

# DÉVELOPPEMENT ET APPLICATION DES RÉSEAUX DE NEURONES PAR GRAPHES POUR LA PRÉVISION SPATIO-TEMPORELLE DES NIVEAUX D'EAU SOUTERRAINE

Par

Xiao Xia Liang

Thèse présentée pour l'obtention du grade de  
*Philosophiae Doctor, Ph.D.*  
en sciences de la Terre

## Jury d'évaluation

Président du jury et examineur interne	J. Christian Dupuis Département de géologie et de génie géologique Faculté de science et de génie Université Laval
Examineur externe	Gilles Bellefleur Commission géologique du Canada Ressources naturelles Canada
Examineur externe	Jean-François Lalonde Département de génie électrique et de génie informatique Faculté de science et de génie Université Laval
Codirecteur de recherche	Daniel Paradis Centre Eau Terre Environnement Institut national de la recherche scientifique
Codirecteur de recherche	Maxime Claprood Département des sciences appliquées Université du Québec à Chicoutimi
Directeur de recherche	Erwan Gloaguen Centre Eau Terre Environnement Institut national de la recherche scientifique



## REMERCIEMENTS

Les quatre dernières années de mes études peuvent être résumées comme intéressantes et différentes. Il y a eu de nombreux hauts et bas pendant cette période, et puis il y a eu la COVID-19. Je ne suis pas certaine que je prendrais à nouveau cette décision si j'avais le recul nécessaire, mais une chose dont je suis certaine, c'est d'avoir eu Erwan comme directeur. Sans son soutien et ses encouragements, je ne serais pas ici aujourd'hui. Ce que je veux dire, merci Over-Lord! Nous avons nos accords et désaccords, mais même avec du recul, je ne changerais rien.

Je tiens à remercier mes co-directeurs et amis, Max et Daniel, merci pour votre soutien. Je souhaite également remercier mes collègues et amis. Merci d'avoir été là pour moi à travers tous les hauts et les bas. Merci d'avoir écouté mes plaintes. Et un immense merci au Dr Philippe Renard et à son groupe pour m'avoir accueillie en tant qu'étudiante d'échange et intégrée à leur équipe.

Quant à ma famille, je voudrais remercier Ying, Billy, Rhaine, Penelope, Ping-Ping et Junior d'avoir été là pour moi, de m'avoir nourrie et de m'avoir donné l'espoir et la force de continuer à avancer. Je remercie également ma grand-mère et mon grand-père de m'avoir gardée en vie lorsque j'étais enfant et de m'avoir inculqué l'idée que je peux accomplir tout ce que je décide d'entreprendre; merci d'avoir toujours cru en moi.

Enfin, mais non des moindres, je tiens à remercier Simon et Tristan. Sans votre amour, votre rire et votre soutien, je ne serais pas ici aujourd'hui. J'ai cette belle vie grâce à vous!

---

The last 4 years of my studies could be summarized as interesting and different. There were many ups and downs during this time, and then there was COVID-19. I am not certain I would make this decision again if I had the hindsight, but one thing that I am certain of, is having Erwan as my supervisor. Without his support and encouragement, I would not be here today. What I want to say is - thank you Overlord! We have our agreements and disagreements, but I would not change that even with hindsight.

I would like to thank my co-supervisors and friends, Max and Daniel, thank you for your support. I would also like to thank my colleagues and friends. Thank you for being there for me for all the ups and downs. Thank you for listening to my rants. And a big thank you to Dr. Philippe Renard, and his group for accepting me as an exchange student and including me into their group.

As for my family, I would like to thank, Ying, Billy, Rhaine, Penelope, Ping-Ping, and Junior for being there for me, feeding me and giving me hope and strength to keep going forward. I would like to thank my grandmother and grandfather for keeping me alive as a young child and instilling in me that I can do and achieve what I set my mind on; thank you for always believing in me.

Last, but not least, I would like to thank Simon and Tristan. Without your love, laughter, and support, I would not be here today. I have this beautiful life because of you!



# RÉSUMÉ

La prévision spatio-temporelle des niveaux des nappes phréatiques est une approche de modélisation prédictive qui utilise des données spatiales et temporelles pour estimer les niveaux futurs des nappes phréatiques. Cette méthodologie joue un rôle essentiel dans la gestion et la planification des ressources en eau en fournissant des indicateurs précoces de pénuries potentielles, permettant ainsi une atténuation proactive des sécheresses et favorisant une utilisation durable de l'eau. Dans cette dissertation, nous explorons des modèles d'apprentissage automatique afin d'améliorer la précision et la fiabilité de ces prévisions. Plus précisément, nous nous concentrons sur l'applicabilité et la capacité des réseaux de neurones par graphe (RNG) à modéliser les niveaux spatio-temporels des niveaux d'eau souterraine. Les RNG conviennent particulièrement bien à cette tâche, car ils peuvent capturer les relations complexes et non linéaires ainsi que les schémas de connectivité inhérents aux données spatiales, offrant ainsi un cadre robuste pour estimer les interactions entre les différents points de surveillance des nappes phréatiques. Pour évaluer de manière approfondie le potentiel de prévision de ces modèles RNG, nous commençons par tester l'approche en utilisant des données simulées dérivées de modèles physiques qui reproduisent le comportement des puits de pompage et de surveillance. Cet environnement contrôlé nous permet d'ajuster nos modèles par rapport à des paramètres connus et de valider leurs performances dans des conditions idéales. En nous appuyant sur cette base, nous étendons ensuite notre analyse en intégrant un ensemble de données de puits de surveillance mesurées sur le terrain, ce qui fournit un cas d'essai plus réaliste et exigeant. Enfin, pour démontrer la polyvalence de notre approche, nous appliquons également notre cadre de prévision aux débits des sources karstiques, illustrant ainsi comment les modèles RNG peuvent être utilisés efficacement dans une gamme de contextes hydrologiques et de types de données.

Trois articles sont présentés dans cette thèse. Le premier article présente un cadre de RNG pour la prévision spatio-temporelle des niveaux des nappes phréatiques dans les puits de surveillance et de pompage. Ce cadre est développé en tant que substitut aux modèles physiques hydrogéologiques afin de surmonter des défis tels que l'adaptabilité et l'efficacité computationnelle. Le deuxième article présente une étude sur l'importance de la résolution et de la taille des données pour obtenir une prévision précise des niveaux spatio-temporels d'eau souterraine. Le troisième article présente une nouvelle fonction de perte, l'extrême perte, visant à améliorer les capacités de prévisions spatio-temporelles de RNG pour prédire les événements extrêmes, tels que les inondations et les sécheresses. Ces articles visent à développer des outils en accès libre pour aider à la gestion des ressources en eau souterraine.

**Mots-clés** Apprentissage Automatique; Réseau de Neurones Profond; Réseau de Neurones par graphe; Prévision Spatio-temporelle; Prévision des Niveaux des Nappes Phréatiques; Prévision de Sécheresse; Développement de Fonctions de Perte; Prévision des Sources Karstiques



## ABSTRACT

Spatio-temporal groundwater (GW) level forecasting is a predictive modeling approach that utilizes spatial and temporal data to estimate future groundwater levels. This methodology plays a critical role in the management and planning of water resources by providing early indicators of potential shortages, thereby enabling proactive drought mitigation and promoting sustainable water use. In this dissertation, we explore data-driven machine learning models to enhance the precision and reliability of these forecasts. More specifically, we focus on the applicability and capability of graph neural networks (GNNs) for modeling spatio-temporal groundwater levels. GNNs are particularly well-suited to this task because they can capture the complex, non-linear relationships and connectivity patterns inherent in spatial data, thereby offering a robust framework to estimate the interactions between different groundwater monitoring points. To thoroughly assess the forecasting potential of these GNN models, we begin by testing the approach using simulated data derived from physical models that replicate the behavior of pumping and monitoring wells. This controlled environment allows us to fine-tune our models against known parameters and validate their performance under ideal conditions. Building on this foundation, we then extend our analysis to incorporate a dataset of field-measured monitoring well data, which provides a more realistic and challenging test case. Finally, to demonstrate the versatility of our approach, we further apply GNNs' spatio-temporal forecasting capabilities to predict dynamic karst spring discharges, illustrating how GNNs can be effectively utilized across a range of hydrological contexts and data types.

Three articles are presented in this dissertation. The first article presents a GNN framework for spatio-temporal GW level forecasting in monitoring and pumping wells. This framework is developed as a surrogate for hydrogeological physical models to overcome challenges such as adaptability and computational efficiency. The second article presents a study on the importance of data resolution and size for obtaining accurate spatio-temporal GW level forecasting. The third article presents a novel loss function, extreme loss, to enhance GNN's spatio-temporal forecasting capabilities to predict extreme events, such as floods and droughts. These articles are aimed at developing open-source tools to aid in GW resource management.

**Keywords** Machine Learning; Deep Neural Network; Graph Neural Network; Spatio-Temporal Forecasting; Groundwater Level Forecasting, Drought Forecast, Loss Function Development, Karst Spring Forecast



# TABLE DES MATIÈRES

<b>REMERCIEMENTS</b> .....	<b>iii</b>
<b>RÉSUMÉ</b> .....	<b>v</b>
<b>ABSTRACT</b> .....	<b>vii</b>
<b>TABLE DES MATIÈRES</b> .....	<b>ix</b>
<b>LISTE DES FIGURES</b> .....	<b>xi</b>
<b>LISTE DES TABLEAUX</b> .....	<b>xvii</b>
<b>LISTE DES ABRÉVIATIONS</b> .....	<b>xix</b>
<b>1 INTRODUCTION</b> .....	<b>1</b>
1.1 CONTEXTE.....	1
1.2 OBJECTIFS .....	3
1.2.1 <i>Sous-objectif 1</i> .....	3
1.2.2 <i>Sous-objectif 2</i> .....	4
1.2.3 <i>Sous-objectif 3</i> .....	4
1.3 L'APPRENTISSAGE AUTOMATIQUE.....	4
1.3.1 <i>Réseau de Neurones</i> .....	5
1.3.2 <i>Réseau de Neurones par Graphe</i> .....	8
1.3.3 <i>Utilisation des RNG dans cette thèse</i> .....	12
1.4 RÉSUMÉ DES ARTICLES .....	12
1.4.1 <i>Contributions de l'article 1</i> .....	13
1.4.2 <i>Contributions de l'article 2</i> .....	16
1.4.3 <i>Contributions de l'article 3</i> .....	18
1.5 PRINCIPALES CONTRIBUTIONS .....	20
<b>2 ARTICLE I : GRAPH NEURAL NETWORK FRAMEWORK FOR SPATIO-TEMPORAL GROUNDWATER LEVEL FORECASTING</b> .....	<b>21</b>
2.1 INTRODUCTION .....	23
2.2 GRAPH NEURAL NETWORK .....	25
2.2.1 <i>Graph Notations</i> .....	25
2.2.2 <i>Graph Convolution</i> .....	26
2.2.3 <i>GCN-LSTM Theory</i> .....	26
2.3 THE NUMERICAL MODEL .....	27

