



Gestion des données manquantes en thermique du bâtiment

MADANI Sarra Amel OUACHOUR Hanane

Sous la supervision de :
STOLL Yannick , DJATOUTI Zohra
et
PRUD'HOMME Christophe

Université de Strasbourg

- 1 Introduction
- 2 Rappels
- 3 Requêtes aléatoires
- 4 Test de l'algorithme d'imputation (Random Forest)
- 5 Réseaux de neurones récurrents
 - Modèle univarié
 - Modèle multivarié
- 6 Application des données au confort thermique
- 7 Conclusion

Introduction

Gestion des
données
manquantes
en thermique
du bâtiment

MADANI
Sarrah Amel
OUACHOUR
Hanane

Introduction

Rappels

Requêtes
aléatoires

Test de
l'algorithme
d'imputation
(Random
Forest)

Réseaux de
neurons
récurrents

Modèle univarié
Modèle multivarié

Application
des données
au confort
thermique

Conclusion



FIGURE – Transition énergétique des bâtiments [1]

Présentation du projet

Gestion des données manquantes en thermique du bâtiment

MADANI
Sarra Amel
OUACHOUR
Hanane

Introduction

Rappels

Requêtes aléatoires

Test de l'algorithme d'imputation (Random Forest)

Réseaux de neurones récurrents

Modèle univarié
Modèle multivarié

Application des données au confort thermique

Conclusion



FIGURE – Le bâtiment connecté du laboratoire ICube [2]

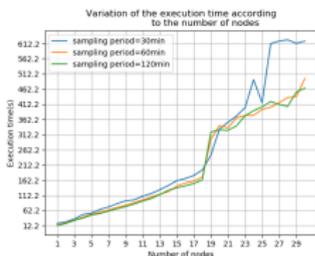


Le bâtiment étudié

- IOT du pôle API à Illkirch-Graffenstaden
- 71 capteurs
- 400 GB de données récoltées

- Étude de l'efficacité de l'algorithme d'imputation basé sur les Random Forest.
- Benchmarking du temps total d'exécution en appliquant une requête aléatoire des nœuds.
- Prédiction de la température à l'aide des réseaux de neurones récurrents.
- Faire une étude comparative des performances des deux algorithmes de Machine Learning (Random Forest et Réseaux de Neurones).
- Application du confort thermique sur les données de la BD.



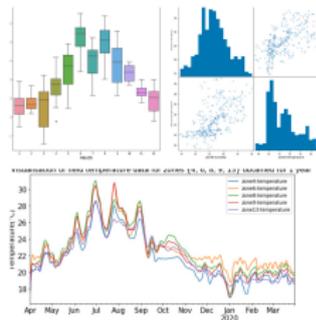


Etude de benchmarking

- les 30 premiers nœuds
- l'ensemble des zones
- les algorithmes d'imputation et de lissage basés sur les RF

Statistiques descriptives

- agrégation des données par zone
- étude statistique des séries temporelles



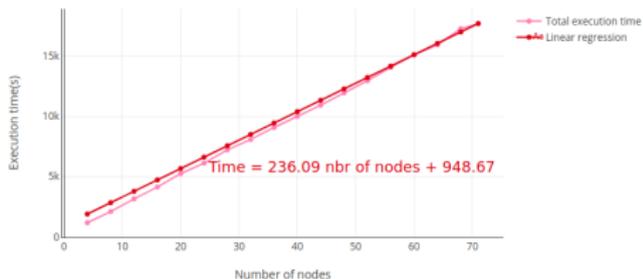


FIGURE – Représentation du temps total d'exécution en fonction du nombre de nœuds

Paramètres

- requêter jusqu'à 71 nœuds
- période d'échantillonnage de 60 min
- 1 année de données
- nombre d'exécution : 5

Requêtes aléatoires

Gestion des données manquantes en thermique du bâtiment

MADANI Sarra Amel OUACHOUR Hanane

Introduction

Rappels

Requêtes aléatoires

Test de l'algorithme d'imputation (Random Forest)

