." **175**: 85.

Cunningham, P. et S. J. J. a. p. a. Delany (2020). "k-Nearest Neighbour Classifiers."

Cutler, A., et al. (2012). Random forests. Ensemble machine learning, Springer: 157-175.

Dey, A. (2016). Machine Learning Algorithms : A Review.

Géron, A. (2017). Hands-On Machine Learning with Scikit-Learn and TensorFlow, O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472.

Hewamalage, H., et al. (2020). "Recurrent neural networks for time series forecasting: Current status and future directions."

Iseh, A., et al. (2013). "Weather forecasting models, methods and applications." **2**(12): 1945-1956.

Jakaria, A., et al. (2020). "Smart weather forecasting using machine learning: a case study in tennessee."

Kestwal, M. C., et al. (2014). "Prediction of rain attenuation and impact of rain in wave propagation at microwave frequency for tropical region (Uttarakhand, India)." **2014**.

## RÉFÉRENCES

---

- Linacre, E. et B. Geerts (1997). Climates and weather explained, Routledge.
- Murphy, A. H. et E. S. J. M. w. r. Epstein (1989). "Skill scores and correlation coefficients in model verification." **117**(3): 572-582.
- Shrestha, A. et A. J. I. A. Mahmood (2019). "Review of deep learning algorithms and architectures." **7**: 53040-53065.
- Tsalefac, M., et al. (2015). "Climat de l'Afrique centrale: passé, présent et futur."
- Yadav, K. K. e. S. (2018). "Linear regression analysis study." Journal of the Practice of Cardiovascular Sciences.
- Yu, Z., et al. (2015). Using bidirectional LSTM recurrent neural networks to learn high-level abstractions of sequential features for automated scoring of non-native spontaneous speech. 2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU), IEEE.
- Zaytar, M. A. et C. J. I. J. o. C. A. El Amrani (2016). "Sequence to sequence weather forecasting with long short-term memory recurrent neural networks." **143**(11): 7-11.
- Abhishek Sharma, (2020), <https://www.analyticsvidhya.com/blog/2020/05/decision-tree-vs-random-forest-algorithm/> consulté le 25 octobre 2020
- Bastien L, (2019), <https://www.lebigdata.fr/reseau-de-neurones-artificiels-definition> consulté le 27 octobre 2020
- METEOTERM      Organisation      Météorologique      Mondiale      (OMM),  
<https://unterm.un.org/unterm/Display/record/WMO/NA?OriginalId=529c7cb4-db0e-4bd0-85b0-88582805b511> consulté le 01<sup>er</sup> novembre 2020.
- National Centers for Environmental Prediction (NCEP), <https://globalweather.tamu.edu/> consulté le 20 octobre 2020.
- Observatoire Nationale sur les Changements Climatiques (ONACC), (2016), Étude sur la définition des formats et protocole de collecte des données climatiques, 8-121.
- Observatoire Nationale sur les Changements Climatiques (ONACC), (2016), Étude sur l'état des lieux des sources d'informations en vue de la mise en place d'une plateforme multi-acteurs d'échanges sur les données climatiques, 9-80.

## RÉFÉRENCES

---

Observatoire Nationale sur les Changements Climatiques (ONACC), Alertes climatiques décennales et impacts probables pour la période allant du 21 au 30 novembre 2020, bulletin N° 63, 2-17.

Observatoire Nationale sur les Changements Climatiques (ONACC), Alertes climatiques décennales et impacts probables pour la période allant du 01<sup>er</sup> au 10 décembre 2020, bulletin N° 64, 2-17.

Organisation Météorologiques Mondiale, (2016), Les Métiers de la Météorologie, N°126, 4-35.

Roman Paolucci, (2020), <https://towardsdatascience.com/linear-regression-v-s-neural-networks-cd03b29386d4> consulté le 23 octobre 2020

Sanjay.M, (2018), <https://towardsdatascience.com/knn-using-scikit-learn-c6bed765be75> consulté le 23 octobre 2020.

Sweet A.I, (2020), <https://medium.com/@sweetai/a-brief-introduction-to-support-vector-machine-ba2ae5ea1d4e> consulté le 24 octobre 2020

Younes Benzaki, (2017), <https://mrmint.fr/logistic-regression-machine-Learning-introduction-simple> consulté le 24 octobre 2020

<https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.MinMaxScaler.html> consulté le 01 novembre 2020

<https://glossary.ametsoc.org/wiki/Climatology> consulté le 03 décembre 2020

<https://pimylifeup.com/raspberry-pi-influxdb/> consulté le 15 novembre 2020

<http://cactus.io/projects/weather/arduino-weather-station-davis-anemometer> consulté le 13 novembre 2020

<https://www.raspberrypi.org/blog/buster-the-new-version-of-raspbian/> consulté le 20 novembre 2020

<https://www.raspberrypi.org/products/raspberry-pi-4-model-b/> consulté le 21 novembre 2020

<https://store.arduino.cc/arduino-uno-rev3> consulté le 21 novembre 2020

<https://www.accuweather.com/fr/cm/yaounde/47601/november-weather/47601> consulté le 06 décembre 2020