2.4	METHODOLOGY .....	30
2.4.1	<i>Framework Construction</i> .....	30
2.4.2	<i>GCN-LSTM Model Training and Optimization</i> .....	35
2.5	RESULTS .....	36
2.6	DISCUSSION AND CONCLUSION .....	43
<b>3</b>	<b>ARTICLE II : COMPARISON OF FORECASTING CAPABILITIES OF DEEP LEARNING MODELS WITH VARYING TIME SERIES RESOLUTION FOR ACCURATE KARST SPRING DISCHARGE PREDICTIONS .....</b>	<b>47</b>
3.1	INTRODUCTION .....	49
3.2	STUDY AREA .....	51
3.2.1	<i>Time Series Data</i> .....	52
3.3	DNN MODELS .....	53
3.3.1	<i>GWN model</i> .....	53
3.3.2	<i>GRU and LSTM Auto-Encoder Models</i> .....	55
3.4	EXPERIMENT SETUP.....	58
3.5	RESULTS.....	62
3.6	CONCLUSIONS AND DISCUSSIONS.....	69
3.7	ACKNOWLEDGMENTS.....	70
3.7.1	<i>Authors' Note</i> .....	70
3.7.2	<i>Code and Data Availability</i> .....	70
<b>4</b>	<b>ARTICLE III : DEEP NEURAL NETWORK LOSS FUNCTION DEVELOPMENT FOR FORECASTING EXTREME EVENTS IN TIME SERIES DATA.....</b>	<b>73</b>
4.1	INTRODUCTION .....	75
4.2	STUDY AREAS.....	77
4.2.1	<i>Milandre Karst System</i> .....	77
4.2.2	<i>Yamaska Watershed</i> .....	77
4.3	TIME SERIES DATA.....	78
4.3.1	<i>Milandre karst system data</i> .....	78
4.3.2	<i>Yamaska Monitoring Well Data</i> .....	79
4.3.3	<i>MPS Time Series Gap Filling</i> .....	81
4.4	EXTREME LOSS FUNCTION .....	81
4.5	DEEP LEARNING MODELS .....	83
4.5.1	<i>GWN Model</i> .....	83
4.5.2	<i>GRU Auto-Encoder Model</i> .....	86

4.5.3	<i>Experimental Setup</i> .....	86
4.6	RESULTS .....	87
4.6.1	<i>GWN Model Results</i> .....	87
4.6.2	<i>GRU Model Results</i> .....	88
4.6.3	<i>High Resolution Dataset</i> .....	89
4.7	CONCLUSIONS AND DISCUSSIONS .....	91
<b>5</b>	<b>DISCUSSION</b> .....	<b>105</b>
<b>6</b>	<b>CONCLUSION</b> .....	<b>109</b>
	<b>BIBLIOGRAPHIE</b> .....	<b>111</b>



## LISTE DES FIGURES

FIGURE 1.1 LA STRUCTURE IMBRIQUÉE DE LA CLASSIFICATION DES RÉSEAUX DE NEURONES PAR GRAPHE. ....	5
FIGURE 1.2 LA STRUCTURE DU MODÈLE GCN. LA FIGURE EST MODIFIÉE À PARTIR KIPF ET AL. (2016) .....	9
FIGURE 1.3 PROJECTION DU GRAPHE DANS L'ESPACE DES DONNÉES. CHAQUE NŒUD REPRÉSENTE UN PUIT, LES ARÊTES REPRÉSENTENT LA CONNECTIVITÉ HYDRAULIQUE ENTRE LES PUIT. LA COULEUR SPÉCIFIQUE DES NŒUDS DÉFINIT UN VOISINAGE DE NŒUDS QUI SONT LES PLUS PROCHES DANS L'ESPACE DES DONNÉES. COMME CE GRAPHE EST UNE PROJECTION DANS L'ESPACE DES DONNÉES, LES POSITIONS DES NŒUDS DANS CE GRAPHE N'ONT AUCUNE RELATION AVEC L'EMPLACEMENT RÉEL DES PUIT DANS LE MONDE PHYSIQUE. L'ÉPAISSEUR DES ARÊTES REPRÉSENTE LA CONNECTIVITÉ DES PUIT.....	10
FIGURE 1.4 UN EXEMPLE DE GRAPHE DE SYSTÈME KARSTIQUE. IL S'AGIT D'UN GRAPHE PONDÉRÉ ET ORIENTÉ AVEC DES NŒUDS FIXÉS À LEUR EMBLACEMENT PHYSIQUE. LES FLÈCHES DES ARÊTES VERTES INDIQUENT LA DIRECTION GÉNÉRALE DE L'ÉCOULEMENT DES EAUX SOUTERRAINES ENTRE CHAQUE STATION.....	13
FIGURE 2.1 THE FRAMEWORK CONSISTS OF 5 MAIN STEPS. STEP 1 : DATA RESAMPLING, NORMALIZATION, AND COMPILATION. STEP 2 : GRAPH CONSTRUCTION USING THE SPATIAL FEATURES. STEP 3 : PCA TRANSFORM AND COMPONENTS SPLITTING USING THE TEMPORAL FEATURES OF THE DATA. STEP 4 : GNN MODEL TRAINING BY TAKING THE ADJACENCY MATRIX OF THE GRAPH FROM STEP 2 AND THE SPLIT PCA COMPONENT OF THE TRANSIENT GW LEVEL FROM STEP 3. STEP 5 : GW LEVEL PREDICTIONS BY INVERSE TRANSFORM OF GNN PCA COMPONENTS. ....	28
FIGURE 2.2 LEFT : LOCATION OF THE NUMERICAL MODEL ON THE SOUTH SHORE OF THE ST. LAWRENCE RIVER IN QUEBEC (CANADA). THICK RED LINE IS THE OUTLINE OF THE NUMERICAL MODEL. RIGHT : 3D NUMERICAL MODEL (VERTICAL EXAGGERATION OF 15 TO 1)..	29
FIGURE 2.3 3D GEOMODEL OF THE HYDROSTRATIGRAPHIC UNITS AT VILLE MERCIER, QUEBEC, CANADA. REGIONAL VIEW HAS A VERTICAL EXAGGERATION OF 40, AND THE LOCAL VIEW HAS AN EXAGGERATION OF 20. THIS FIGURE IS MODIFIED FROM (CLAPROOD ET AL., 2023).....	29
FIGURE 2.4 PUMPING SCENARIOS APPLIED TO THE NUMERICAL HYDROGEOLOGICAL MODEL TO CARRYOUT SIMULATIONS.....	30
FIGURE 2.5 WELL GW LEVELS FROM TRANSIENT NUMERICAL MODEL SIMULATIONS. THE BLACK CIRCLES IN BOTH PLOTS HIGHLIGHT SOME OF THE INSTABILITIES THAT EMERGED DURING NUMERICAL MODEL SIMULATIONS.....	31

FIGURE 2.6 GRAPH HOMOPHILY PROJECTION OF WELL NODES NUMBERED FROM 0 TO 43. THE GRAPHS ARE CONSTRUCTED WITH A RADIUS CUT-OFF OF 0.15, 0.20, 0.25, AND 0.30 FOR PLOTS 1, 2, 3, AND 4, RESPECTIVELY. THE RADIUS CUT-OFF IS SHOWN TO INFLUENCE THE DEGREE OF COMPLEXITY OF THE RELATIONSHIPS BETWEEN THE WELLS (BASTIAN ET AL., 2009A).....	34
FIGURE 2.7 TRAINING AND VALIDATION LOSS OF DATASET 1 AND 2. THE 10-MODEL TRAINING AND VALIDATION LOSS ARE PLOTTED FOR EACH DATASET.....	37
FIGURE 2.8 DATASET 1 OF TRAINING, VALIDATION, AND TESTING OF THE GNN MODEL. MONITORING WELLS 2 AND 11, AND PUMPING WELLS 30 AND 37 ARE SHOWN FOR DATASET 1. PLOTS 1, 3, 5, AND 7 ARE THE PCA GW SIGNAL. PLOTS 2, 4, 6, AND 8 ARE THE PCA INVERSE TRANSFORMED GW LEVELS MEASURED IN METERS ABOVE SEA LEVEL. THE BLACK DASH LINE IN ALL PLOTS ARE THE SIMULATED GW LEVELS AND RED LINES ARE THE PREDICTED VALUES FROM GCN-LSTM NETWORK. ....	41
FIGURE 2.9 DATASET 2 OF TRAINING, VALIDATION, AND TESTING OF THE GNN MODEL. MONITORING WELLS 2 AND 11, AND PUMPING WELLS 30 AND 37 ARE SHOWN FOR DATASET 1. PLOTS 1, 3, 5, AND 7 ARE THE PCA GW SIGNAL. PLOTS 2, 4, 6, AND 8 ARE THE PCA INVERSE TRANSFORMED GW LEVELS MEASURED IN METERS ABOVE SEA LEVEL. THE BLACK DASH LINE IN ALL PLOTS ARE THE SIMULATED GW LEVELS AND RED LINES ARE THE PREDICTED VALUES FROM GCN-LSTM NETWORK. ....	42
FIGURE 3.1 BETTERAZ SPRING STUDY AREA. DATA COLLECTED FROM THE 4 STATIONS, BETTERAZ SPRING, CA2, SRM2 AND CP1 ARE USED TO TRAIN THE DNN MODELS .....	52
FIGURE 3.2 THE TOP PLOT CONTAINS ALL TIME SERIES DATA FROM THE 4 MONITORING STATIONS. THE BOTTOM PLOT IS DATA FROM THE BETTERAZ SPRING STATION.....	53
FIGURE 3.3 THE GRAPH WAVE NET FRAMEWORK. THIS FIGURE IS TAKEN FROM WU ET AL. (2019). ....	55
FIGURE 3.4 ILLUSTRATIONS OF THE DIRECTED GRAPH USED FOR GWN MODEL TRAINING. ON THE LEFT, THE PHYSICAL LOCATION OF THE STATIONS ARE SHOWN AS NODES WITH DIRECTED EDGES THAT FOLLOW THE GENERAL FLOW DIRECTION OF GROUNDWATER. ON THE RIGHT, A GRAPH PROJECTION OF NODES AND EDGES ARE SHOWN. THIS DIAGRAM IS A PROJECTION, THEREFORE, THE NODE STATIONS ARE NOT BASED ON PHYSICAL LOCATIONS, BUT RATHER A RANDOM PROJECTION IN THAT DATA SPACE. ....	56
FIGURE 3.5 GRU UNIT USED IN THE GRU AUTO-ENCODER (CHO ET AL., 2014). THIS UNIT CONTAINS 2 GATES : THE RESET AND THE UPDATE. IT ALSO CONTAINS A HIDDEN STATE THAT LOOPS BACK INTO THE NEXT RECURSIVE STEP. ....	57
FIGURE 3.6 THE LSTM UNIT CONTAINS 3 GATES : THE FORGET, THE INPUT AND THE OUTPUT. THERE ARE 2 STATES, THE HIDDEN AND THE CELL, THAT LOOPS BACK INTO THE UNIT FOR THE NEXT RECURSIVE STEP. ....	58
FIGURE 3.7 AN ILLUSTRATION OF AN AUTO-ENCODER MODEL. THE INPUT DATA $X$ IS INPUTTED INTO THE ENCODER MAPPING FUNCTION AND A FIXED LENGTH ENCODED VECTOR, $c$ , IS OBTAINED AND FORWARDED TO THE DECODER. THE DECODER THEN DECODES THE ENCODED INFORMATION IN THE MAPPING FUNCTION AND MAKES A PREDICTION OUTPUT, $Y$ . ..	59