Réseaux de neurones récurrents

Modèle univarié
Modèle multivarié

Application des données au confort thermique

Conclusion

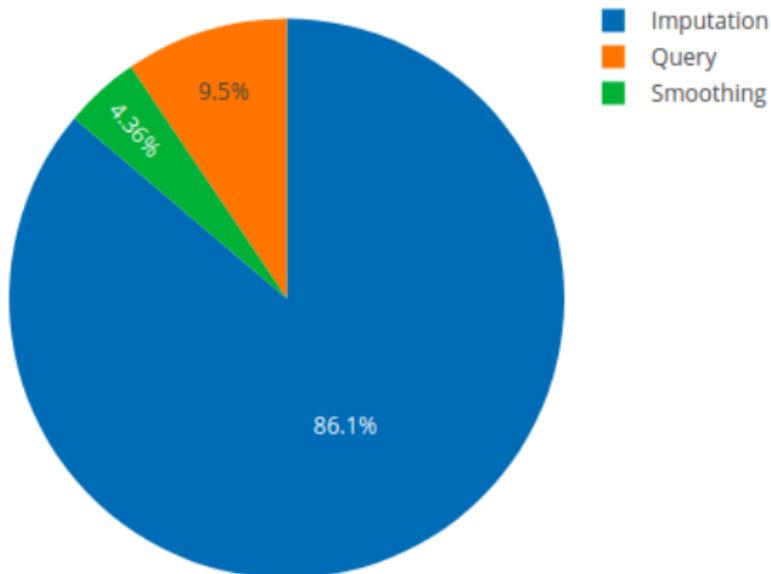


FIGURE – Graphique circulaire représentant les différents temps d'exécution

Principe général

Gestion des
données
manquantes
en thermique
du bâtiment

MADANI
Sarrah Amel
OUACHOUR
Hanane

Introduction

Rappels

Requêtes
aléatoires

Test de
l'algorithme
d'imputation
(Random
Forest)

Réseaux de
neurones
récurrents

Modèle univarié
Modèle multivarié

Application
des données
au confort
thermique

Conclusion

- Création de trous de données d'une marge de Δt
- Date de début fixée au 21 juin 2019 à 00h
- Imputation basée sur la méthode des Random Forest
- Utilisation d'observations passées et futures

Comparaison des valeurs réelles et des valeurs inférées

MADANI
Sarra Amel
OUACHOUR
Hanane

Introduction

Rappels

Requêtes aléatoires

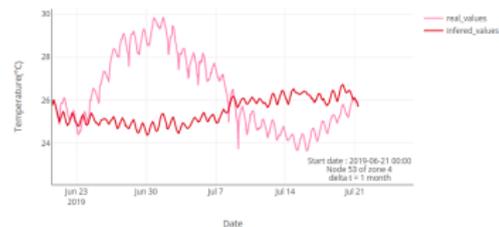
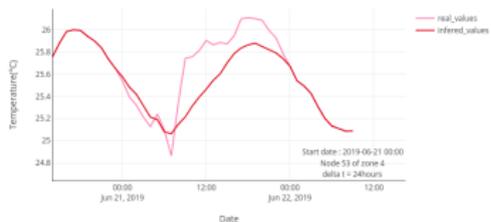
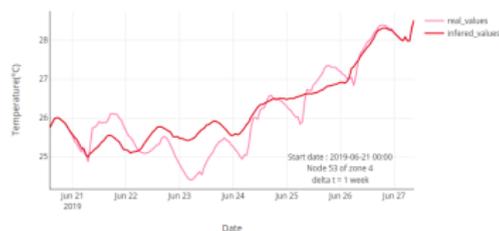
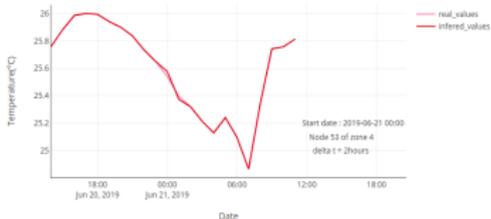
Test de l'algorithme d'imputation (Random Forest)

Réseaux de neurones récurrents

Modèle univarié
Modèle multivarié

Application des données au confort thermique

Conclusion



Valeurs réelles et inférées pour
 $\Delta(t) = \{2h, 1\text{journée}, 1\text{semaine}, 1\text{mois}\}$

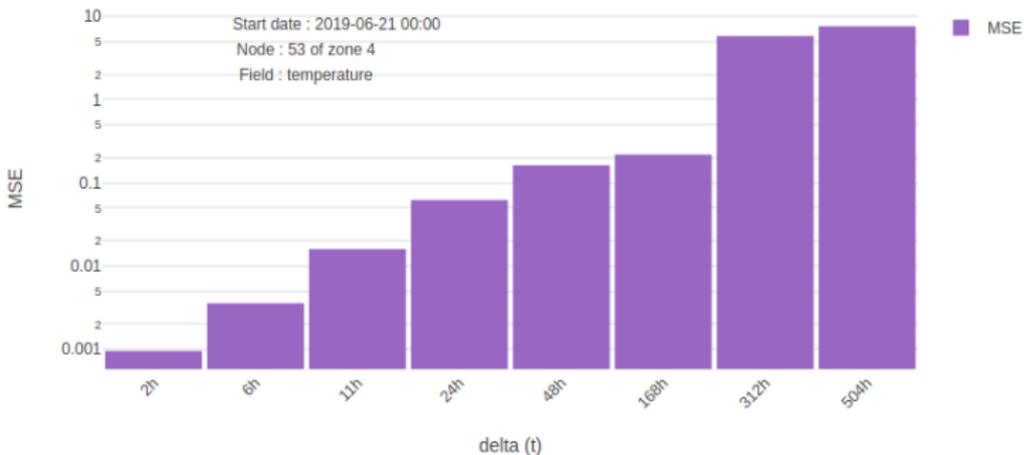


FIGURE – MSE en fonction de $\Delta(t)$

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

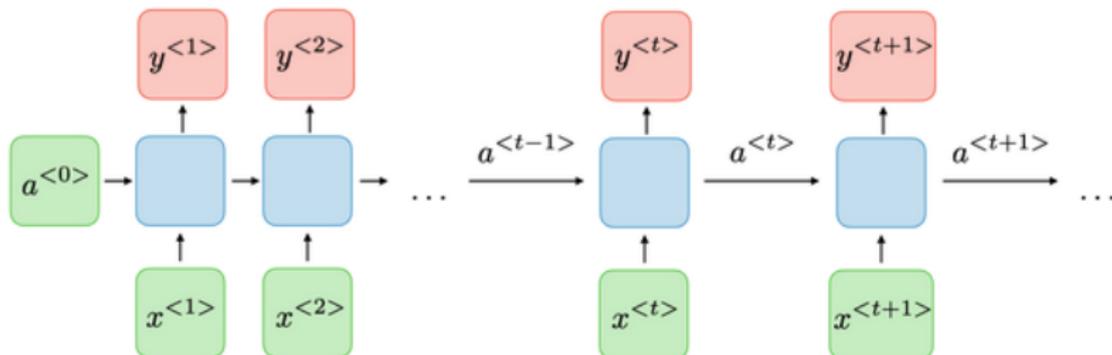


FIGURE – Réseau de neurones récurrents[3]

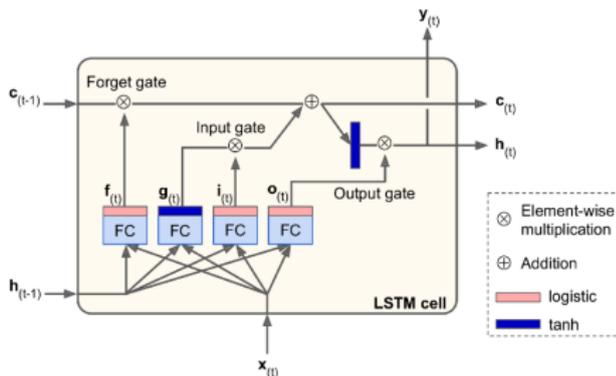


FIGURE – Cellule LSTM [4]

Concernant les LSTM

- méthode de Deep Learning
- la solution à la mémoire à court terme
- possèdent des portes qui régularisent le flux d'informations

Les paramètres utilisés pour le modèle univarié :

Taille des données

- températures du nœud 28 sur 2 mois (1441 observations)
- TRAIN-SPLIT : 1200 observations
- prédiction à l'instant t en exploitant les 20 dernières observations

```
TRAIN_SPLIT = 1200
univariate_past_history = 20
univariate_future_target = 0

x_train_uni, y_train_uni = univariate_data(uni_data, 0, TRAIN_SPLIT,
                                           univariate_past_history,
                                           univariate_future_target)
x_val_uni, y_val_uni = univariate_data(uni_data, TRAIN_SPLIT, None,
                                       univariate_past_history,
                                       univariate_future_target)
```