FIGURE 3.8 DATA SPLIT FOR DNN MODEL TRAINING, VALIDATION, AND TESTING ARE 70%, 15%, AND 15% RESPECTIVELY. ....	59
FIGURE 3.9 THE SEQUENCE-TO-SEQUENCE DATA PREPARATION FOR DNN MODEL TRAINING 62	
FIGURE 3.10 HOURLY MEASURED GROUNDWATER LEVEL DATA. THE FORECASTING OF 3, 6 AND 12 TIME STEPS ARE PLOTTED. ....	64
FIGURE 3.11 THE RESOLUTION OF THE DATA IS AT 4-HOUR PER TIME STEP. THE FORECASTING OF 3, 6 AND 12 TIME STEPS ARE PLOTTED.....	65
FIGURE 3.12 THE RESOLUTION OF THE DATA IS AT 8-HOUR PER TIME STEP. THE FORECASTING LENGTHS OF 3, 6 AND 12 ARE PLOTTED. ....	66
FIGURE 3.13 THE RESOLUTION OF THE DATA IS AT 12-HOUR PER TIME STEP. THE FORECASTING LENGTHS OF 3, 6 AND 12 ARE PLOTTED. ....	67
FIGURE 3.14 THE RESOLUTION OF THE DATA IS AT 24-HOUR PER TIME STEP. THE FORECASTING LENGTHS OF 3, 6 AND 12 ARE PLOTTED. ....	68
FIGURE 3.15 THE INTERPOLATION OF THE $R^2$ , RMSE, MAE AND MAPE METRIC RESULTS OF ALL MODELS. THE SCATTER POINTS ARE THE SAMPLED LOCATIONS AND THE SIZE OF THE POINTS ARE THE DATA SIZE AVAILABLE FOR MODEL TRAINING. ....	72
FIGURE 4.1 MILANDRE CATCHMENT WITH LOCATIONS OF THE MONITORING AND SPRING MEASURING STATIONS. DATA USED IN THIS STUDY IS FROM STATIONS (MILANDRINE, BÂME, SAIVU, AND FONT) ENCIRCLED IN RED. THIS MAP IS MODIFIED FROM VUILLEUMIER ET AL. (2021).....	78
FIGURE 4.2 THE YAMASKA WATERSHED IS LOCATED ON THE SOUTH SHORE OF THE SAINT-LAWRENCE RIVER. THE LOCATIONS AND LABELS OF THE MONITORING WELLS ARE SHOWN. 79	
FIGURE 4.3 MEASURED DISCHARGE FROM STATIONS MILANDRINE, SAIVU, FONT AND BÂME. 80	
FIGURE 4.4 MEASURED DECLINE IN THE YAMASKA GROUNDWATER TABLE. THE 3 DECLINE PERIODS ARE BOUNDED BY RED DOTTED LINES. ....	80
FIGURE 4.5 MILANDRE MEASURED AND SIMULATED MISSING DATA. THE SIMULATED MISSING TIME SERIES DATA IS GENERATED FROM MPS SIMULATIONS. ....	82
FIGURE 4.6 WEIGHTED AND DIRECTED GRAPH OF THE MILANDRE MONITORING STATIONS. THE MAP ON THE LEFT SHOWS THE PHYSICAL LOCATIONS OF THE STATIONS WITH THEIR SPATIAL CONNECTIONS CALCULATED USING EQUATION 4.1. THE RIGHT ILLUSTRATION SHOWS THE WEIGHTED AND DIRECTED GRAPH OF THE STATIONS PROJECTED IN DATA SPACE. THE EDGE THICKNESS REPRESENTS THE WEIGHTED SPATIAL CONNECTIONS OF THE STATIONS (BASTIAN ET AL., 2009B).....	84

FIGURE 4.7 THE LEFT MAP SHOWS THE YAMASKA MONITORING WELL GRAPH FIXED IN PHYSICAL LOCATION. THE RIGHT GRAPH SHOWS THE YAMASKA MONITORING WELLS PROJECTED IN DATA SPACE. THE GRAPH IS CALCULATED USING EQUATION 4.2. THE COLOR OF NODE INDICATES THE NODES TO BE IN THE SAME NEIGHBORHOOD AS THEY ARE THE CLOSEST NEIGHBORS CALCULATED USING THE 6 PARAMETERS. THE EDGE THICKNESS REPRESENTS THE WEIGHTS OF SPATIAL CONNECTIONS BETWEEN THE WELLS. THE SIZE OF THE NODES REPRESENTS THE DEGREE OF CONNECTIONS (BASTIAN ET AL., 2009B).	85
FIGURE 4.8 GWN MODEL FORECAST RESULTS (MILANDRE) - 6 TIME STEP FORECAST LENGTHS.	89
FIGURE 4.9 CROSS PLOT OF MEASURED AND GWN PREDICTIONS (MILANDRE) OF 6 TIME STEP FORECAST LENGTHS.	90
FIGURE 4.10 GWN MODEL FORECAST RESULTS (MILANDRE) - 9 TIME STEP FORECAST LENGTHS.	91
FIGURE 4.11 CROSS PLOT OF MEASURED AND GWN PREDICTIONS (MILANDRE) OF 9 TIME STEP FORECAST LENGTHS.	92
FIGURE 4.12 GWN MODEL FORECAST RESULTS (YAMASKA) - 3 TIME STEP FORECAST LENGTHS.	93
FIGURE 4.13 CROSS PLOT OF MEASURED AND GWN PREDICTIONS (YAMASKA) OF 3 TIME STEP FORECAST LENGTHS.	94
FIGURE 4.14 GWN MODEL FORECAST RESULTS (YAMASKA) - 4 TIME STEP FORECAST LENGTHS.	95
FIGURE 4.15 CROSS PLOT OF MEASURED AND GWN PREDICTIONS (YAMASKA) OF 4 TIME STEP FORECAST LENGTHS.	96
FIGURE 4.16 GRU MODEL FORECAST RESULTS (MILANDRE) - 6 TIME STEP FORECAST LENGTHS.	97
FIGURE 4.17 CROSS PLOT OF MEASURED AND GRU PREDICTIONS (MILANDRE) OF 6 TIME STEP FORECAST LENGTHS.	98
FIGURE 4.18 GRU MODEL FORECAST RESULTS (MILANDRE) - 9 TIME STEP FORECAST LENGTHS.	99
FIGURE 4.19 CROSS PLOT OF MEASURED AND GRU PREDICTIONS (MILANDRE) OF 9 TIME STEP FORECAST LENGTHS.	100
FIGURE 4.20 GRU MODEL FORECAST RESULTS (YAMASKA) - 3 TIME STEP FORECAST LENGTHS.	101
FIGURE 4.21 CROSS PLOT OF MEASURED AND GRU PREDICTIONS (YAMASKA) OF 3 TIME STEP FORECAST LENGTHS.	102
FIGURE 4.22 GRU MODEL FORECAST RESULTS (YAMASKA) - 4 TIME STEP FORECAST LENGTHS.	103
FIGURE 4.23 CROSS PLOT OF MEASURED AND GRU PREDICTIONS (YAMASKA) OF 4 TIME STEP FORECAST LENGTHS.	104

## LISTE DES TABLEAUX

TABLEAU 2.1	FEATURES USED FOR GRAPH CONSTRUCTION AND MODEL TRAINING.....	32
TABLEAU 2.2	OPTIMIZED HYPERPARAMETERS FOR GCN-LSTM MODEL.....	36
TABLEAU 2.3	TESTING RADIUS CUT-OFF VALUES FOR GRAPH CONSTRUCTION ON DATASET 1.....	38
TABLEAU 2.4	TESTING RADIUS CUT-OFF VALUES FOR GRAPH CONSTRUCTION ON DATASET 2.....	38
TABLEAU 2.5	MEAN ERROR METRICS AND LOSS VALUES OF 10 GNN OPTIMIZATIONS FOR THE 2 DATASETS IN METERS.....	38
TABLEAU 3.1	THE WEIGHTED AND DIRECTED ADJACENCY MATRIX USED FOR GWN MODEL TRAINING. ....	54
TABLEAU 3.2	DATA COMPILATION FOR DNN MODEL TRAINING.....	61
TABLEAU 3.3	METRICS CALCULATED FROM THE TESTING DATASET FROM EACH OF THE DNN MODELS.....	71
TABLEAU 4.1	DATA COMPILATION FOR DL MODEL TRAINING.....	87
TABLEAU 4.2	GWN TEST DATASET METRICS.....	88
TABLEAU 4.3	GRU TEST DATASET METRICS.....	88
TABLEAU 4.4	TEST DATASET METRICS FROM GWN MODEL WITH HOURLY MEASURED DATA (MILANDRE).....	98