Configuration du modèle

- batch size : 32, buffer size : 1000
- 1 couche LSTM en entrée, et une couche en sortie
- optimiseur : SGD, fonction de perte : RMSE
- nombre d'epochs : 20

```
simple_lstm_model = tf.keras.models.Sequential([  
    tf.keras.layers.LSTM(8, input_shape=x_train_uni.shape[-2:]),  
    tf.keras.layers.Dense(1)])
```

```
simple_lstm_model.compile(optimizer='SGD', loss=root_mean_squared_error)
```

- batch size : nombre d'exemples de formation utilisés dans une itération
- buffer size : nombre maximum d'éléments mis en mémoire tampon lors de la pré-lecture
- epoch : un passage sur l'ensemble des données

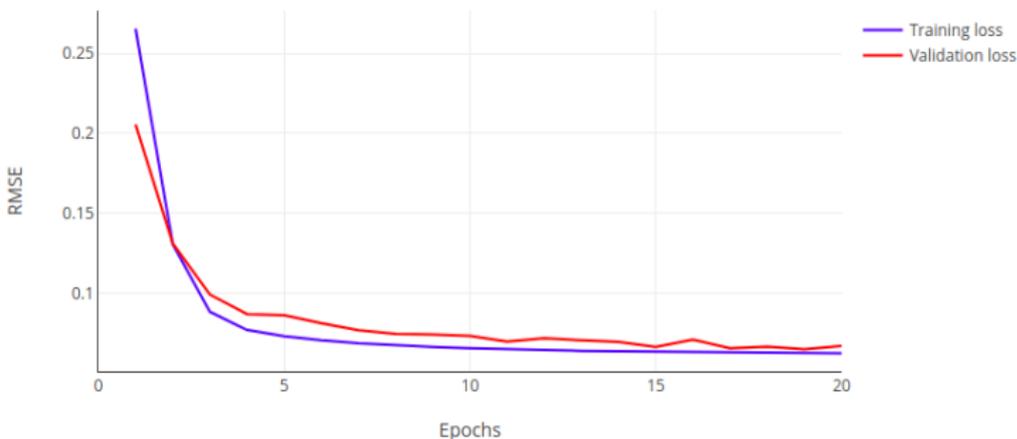


FIGURE – Perte de validation et d'entraînement pour le modèle univarié

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

Prédiction à partir du modèle univarié

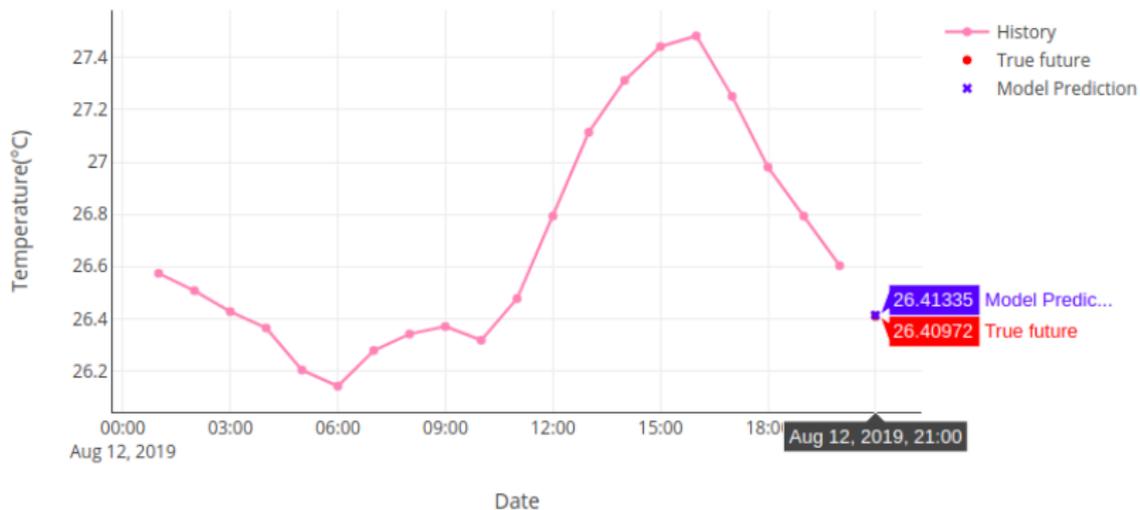


FIGURE – Prédiction de la température avec le modèle LSTM univarié

Modèle multivarié

- prédiction de la température en fonction d'autres champs physiques
- échantillonnage des 120 dernières observations
- modèle 'single-step' : prédiction à l'instant t
- modèle 'multi-step' : prédiction à l'instant t , $t + 1$ et $t + 2$

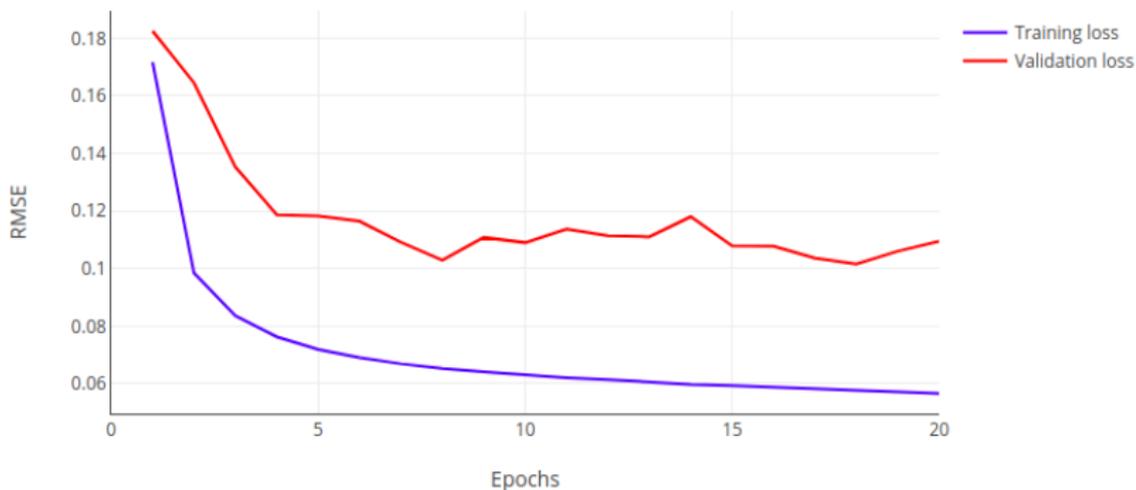


FIGURE – Perte de validation et d'entraînement pour le modèle multivarié single-step

```
single_step_model = tf.keras.models.Sequential()  
single_step_model.add(tf.keras.layers.LSTM(32,  
  
input_shape=x_train_single.shape[:-2]))  
single_step_model.add(tf.keras.layers.Dropout(.09))  
single_step_model.add(tf.keras.layers.Dense(1))  
  
single_step_model.compile(optimizer='SGD', loss=root_mean_squared_error)
```

```
single_step_model = tf.keras.models.Sequential()  
single_step_model.add(tf.keras.layers.LSTM(32,  
input_shape=x_train_single.shape[:-2]))  
single_step_model.add(tf.keras.layers.Dropout(.09))  
single_step_model.add(tf.keras.layers.Dense(1))  
  
single_step_model.compile(optimizer='SGD', loss=root_mean_squared_error)
```

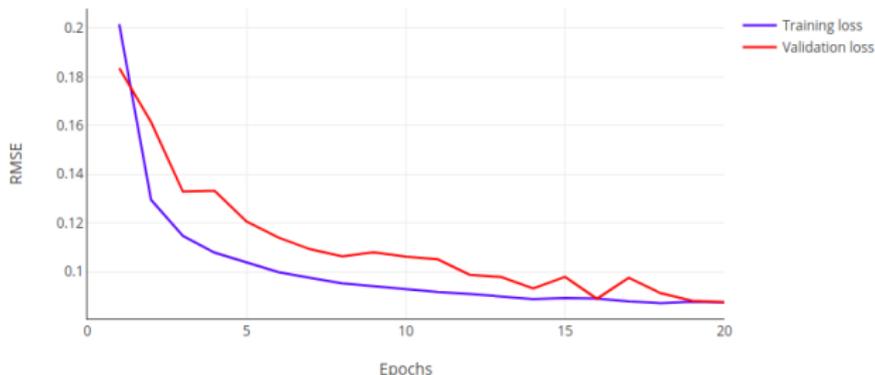


FIGURE – Perte de validation et d'entraînement pour le modèle multivarié single-step avec la méthode de DROPOUT