## LISTE DES ABRÉVIATIONS

$R^2$	Coefficient de détermination
AA	Apprentissage automatique
ACP	Analyse en composantes principales
Adam	Adaptive moment estimation
AP	Apprentissage profond
CNN	Convolutional neural network
DL	Deep learning
DNN	Deep neural network
ELF	Extreme Loss Function
EMA	Erreur moyenne absolue
EMAP	Erreur moyenne absolue en pourcentage
EMQ	Erreur moyenne quadratique
GCN	Graph convolutional network
GNN	Graph neural network
GRU	Gated recurrent unit
GW	Goundwater
GWN	Graph WaveNet
IA	Intelligence artificielle
LSTM	Long short-term memory
MAE	Mean absolute error
MAPE	Mean absolute percentage error
MASL	Meters above sea level
ML	Machine learning
MPS	Multiple point statistics
MSE	Mean square error
NARX	Nonlinear autoregressive network
NSE	Nash-Sutcliffe Efficiency
PCA	Principal component analysis
PINN	Physics-informed neural networks

RMSE	Root mean squared error
RNC	Réseau de neurones convolutif
RNG	Réseau de neurones par graphe
RNN	Recurrent neural network
RNP	Réseau de neurones profond
RNPA	Réseau de neurones à propagation avant
RNR	Réseau de neurones récurrent
T-GCN	Temporal graph convolutional network
TCN	Temporal convolutional network

# 1 INTRODUCTION

---

L'eau souterraine est une ressource vitale dont l'accès a un impact majeur sur la population mondiale, car elle affecte directement la santé humaine, l'environnement et la survie des espèces, tout en influençant également le développement économique des communautés. L'eau souterraine constitue la principale source d'eau utilisée pour les besoins domestiques, industriels et agricoles (Sadoff et al., 2007). Plus de la moitié de la population humaine dépend des eaux souterraines pour sa consommation, ce chiffre atteignant plus de 90% dans les zones rurales (Pltonykova et al., 2020). L'Organisation mondiale de la santé estime que d'ici 2025, la moitié de la population mondiale vivra dans des régions soumises à un stress hydrique ou en pénurie d'eau.

Selon Bates et al. (2008), les ressources en eaux souterraines ne sont pas seulement menacées par les changements climatiques, mais aussi par les activités anthropiques. Quel que soit le type d'activité humaine, les besoins en eau souterraine ne cessent de croître. Il est donc plus que jamais crucial de mettre en place des politiques de gestion efficaces, fondées sur des connaissances scientifiques et des faits, afin de réguler cette ressource précieuse. Comme l'indiquent Sadoff et al. (2007), il est possible d'adapter la gestion des ressources en eau grâce à un aménagement et une administration du territoire appropriés. Une des hypothèses de mes travaux de thèse est qu'en utilisant des outils scientifiques adéquats, l'humanité peut s'adapter aux insécurités hydriques provoquées par le changement climatique et l'augmentation des demandes anthropiques.

## 1.1 Contexte

Traditionnellement, les hydrogéologues et les gestionnaires utilisent des modèles numériques reproduisant la physique des bassins versants pour effectuer des prévisions hydrogéologiques, telles que l'estimation du stockage des aquifères, l'écoulement de l'eau souterraine, ainsi que le transport de masse pour la gestion et la réhabilitation des sites contaminés. Bien qu'un modèle numérique soit un outil efficace et largement utilisé en hydrogéologie, il présente également des limitations. Les modèles d'écoulement des eaux souterraines ne sont pas une représentation exacte du sous-sol, mais plutôt une représentation simplifiée qui souffre de non-unicité et d'incertitude (Claprood et al., 2023). Cela ne signifie pas que les modèles simplifiés ou généralisés ne sont pas utiles dans certaines situations, comme l'a bien exprimé George P.E. Box : « Tous les modèles sont faux, mais certains sont utiles ».

L'incertitude peut découler de nombreux facteurs liés à la représentation des processus modélisés. L'un de ces facteurs est le choix de l'algorithme. En sélectionnant un algorithme spécifique pour résoudre les processus, le modélisateur limite la représentation des processus naturels à

---

ceux inclus dans l'algorithme choisi. D'autres processus pourraient exister dans le système sans être pris en compte dans le code sélectionné. Une autre source d'incertitude provient des « inconnues des inconnues », un concept introduit par Hunt et al. (2010). En modélisation hydrogéologique, un modélisateur ne peut pas représenter des processus dont il ne connaît pas l'existence.

D'un point de vue plus pratique, les logiciels commerciaux peuvent être coûteux. De plus, les modèles numériques sont longs à construire et nécessitent un modélisateur expérimenté et/ou une personne possédant une connaissance approfondie du système modélisé. Une fois les modèles numériques construits, s'il faut ajouter de nouvelles données ou tester de nouveaux scénarios, cela requiert encore une personne hautement qualifiée pour exécuter ces modèles. Cette ressource humaine fait généralement défaut aux équipes de gestion des municipalités ou des associations à but non lucratif en charge des bassins versants. Un des exemples frappants est le modèle des lagunes de Mercier (Claprood et al., 2023). Les premières caractérisations ont eu lieu dans les années 1980 et de nombreux modèles numériques de l'écoulement ont été créés avec des ensembles de données différents dans différents logiciels. À chaque campagne de terrain, les modèles numériques devaient être mis à jour dans de nouveaux logiciels pour tenir compte des nouvelles données, ce qui engendre des coûts prohibitifs et des temps de modélisation très importants.

Les modèles axés sur les données (data-driven) peuvent aider à atténuer certains des problèmes rencontrés par les modèles numériques tout en accélérant le temps de calcul des prévisions. Si les données sont de bonne qualité et contiennent assez d'information, les modèles axés sur les données permettent de réduire une partie de l'incertitude rencontrée par les modèles numériques basés sur des processus grâce à leur capacité à représenter des relations non linéaires et complexes sans nécessiter de relations explicitement définies (Solomatine et al., 2008; Montáns et al., 2019). En particulier, les algorithmes de type réseau de neurones par graphe (RNG) sont des modèles axés sur les données qui ne prennent pas en compte explicitement les processus physiques, les formulations ou les relations entre les paramètres mais ont la capacité d'apprendre ces processus à partir de l'information contenue dans les données non spatialement structurées (Scarselli et al., 2008).

En effet, lorsqu'on utilise un RNG entraîné comme substitut aux modèles physiques, il n'est plus nécessaire d'exécuter des simulations futures sur le modèle physique. Cette approche permet d'économiser à la fois la puissance de calcul et les coûts de licence. Le modèle de substitution peut être entraîné à l'aide de données simulées provenant du modèle physique, ce qui lui permet d'apprendre les schémas de prévision du système physique et, par conséquent, de fonctionner efficacement comme substitut ou sur des séries temporelles de données réelles (An et al., 2018; Previati et al., 2024; Yongkai et al., 2022).

À l'instar des réseaux de neurones conventionnels, les RNG sont capables de représenter des relations non linéaires et complexes au sein des ensembles de données — des relations qui peuvent dépasser la compréhension humaine qui permettent de reproduire le comportement de

---

systèmes dynamiques complexes (<https://github.com/google-deepmind/graphcast>). En minimisant la subjectivité et les biais humains, ces modèles peuvent révéler des relations complexes jusque là inobservées dans les données.

Il est important de souligner que l'exécution d'un modèle RNG entièrement entraîné ne requiert pas une expertise en apprentissage automatique (AA) ou en modélisation hydrogéologique. Une personne disposant de connaissances de base en Python et en hydrogéologie peut exploiter le modèle (il est néanmoins conseillé de faire appel à un expert en hydrogéologie pour vérifier les résultats). Cette accessibilité accrue peut améliorer les prévisions pour la gestion des ressources en eau, notamment pour les municipalités ou les associations de bassin à but non lucratif.

## **1.2 Objectifs**

L'objectif technique global de cette thèse est de proposer des algorithmes de RNG afin de mieux prédire les niveaux d'eau souterraine en fonction des pressions anthropiques et des changements de recharge afin d'en faire une gestion durable de la ressource. Dans cette thèse, trois sous-objectifs principaux ont été atteints, chaque sous-objectif correspondant à un article présenté dans ce travail.

### **1.2.1 Sous-objectif 1**

Le premier objectif est de développer un cadre d'AA basé sur les modèles RNG qui peut être couplé avec des modèles hydrogéologiques numériques en tant que « substitut » ou « intermédiaire » pour prévoir les niveaux des eaux souterraines subissant différents scénarios de pompage. Ce cadre est conçu pour accélérer le processus de prévision spatio-temporelle des niveaux des eaux souterraines sous différentes contraintes aquifères. Il peut être utilisé comme substitut en exploitant les données de simulation de modèles physiques pour l'entraînement. Une fois entraîné, le modèle RNG peut ensuite réaliser des prévisions futures avec de nouvelles variables d'entrée, telles que la recharge et le pompage. Cette approche de substitution permettrait de réduire la charge computationnelle et l'expertise technique nécessaire pour faire l'évaluation des ressources en eau. Enfin, le modèle RNG utilisé dans ce cadre peut également fonctionner comme un modèle autonome entraîné pour prévoir les niveaux des eaux souterraines à partir de données de terrain. Son utilisation n'est pas limitée au cadre proposé ni exclusivement aux données issues de simulations numériques.

---

### 1.2.2 Sous-objectif 2

Le deuxième objectif de ce projet est d'étendre les capacités de prévision spatio-temporelle des modèles RNG afin de prédire les données mesurées sur le terrain dans un système karstique alpin. Si les modèles RNG peuvent prévoir avec précision les niveaux des eaux souterraines lorsqu'ils sont entraînés avec des données simulées sous des conditions idéales, peuvent-ils également être utilisés pour prévoir avec précision les données de terrain dans des contextes hydrogéologiques très complexes? Ainsi, l'objectif principal de cette phase est de tester les capacités de prévision spatio-temporelle des modèles RNG à l'aide de données de terrain. Ici, la résolution et la quantité des données mesurées sont examinées afin d'estimer la robustesse de l'entraînement des RNG et donc, des capacités à faire des prévisions précises. L'utilisation de modèles d'AA comme les modèles RNG permet d'éviter complètement la construction de modèles physiques.