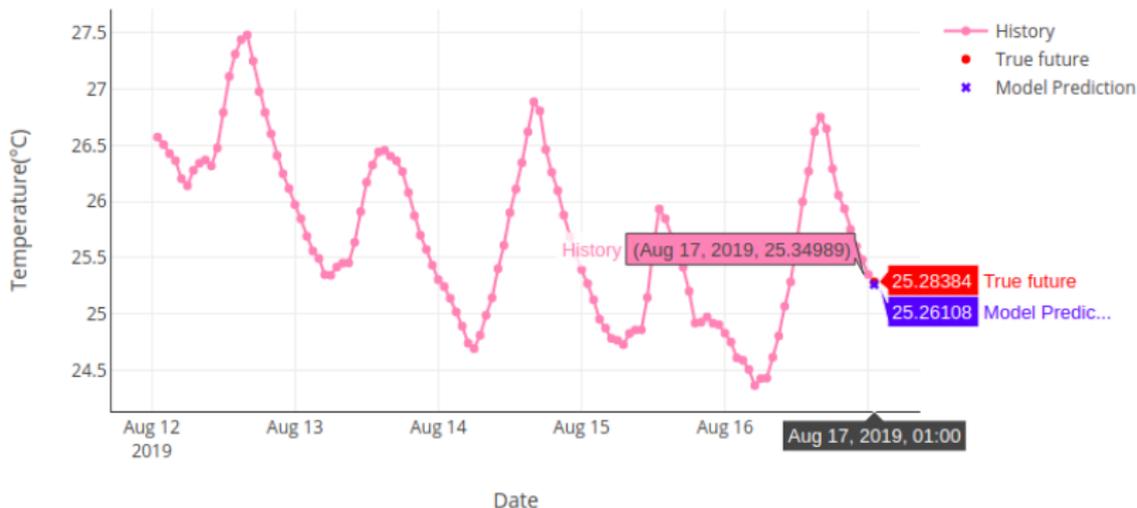


FIGURE – Prédiction de la température avec le modèle LSTM single-step

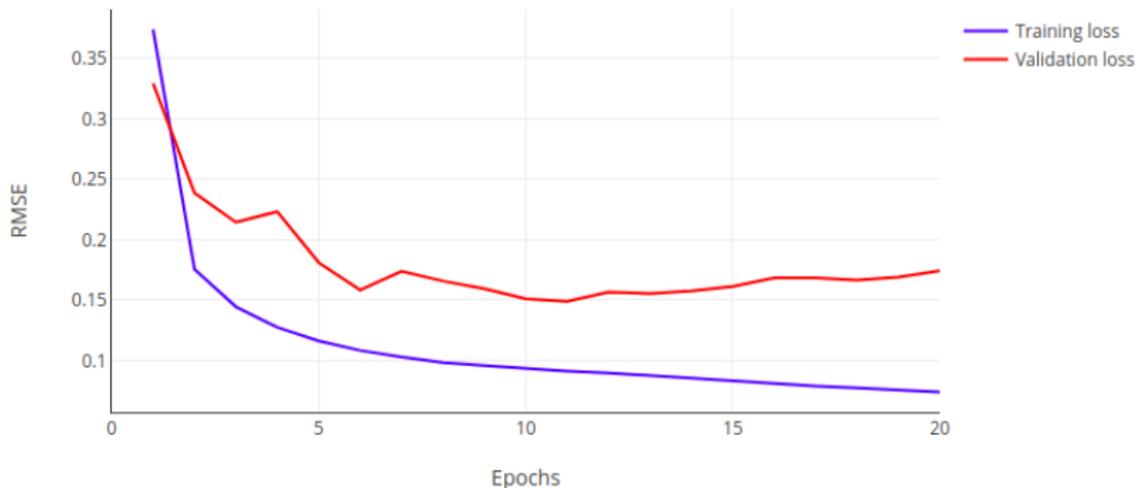


FIGURE – Perte de validation et d'entraînement pour le modèle multivarié multi-step