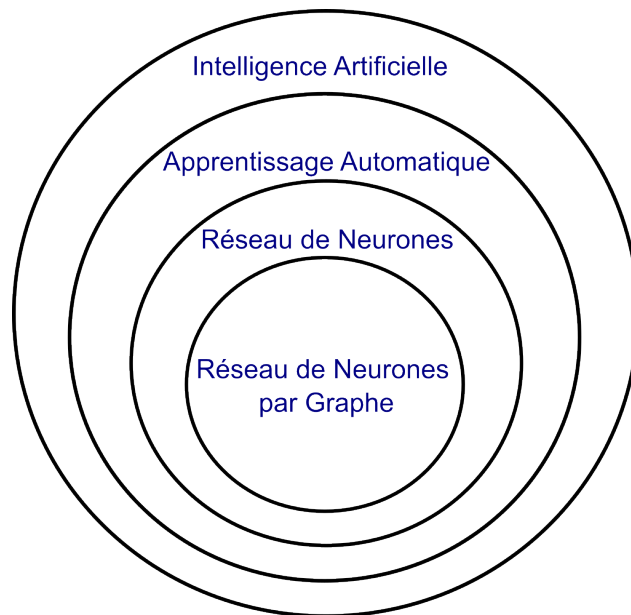
### 1.2.3 Sous-objectif 3

Le troisième objectif de cette thèse est de développer une fonction de perte universelle pouvant être utilisée par n'importe quel modèle d'AA, y compris les modèles RNG, afin d'améliorer la prévision des événements extrêmes dans les données mesurées sur le terrain. Cet objectif s'inscrit dans la continuité de l'Objectif 2. Si, dans l'Objectif 2, des prévisions précises des niveaux spatio-temporels des eaux souterraines peuvent être obtenues à l'aide de modèles d'AA tels que les modèles RNG, alors lors d'événements hydrologiques extrêmes comme les sécheresses, ces modèles peuvent-ils également prédire avec précision les niveaux des eaux souterraines? Avec les changements climatiques, les sécheresses et les inondations deviennent de plus en plus fréquentes. Ces événements menacent la santé et la gestion durable des aquifères. Il est donc essentiel de pouvoir prévoir ces phénomènes avec précision afin d'améliorer la gestion des eaux souterraines. Il est donc nécessaire de développer des stratégies d'entraînement des RNG permettant de tenir compte de ces variations extrêmes.

Dans la section suivante, nous allons décrire le cadre technique des algorithmes et des données utilisés durant cette thèse.

## 1.3 L'Apprentissage Automatique

L'AA est une approche axée sur les données, utilisée dans de nombreux domaines scientifiques. Il est particulièrement employé dans des domaines tels que le biomédical (Latif et al., 2019; Lundervold et al., 2018), la biologie (Irshad et al., 2020; Jin et al., 2021), la vision par ordinateur (Gardner et al., 2017; Gopalakrishnan et al., 2017), le traitement du langage naturel (Kang et al., 2016; Sak et al., 2014) et la traduction automatique (Vaswani et al., 2017).



**FIGURE 1.1 : La structure imbriquée de la classification des réseaux de neurones par graphe.**

Dans les géosciences, les chercheurs utilisent des méthodes d'AA pour résoudre des problèmes depuis les années 1950. La première méthode d'AA développée dans les géosciences pour l'exploitation minière fut le Krigeage par Krige (1951), tandis que d'autres méthodes comme les arbres de décision (Decision Trees) (Newendorp, 1976), le K-Means (Preston et al., 1964) et les modèles de chaînes de Markov (Markov Chains) (Krumbein et al., 1969) ont été parmi les premières adaptées à ce domaine. Au cours des dernières décennies, d'autres méthodes d'AA ont été développées et utilisées par les géoscientifiques. Parmi ces méthodes avancées figurent la forêt aléatoire (Random Forest) (Ho, 1995), les machines à vecteurs de support (Support Vector Machines) (Cortes et al., 1995), l'apprentissage par renforcement (Reinforcement Learning) (Watkins et al., 1989), ainsi que les réseaux de neurones profonds (RNP), incluant les réseaux convolutifs (Fukushima, 1988) et les réseaux récurrents (Hopfield, 1982). Plus récemment, l'utilisation des RNP a augmenté de façon exponentielle en géosciences (Dramschi, 2020). Dans certains cas, les chercheurs utilisent indifféremment le terme RNP et apprentissage profond (AP).

### **1.3.1 Réseau de Neurones**

Un RNG est un sous-ensemble des réseaux de neurones au sein de l'AA, lui-même faisant partie du domaine de l'intelligence artificielle (IA). Certains chercheurs en IA décrivent cette hiérarchie comme une poupée russe, où chaque thème est un composant du précédent (Figure 1.1).

Un RNP entièrement connecté est composé d'une couche d'entrée, suivie de plusieurs couches cachées, puis d'une couche de sortie. Un réseau de neurones comportant de multiples couches cachées est dit « réseau de neurones profond ». Chaque couche cachée d'un réseau neuronal

---

contient des neurones ou perceptrons, qui sont des nœuds dans lesquels s'effectue un calcul qui est transmis à un ou plusieurs neurones (Scarselli et al., 1998).

Le nombre de nœuds dans chaque couche cachée peut varier, et il n'est pas nécessaire qu'il soit identique d'une couche à l'autre. De plus, le nombre de nœuds dans les couches cachées n'a pas de corrélation directe avec le nombre de caractéristiques d'entrée, de sortie ou le nombre total de couches cachées. Actuellement, la sélection du nombre de nœuds dans les couches de réseau de neurones se fait empiriquement à l'aide de l'optimisation des hyperparamètres (Scarselli et al., 1998; Fine, 2006).

Il existe de nombreuses architectures de réseaux de neurones profonds. Les premiers réseaux étaient les réseaux entièrement connectés (fully connected). Le fait qu'un réseau soit entièrement connecté signifie que chaque nœud est relié à tous les autres nœuds de la couche suivante. Ce type de structure est connu sous le nom de réseau de neurones à propagation avant (RNPA) (Fully Connected Feedforward Neural Network). Un réseau de neurones convolutif (RNC) est un type de réseau « feedforward » qui contient des couches de convolutions suivies d'un réseau « fully connected ». En revanche, la structure d'un réseau de neurones récurrent (RNR) ne fonctionne pas en mode « feedforward », mais réinjecte ses sorties en lui-même, permettant ainsi la prise en compte des dépendances temporelles et des séquences de données.

Ces dernières années, les réseaux de neurones ont été développés et utilisés dans de nombreux domaines scientifiques. En géosciences, les chercheurs ont tiré parti de ces nouvelles avancées pour résoudre des problèmes en hydrologie (Afzaal et al., 2019; Yin et al., 2021), hydrogéologie (Li et al., 2021; Wunsch et al., 2021, 2018), géophysique (Côte et al., 2020; Dramsch et al., 2021; Lee et al., 2021) et pétrophysique (Aabø et al., 2020; Xu et al., 2019). En effet, la communauté des développeurs d'algorithmes d'AA est une communauté axée sur les algorithmes en accès libre (open-source).

En hydrogéologie, les réseaux de neurones ont été employés pour résoudre des problèmes liés à la gestion des ressources en eaux souterraines et de surface (Afzaal et al., 2019; Assem et al., 2017; Wunsch et al., 2021; Zhang et al., 2018), à l'estimation spatiale des propriétés hydrauliques (Wu et al., 2021), à l'estimation de la conductivité hydraulique à partir des propriétés du sol (Ghanbarian-Alavijeh et al., 2010) et à la caractérisation de la contamination lors du transport de masse (Li et al., 2021; Mo et al., 2019; Wu et al., 2021). En gestion des ressources en eau, Wunsch et al. (2021) et Zhang et al. (2018) ont utilisé avec succès des RNR pour prévoir les niveaux piézométriques en entraînant les modèles avec des données de terrain spécifiques telles que les précipitations, l'humidité, la température et la charge hydraulique. Zhang et al. (2018) ont mené une étude sur la gestion de l'eau agricole en prévoyant les profondeurs de la nappe phréatique à l'aide d'un type de RNR, Long Short-Term Memory (LSTM) et d'un RNPA. Ils ont constaté que le modèle LSTM surpassait le modèle RNPA pour les six bassins versants étudiés, en comparant les prévisions aux données mesurées. Par ailleurs, Wunsch et al. (2021) ont réalisé une étude comparative des prévisions des nappes phréatiques en utilisant un RNC, un LSTM et un réseau

---

non linéaire auto-régressif (nonlinear autoregressive network; NARX). Ils ont constaté que la méthode NARX, un RNR peu profond avec deux couches cachées, était plus précise pour estimer le niveau des nappes phréatiques avec moins de points de données, tout en suggérant que le RNC et le LSTM offrirait de meilleures performances avec davantage de points de données.

D'autres chercheurs, tels que Mo et al. (2019) et Li et al. (2021), ont utilisé des modèles RNP pour identifier les sources de contamination à l'aide de modèles directs synthétiques en 1D et 2D. Mo et al. (2019) ont utilisé un RNC avec un modèle encodeur-décodeur, tandis que Li et al. (2021) ont eu recours à un LSTM. Les deux équipes de chercheurs ont réussi à déterminer avec un haut niveau de confiance la source de contamination grâce à leurs modèles.

Le développement le plus récent des DNN en géosciences concerne les réseaux de neurones informés par la physique (Physics-Informed Neural Networks; PINN) (He et al., 2020, 2021a; Tartakovsky et al., 2020; He et al., 2021b; Wu et al., 2021). Dans ces modèles, les équations différentielles qui gouvernent les processus physiques (par exemple, l'équation de flux ou l'équation d'advection-dispersion) sont incorporées dans la fonction de perte pour être minimisées, au lieu d'utiliser uniquement des données étiquetées (les données dites étiquetées correspondent, la plupart du temps, à des exemples identifiés manuellement par un expert). Cette approche constitue un bon compromis entre les méthodes d'AA basées exclusivement sur les données et les modèles physiques, car elle nécessite moins de données pour l'entraînement. Cette méthode est particulièrement utile en géosciences, car les mesures de terrain sont chronophages et coûteuses, rendant difficile la constitution de grands ensembles de données.

Un autre défi en géosciences réside dans la complexité des phénomènes physiques des systèmes naturels, ce qui rend difficile la construction de modèles physiques précis. Les réseaux de neurones informés par la physique ont été développés par Raissi et al. (2017a,b). He et al. (2020) ont repris cette idée et ont publié une architecture de réseau neuronal adaptée au transport en milieu souterrain. He et al. (2021a) ont utilisé ce nouveau réseau de neurones informé par la physique en intégrant les équations d'advection-dispersion directe et inverse pour tester des modèles hétérogènes et anisotropes en 1D et 2D, avec une plage de nombres de Péclet. Leur modèle a été entraîné en intégrant les résidus des équations différentielles partielles d'advection-dispersion, ainsi que les conditions aux limites de Dirichlet et de Neumann dans la fonction de perte à minimiser.

Tartakovsky et al. (2020) ont utilisé un RNPA avec deux couches cachées pour apprendre les paramètres et les relations constitutives dans l'écoulement en milieu souterrain. Ils ont sélectionné aléatoirement des points sur un modèle synthétique 2D à éléments finis représentant un champ de conductivité hydraulique et un champ d'état, tel que la charge hydraulique ou la pression capillaire, comme données d'entrée. Le modèle a été entraîné en minimisant la fonction de perte à l'aide des résidus des équations d'écoulement en régime permanent, ainsi que des conditions aux limites de Neumann et de Dirichlet. Ils ont comparé la méthode informée par la physique à un RNPA basé sur les données et à une méthode d'estimation de maximum a posteriori. Ils ont constaté que leur

---

modèle de réseau de neurones informé par la physique surpassait l'estimation des paramètres de maximum a posteriori des équations de diffusion linéaire et non linéaire. En revanche, ils ont trouvé que leur méthode était plus précise pour estimer ces paramètres lorsque 250 points d'échantillonnage ou plus étaient disponibles. Ils ont proposé trois explications dont l'une est que le modèle physique ne reproduit pas un niveau d'hétérogénéité suffisant pour représenter les systèmes naturels alors que les modèles basés sur les données, si celles-ci contiennent l'information, peuvent servir à modéliser des systèmes très complexes.

### 1.3.2 Réseau de Neurones par Graphe

L'un des aspects cruciaux en géosciences est de tenir compte de la corrélation spatiale entre les mesures. Lorsque les données sont mesurées sur une grille régulière, comme les données géophysiques, la structure spatiale peut être prise en compte par des réseaux convolutifs (LeCun et al., 1995; Alzubaidi et al., 2021; Kawahara et al., 2010). Toutefois, lorsque les données sont acquises de manière non régulière comme les puits en hydrogéologie, seuls les réseaux de neurones par graphe peuvent prendre en charge les corrélations spatiales et temporelles (Zhou et al., 2020). Les RNG ont été proposés pour la première fois par Scarselli et al. (2008). Ceux-ci utilisent des réseaux de neurones existants tels qu'un RNC ou un RNR pour traiter les informations sur une structure en graphe. La structure du graphe est composée de nœuds et d'arêtes. Les nœuds contiennent l'information qui est propagée à travers les arêtes du graphe. Par exemple, le réseau de neurones par graphe convolutionnel (Graph Convolutional Network; GCN) de Kipf et al. (2016) est l'un des RNG les plus couramment utilisés. Dans ce réseau, les données sont représentées sous forme de graphe, et il est composé d'une couche d'entrée, de deux couches cachées et d'une couche de sortie. Les activations se produisent entre la première couche cachée et la deuxième couche cachée, ainsi qu'entre la deuxième couche cachée et la couche de sortie (Figure 1.2). Ce réseau a été initialement conçu pour la classification semi-supervisée des nœuds dans les graphes.

Dans le contexte de cette thèse, les nœuds représentent des puits ou des stations de surveillance, et les arêtes représentent la connectivité hydraulique entre ces puits ou stations (Figure 1.3). Un graphe peut être orienté ou non orienté, c'est-à-dire que l'information transmise d'un nœud à l'autre peut être unidirectionnelle (directed) ou être bidirectionnelle (undirected). Les arêtes d'un graphe peuvent également être pondérées ou non pondérées. Il existe de nombreuses façons de construire un graphe, ce qui nécessite des connaissances ou des informations préalables sur le système ou sur les données étudiées. Par exemple, dans cette thèse, j'ai adapté l'équation de covariance gaussienne Li et al. (2017) pour construire mon graphe de puits (Figure 1.3); la Figure 1.3 représente un graphe pondéré (weighted) et non orienté (undirected).

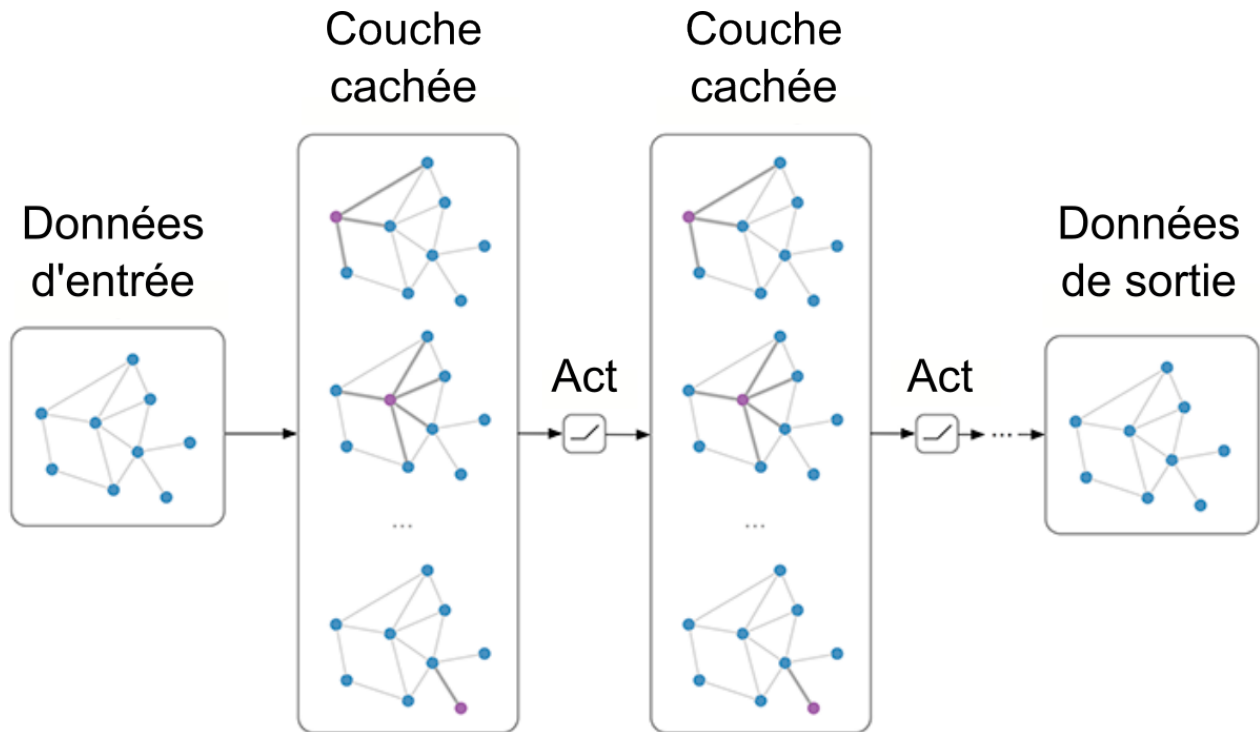
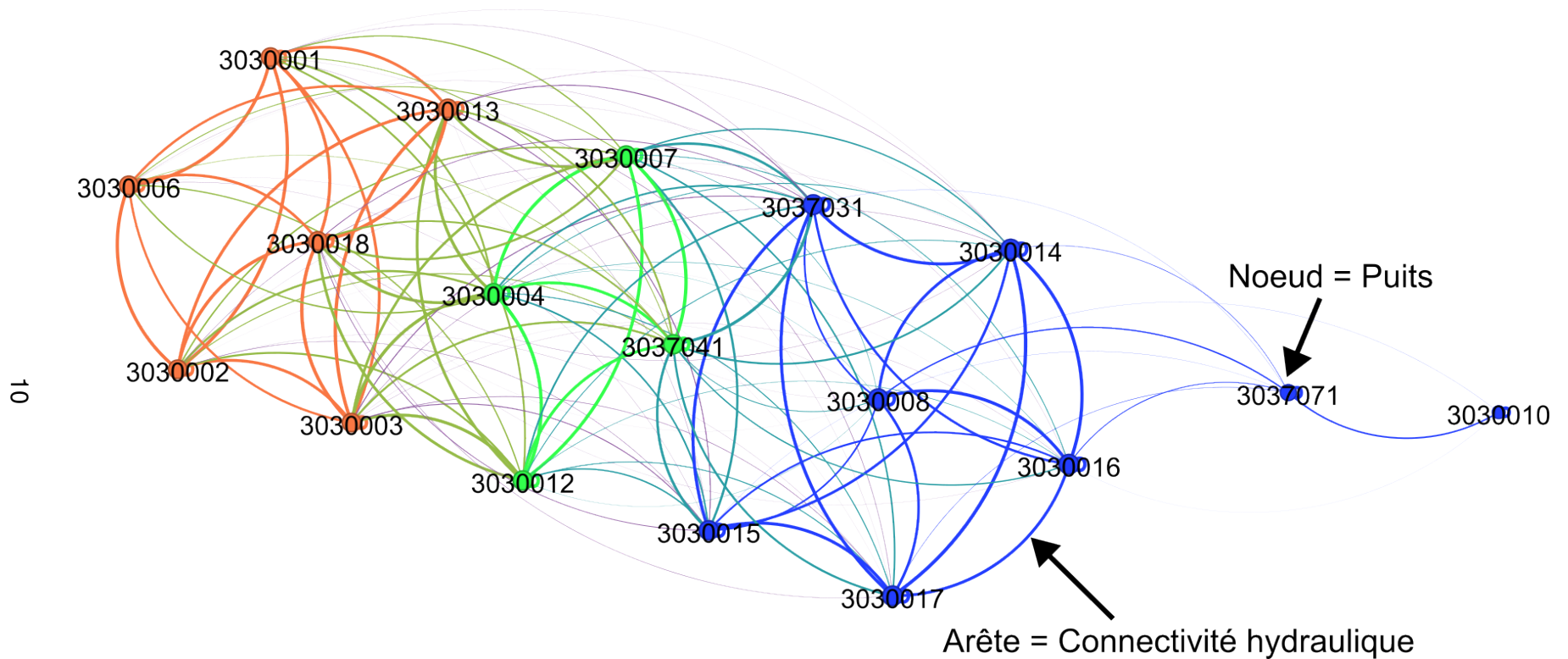


FIGURE 1.2 : La structure du modèle GCN. La figure est modifiée à partir Kipf et al. (2016)

Un graphe peut être représenté mathématiquement par une matrice d'adjacence. La matrice d'adjacence,  $A \in \mathbb{R}^{N \times N}$ , où  $N$  est le nombre de nœuds. Cette matrice représente les connexions spatiales entre tous les nœuds du réseau.