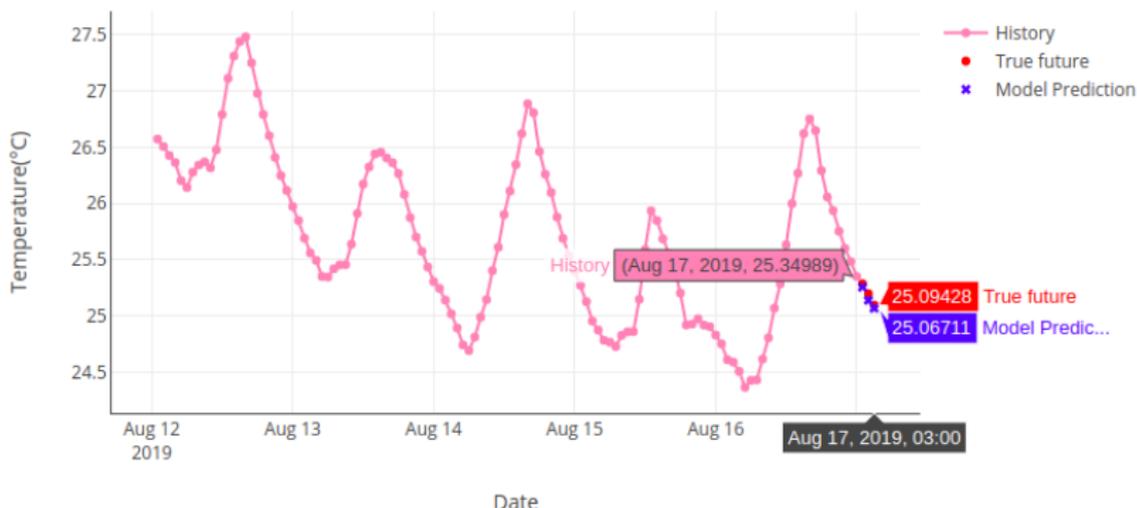


FIGURE – Prédiction de la température avec le modèle LSTM multi-step

PMV

- prédit la réponse moyenne d'un groupe de personnes selon l'échelle de sensation thermique de l'ASHRAE.
- déséquilibre entre les flux thermique réel et requis.
- dépend de plusieurs paramètres.

PMV

- prédit la réponse moyenne d'un groupe de personnes selon l'échelle de sensation thermique de l'ASHRAE.
- déséquilibre entre les flux thermique réel et requis.
- dépend de plusieurs paramètres.

PPD

- prédiction quantitative du pourcentage thermique insatisfaites.
- dépend du PMV.

| Run no. | T_a (°C) | T_{mr} (°C) | v_{ar} (m/s) | RH (%) | M (met) | I_{cl} (clo) | PMV | PPD | Calculated PMV | Calculated PPD | Verify |
|---------|------------|---------------|----------------|--------|---------|----------------|-------|-----|----------------|----------------|---------------|
| 1 | 22.0 | 22.0 | 0.1 | 60 | 1.2 | 0.5 | -0.75 | 17 | -0.75 | 16.85 | correct value |
| 2 | 27.0 | 27.0 | 0.1 | 60 | 1.2 | 0.5 | 0.77 | 17 | 0.76 | 17.17 | correct value |
| 3 | 27.0 | 27.0 | 0.3 | 60 | 1.2 | 0.5 | 0.44 | 9 | 0.43 | 8.86 | correct value |
| 4 | 23.5 | 25.5 | 0.1 | 60 | 1.2 | 0.5 | -0.01 | 5 | -0.02 | 5.01 | correct value |
| 5 | 23.5 | 25.5 | 0.3 | 60 | 1.2 | 0.5 | -0.55 | 11 | -0.56 | 11.57 | correct value |

FIGURE – Valeurs de PMV et PPD selon la norme(ISO-7730)[6]

Application

Gestion des
données
manquantes
en thermique
du bâtiment

MADANI
Sarra Amel
OUACHOUR
Hanane

Introduction

Rappels

Requêtes
aléatoires

Test de
l'algorithme
d'imputation
(Random
Forest)