**FIGURE 1.3 : Projection du graphe dans l'espace des données. Chaque nœud représente un puits, les arêtes représentent la connectivité hydraulique entre les puits. La couleur spécifique des nœuds définit un voisinage de nœuds qui sont les plus proches dans l'espace des données. Comme ce graphe est une projection dans l'espace des données, les positions des nœuds dans ce graphe n'ont aucune relation avec l'emplacement réel des puits dans le monde physique. L'épaisseur des arêtes représente la connectivité des puits.**

---

Les RNG peuvent être classés en trois types de tâches; axées sur les graphes, axées sur les nœuds (Scarselli et al., 2008) et, plus récemment, axées sur les arêtes (Zhou et al., 2020). Les RNG partagent des similitudes avec d'autres réseaux neuronaux en ce qui concerne les types d'apprentissage, y compris les méthodes supervisées (méthodes qui requièrent des données étiquetées), semi-supervisées (méthodes qui nécessitent que certaines données soient étiquetées) et non supervisées (sans données étiquetées).

Actuellement, il existe deux types de convolutions sur les graphes, soit les convolutions spectrales et spatiales. La convolution spectrale transforme les nœuds dans le domaine spectral et calcule la décomposition en valeurs propres de la matrice d'adjacence du graphe. En revanche, la convolution spatiale opère directement sur la structure du graphe en effectuant des convolutions sur le voisinage local de chaque nœud (Defferrard et al., 2016).

Les RNG peuvent appartenir à deux catégories d'apprentissage, transductif ou inductif. L'apprentissage transductif nécessite un ensemble fixe de nœuds et d'arêtes présent lors de l'entraînement du modèle, permettant la prévision ou la prédiction simultanée de tous les nœuds. Cependant, cette approche ne se généralise pas aux nœuds non vus pendant l'entraînement (Scarselli et al., 2008; Hamilton et al., 2017b). À l'inverse, l'apprentissage inductif peut se généraliser à des nœuds non vus lors de l'entraînement. L'apprentissage inductif peut reconnaître la structure de voisinage d'un nœud et comprendre son rôle dans les contextes local et global du graphe (Hamilton et al., 2017a,b). En d'autres termes, les PNG inductifs peuvent apprendre à partir de structures de graphe changeantes, où les nœuds et les arêtes peuvent être ajoutés ou supprimés. Bien que la plupart des RNG, tels que le GCN, soient transductifs, il existe quelques RNG inductifs, comme le Graph WaveNet (GWN) (Wu et al., 2019), GraphSAGE (Hamilton et al., 2017a), le Diffusion Convolutional Neural Network appliqué aux graphes (Li et al., 2017; Atwood et al., 2016) et le Graph Isomorphism Network (Xu et al., 2018).

En hydrogéologie, les RNG sont encore très peu utilisés malgré leur fort potentiel pour assimiler des séries temporelles non structurées spatialement. Cela peut s'expliquer par la complexité de l'entraînement des RNG mais aussi par les mesures hydrogéologiques en tant que telles. En effet, les séries temporelles de longue durée ne sont pas courantes en hydrogéologie bien que, récemment, de nombreux pays commencent à acquérir des données de niveaux d'eau grâce à des sondes automatisées ou des mesures répétées dans le temps (Shapiro et al., 2022). Cependant, Bai et al. (2023) ont utilisé le modèle GWN (Wu et al., 2019b) pour prévoir les niveaux des nappes phréatiques des puits en Colombie-Britannique, au Canada. Taccari et al. (2024) ont développé un RNG innovant pour prédire les niveaux des nappes phréatiques dans des puits situés aux Pays-Bas. Leur réseau peut non seulement prévoir les séries temporelles futures, mais il peut également être utilisé pour l'imputation de séries temporelles manquantes. Sun et al. (2021) ont utilisé des données de débit fluvial provenant du jeu de données Catchment Attributes and Meteorology for Large-sample Studies (CAMELS) pour évaluer les performances des RNG. Les RNG testés dans leur étude incluent le ChebNet (Defferrard et al., 2016), le GCN (Kipf et al., 2016) et le

---

GWN (Wu et al., 2019). Ils ont testé l'apprentissage transductif des RNG sur des bassins jaugés et l'apprentissage inductif sur des bassins non jaugés pour prévoir le débit fluvial.

### **1.3.3 Utilisation des RNG dans cette thèse**

Dans cette thèse, j'ai adapté des architectures de RNGs pour prédire les niveaux spatio-temporels des niveaux d'eau dans deux types d'aquifères, soit des systèmes aquifères granulaires et rocheux au Québec, Canada et des systèmes aquifères karstiques alpins en Suisse. Pour entraîner un RNG, nous devons d'abord construire un graphe afin de représenter nos données sur les nappes phréatiques. Comme mentionné précédemment, il existe plusieurs façons de construire un graphe. En général, un graphe est construit en utilisant les connaissances préalables sur le système.

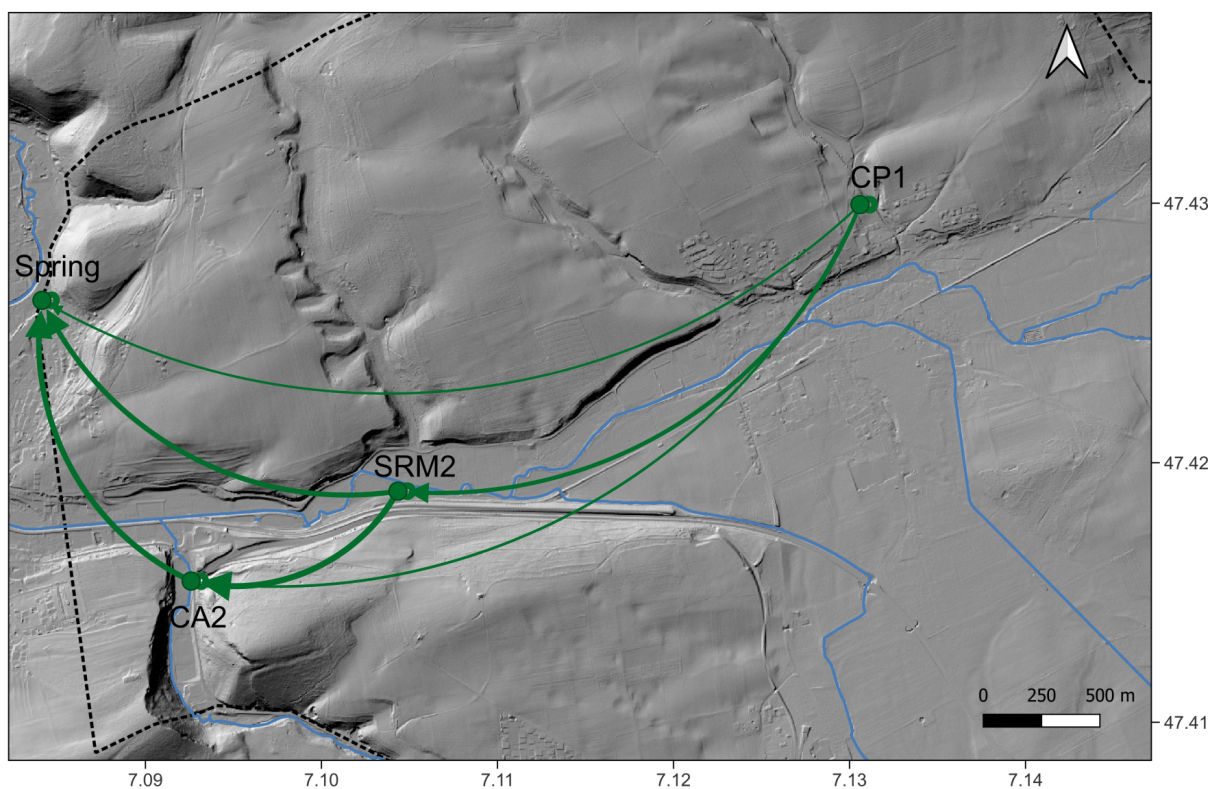
Dans mon cas, j'ai utilisé des données provenant des puits (aquifère canadien) et des stations de surveillance (karsts suisses) pour entraîner les RNG. Par conséquent, les puits et les stations sont représentés par des nœuds dans le graphe, tandis que la connectivité hydraulique entre eux est représentée par des arêtes. La construction de ces arêtes dépend du système aquifère. Par exemple, dans un aquifère karstique, où l'écoulement de l'eau dans les conduits est mesurable et dont la direction est connue, nous pouvons construire un graphe orienté (directed), en utilisant la direction de l'écoulement pour définir l'orientation des arêtes. Les arêtes peuvent être pondérées en fonction des calculs de distance par rapport aux stations de surveillance (Figure 1.4).

Dans le cas de l'aquifère canadien, il n'y a pas de directionnalité a priori car les crépines des puits peuvent être dans des zones connectées ou non et les gradients hydrauliques dépendent du sens de l'écoulement naturel mais aussi des pompages.

En utilisant ces graphes, nous pouvons intégrer des données de séries temporelles dans chaque nœud afin d'entraîner le RNG. Actuellement, il n'existe pas de méthode standard pour la construction de graphes; plusieurs structures de graphes pourraient devoir être développées et testées empiriquement ou utiliser un graphe dynamique pouvant s'auto-adapter pendant le processus d'entraînement du modèle afin d'obtenir les meilleurs résultats.

## **1.4 Résumé des articles**

Chaque objectif a été développé dans un article. Ci-dessous, les articles sont résumés et les contributions de chacun d'eux sont décrites et expliquées.