Réseaux de
neurones
récurrents

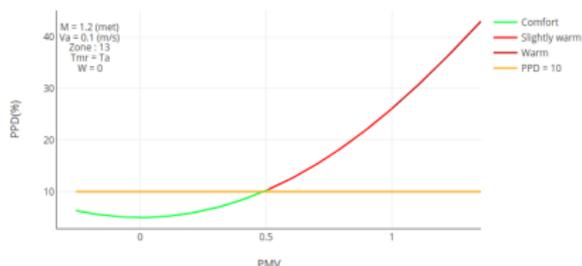
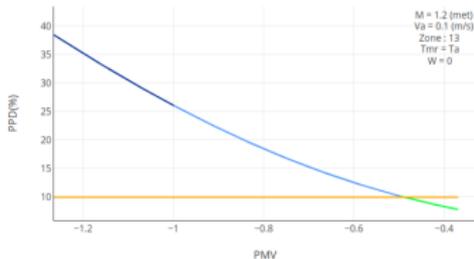
Modèle univarié
Modèle multivarié

Application
des données
au confort
thermique

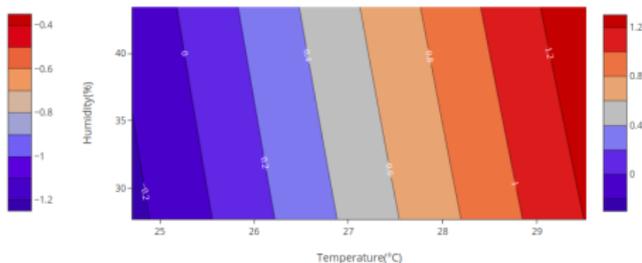
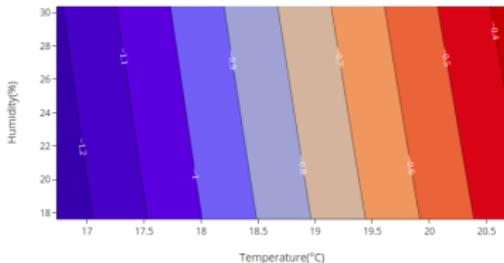
Conclusion

- Agrégation des données par zone.
- Zone étudiée : zone 13.
- Mois étudiés : Janvier et Juillet 2019.
- $T_{mr} = T_a$
- $v_{ar} = 0.01(m/s)$, $M = 1.2(met)$
- $I_{cl} = 0.5(clo)$ en été, et $1(clo)$ en hiver.

PPD en fonction du PMV pour les mois de Janvier et de Juillet



PMV en fonction de la température et de l'humidité pour les mois de Janvier et de Juillet



Travail effectué

Développement de scripts :

- requête à partir d'une BD.
- application de méthodes de Machine Learning.
- évaluation du modèle de confort thermique.

Travail effectué

Développement de scripts :

- requête à partir d'une BD.
- application de méthodes de Machine Learning.
- évaluation du modèle de confort thermique.

Perspectives

- prédiction de la température sur plusieurs temps
- utilisation des LSTM pour compléter l'ensemble des données manquantes
- étude de l'efficacité des méthodes utilisées

Travail effectué

Développement de scripts :

- requête à partir d'une BD.
- application de méthodes de Machine Learning.
- évaluation du modèle de confort thermique.

Perspectives

- prédiction de la température sur plusieurs temps
- utilisation des LSTM pour compléter l'ensemble des données manquantes
- étude de l'efficacité des méthodes utilisées

Retour d'expériences

- maîtrise d'outils de programmation
- acquisition de nouvelles technologies
- méthodologie de travail

Références



Transition énergétique des bâtiments, https://afortech.com/news/transition-energetique-des-batiments-leffet-papillon/?fbclid=IwAR1iDUyM6mtqp77xDvIQDMrDj_j9S9HAC0yTGHP7qhTwZpuaTnyX9aiqjM



Nearby infrastructures and available services, <http://www.telecom-physique.fr/en/school/campus/news/liot-au-coeur-de-la-maintenance-dans-lusine-du-futur-conference-us>



Recurrent Neural Networks cheatsheet, <https://stanford.edu/~shervine/teaching/cs-230/cheatsheet-recurrent-neural-networks>.



Aurélien Géron. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow. September 2019.



Calcul clustérisé pour la modélisation numérique thermique du patrimoine – optimisation énergétique, https://www.dropbox.com/s/km6e4i9hak5knzu/CCH_AMI.docx?dl=0



Ergonomie des ambiances thermiques. Détermination analytique et interprétation du confort thermique par le calcul des indices PMV et PPD et par des critères de confort thermique local. 2005.



Le rapport de stage est disponible sur le site IBat Docs <https://docs.cemosis.fr/ibat/latest/reports/madani-ouachour/index.html>

Merci pour votre attention