**FIGURE 1.4 : Un exemple de graphe de système karstique. Il s'agit d'un graphe pondéré et orienté avec des nœuds fixés à leur emplacement physique. Les flèches des arêtes vertes indiquent la direction générale de l'écoulement des eaux souterraines entre chaque station.**

### 1.4.1 Contributions de l'article 1

Le titre du premier article est : Cadre de réseau de neurones par graphe pour la prévision spatio-temporelle des niveaux de nappe phréatique. Dans cet article, nous présentons un cadre de RNG qui sert de complément et de substitut prédictif au modèle hydrogéologique numérique, permettant de prévoir les niveaux d'eau souterraine dans les puits de surveillance et de pompage sous différentes contraintes de pompage. Cette architecture repose sur deux composantes clés, soit un réseau GCN-LSTM pour la prédiction des niveaux d'eau souterraine et une analyse en composantes principales (ACP) pour la réduction de la dimension des données. Cet algorithme peut être décomposé en 5 étapes principales :

L'étape 1 consiste à réaliser des simulations numériques afin de générer des données synthétiques pour l'entraînement du réseau GCN-LSTM et la construction de la matrice d'adjacence. Des simulations transitoires sont effectuées sous deux scénarios de pompage distincts, générant ainsi les jeux de données 1 et 2.

---

L'étape 2 est dédiée à la construction de la matrice d'adjacence, qui représente la connectivité spatiale des puits. La connectivité des puits est déterminée en calculant les distances euclidiennes entre chaque paire de puits, puis en appliquant un filtre de rayon. Le rayon optimal est sélectionné après des tests réalisés au sein du réseau GCN-LSTM.

L'étape 3 concerne la réduction de la dimension des séries temporelles, où les niveaux transitoires de l'eau souterraine et les débits de pompage sont transformés à l'aide de l'encodage ACP, permettant de projeter les deux séries temporelles de l'espace euclidien vers l'espace ACP. Cette étape est essentielle pour réduire la dimension du tenseur de données en vue de l'entraînement du modèle.

L'étape 4 porte sur l'optimisation et l'entraînement du réseau GCN-LSTM, en utilisant la composante ACP du niveau transitoire de l'eau souterraine issue de l'étape 3 pour entraîner le modèle.

L'étape 5 applique la transformation inverse de l'ACP, en convertissant les prédictions du GCN-LSTM de l'espace ACP vers leur espace d'origine.

Traditionnellement, les modèles hydrogéologiques basés sur des principes physiques ont été largement utilisés pour la régulation et la gestion des eaux souterraines. Malheureusement, ces modèles physiques sont souvent très lourds en temps de calcul et nécessitent une expertise substantielle pour leur construction et leur simulation. Ces méthodes traditionnelles peuvent être lentes et inefficaces, en particulier lorsqu'elles doivent traiter des systèmes hydrogéologiques complexes soumis à diverses pressions climatiques et anthropiques. De plus, les méthodes géostatistiques, couramment utilisées pour prévoir les variations spatiales des niveaux d'eau souterraine, reposent sur des hypothèses de stationnarité et de fortes corrélations entre les variables. Ces hypothèses ne se vérifient pas toujours dans les systèmes aquifères réels. Ces défis soulignent la nécessité de modèles plus efficaces, basés sur les données, capables de gérer la nature dynamique et complexe des systèmes d'eau souterraine.

À cette fin, le cadre GCN-LSTM proposé combine deux approches. La GCN capture les relations spatiales entre les puits en représentant la connectivité hydraulique des puits sous forme de graphe, où les puits sont traités comme des nœuds et leurs connexions spatiales comme des arêtes. Cette représentation basée sur un graphe permet d'intégrer des informations spatiales telles que les distances entre les puits, et la conductivité hydraulique. De son côté, la composante LSTM permet de prendre en compte le traitement des séries temporelles, capturant efficacement les variations temporelles multi-échelles de la fluctuation des niveaux d'eau souterraine.

Comme dans la plupart des cas d'études des systèmes aquifères, il n'existe pas d'échantillonnage spatio-temporel à l'échelle nécessaire pour entraîner un réseau de neurones. L'entraînement du modèle repose sur des données synthétiques générées à partir du modèle hydrogéologique numérique de Ville Mercier, une région située dans le sud du Québec, au Canada. Ce modèle hydrogéologique a été développé pour optimiser le traitement et le piégeage des eaux souterraines d'un site contaminé (Claprod et al., 2023). Cette approche est supportée par de nombreuses

---

études prouvant que l'entraînement sur des données synthétiques réalistes permet d'entraîner de manière efficace des RNP, même mieux qu'avec des bases de données réelles. Ce modèle numérique simule le comportement des eaux souterraines sous différents scénarios de pompage, en intégrant l'hétérogénéité géologique complexe de la région, notamment le socle rocheux fracturé et l'aquifère de surface consistant en un esker de sable et gravier entouré d'argile. Deux scénarios de pompage distincts ont été appliqués, l'un avec des débits de pompage augmentant progressivement et l'autre avec des débits variables qui augmentent et diminuent au fil du temps. Ces scénarios simulent les schémas de pompage utilisés pour le traitement et le piégeage des eaux souterraines et permettent d'évaluer la robustesse du modèle GCN-LSTM dans différentes conditions de stress.

Afin de pouvoir utiliser l'architecture de GCN-LSTM, nous avons appliqué une analyse en composantes principales pour réduire la dimension des données. En effet, étant donné que les données d'entrée consistent de séries temporelles multidimensionnelles provenant de plusieurs puits, l'analyse en composantes principales aide à réduire la dimension des données, permettant d'être prise en compte par les tenseurs du réseau. Cette transformation permet aussi la transformation inverse dans l'espace original.

Les résultats de l'étude démontrent que le modèle GCN-LSTM fournit des prévisions des niveaux d'eau souterraine très précises et efficaces sur le plan informatique. Comparé aux modèles traditionnels, qui peuvent nécessiter des jours pour produire des résultats, le cadre RNG génère des prévisions en quelques secondes ou minutes tout en capturant la dynamique spatio-temporelle de l'écoulement des eaux souterraines, en prédisant avec précision les variations des niveaux d'eau sous différents scénarios de pompage. Des indicateurs d'évaluation tels que l'erreur moyenne quadratique (EMQ) et l'erreur moyenne absolue (EMA) ont confirmé la précision des prévisions du modèle. Le modèle a également obtenu des scores de coefficient de détermination ( $R^2$ ) élevés (0,960 pour le scénario de pompage 1 et 0,926 pour le scénario de pompage 2), indiquant de fortes corrélations entre les niveaux d'eau prédits et simulés.

Malgré ses bonnes performances, le cadre présente certaines limitations. De plus, la performance du modèle dépend également d'un réglage minutieux des hyperparamètres, qui nécessite une expertise technique. Mais une fois le modèle ajusté et entraîné, il peut être utilisé par quiconque. La méthode a été développée pour prédire l'écoulement des eaux souterraines, basée sur un modèle numérique complexe supporté par une caractérisation de terrain détaillée. Toutefois, si le modèle numérique est erroné, les prédictions pourraient être biaisées par ces erreurs.

Un atout important de ce cadre réside dans son potentiel d'applications concrètes. Le modèle peut être adapté pour fonctionner avec des données réelles de terrain, en utilisant des mesures provenant de puits de surveillance et de pompage pour entraîner le réseau. Cette flexibilité signifie que la méthode pourrait être utilisée et adaptée par des spécialistes pour que les municipalités et les décideurs de petites municipalités puissent gérer les ressources en eau de manière plus

---

efficace, en particulier dans les zones où des décisions rapides sont essentielles en raison de la demande fluctuante ou des conditions de sécheresse.

L'étude conclut que le cadre GCN-LSTM représente une avancée significative dans la prévision des niveaux d'eau souterraine, en offrant une solution efficace sur le plan informatique, précise et évolutive pour la gestion des systèmes hydrogéologiques complexes. Le modèle pourrait être adapté en tant que jumeau numérique du modèle physique, offrant un outil d'aide à la décision en temps réel pour la planification de l'utilisation des sols, l'allocation des ressources et l'analyse de la durabilité des eaux souterraines. En réduisant le temps de calcul et en minimisant le besoin d'expertise spécialisée lors de l'étape de prédiction, le cadre a le potentiel de démocratiser la modélisation des eaux souterraines, la rendant accessible à un plus large éventail d'utilisateurs, notamment lors des décisions concernant l'aménagement du territoire.

### 1.4.2 Contributions de l'article 2

Le titre du deuxième article est : Comparaison des performances prédictives des modèles d'apprentissage profond selon la résolution des séries temporelles pour une estimation précise du débit des sources karstiques. Dans l'article 2, nous étudions l'efficacité des modèles RNP dans la prévision des débits des sources karstiques, qui sont essentiels pour la gestion des ressources en eau dans les régions dépendantes des aquifères karstiques. Ici, nous nous intéressons à l'utilisation de données mesurées sur le terrain pour entraîner des modèles RNP tels que les modèles RNG afin de tester les capacités de prévision spatio-temporelle précise des niveaux d'eau souterraine. Nous avons choisi un ensemble de données sur les débits d'une source karstique dynamique pour les tests, car les systèmes karstiques sont notoirement difficiles à modéliser physiquement en raison de leurs dynamiques d'écoulement complexes et de leur forte hétérogénéité. Pour surmonter ces défis, nous avons évalué trois modèles RNP : l'auto-encodeur LSTM, l'auto-encodeur Gated Recurrent Unit (GRU) et le GWN. Les modèles ont été entraînés et testés à l'aide de séries chronologiques collectées à partir d'un ensemble de données provenant d'un système karstique suisse, l'objectif principal étant de déterminer leur précision prédictive à différentes résolutions temporelles et tailles d'ensemble d'entraînement.

La configuration expérimentale consistait à collecter des données sur les niveaux d'eau souterraine provenant de quatre stations de surveillance entre 2015 et 2018, avec des mesures enregistrées toutes les heures. Les données ont ensuite été rééchantillonnées à des intervalles de 4, 8, 12 et 24 heures pour évaluer les performances des modèles à différentes résolutions ou échantillonnage temporel. La capacité de prévision de chaque modèle a été testée sur trois longueurs de prévision différentes : 3, 6 et 12 pas de temps. Des métriques d'évaluation telles que l'erreur moyenne quadratique (EMQ), l'erreur moyenne absolue (EMA), l'erreur moyenne absolue en pourcentage (EMAP) et le  $R^2$  ont été utilisées pour mesurer les performances des modèles à différentes résolutions de données et longueurs de prévision.

---

Les résultats montrent qu'avec des données à haute résolution, tous les modèles ont obtenu des prévisions presque parfaites, avec des valeurs de  $R^2$  proches de 0,99. Cependant, à mesure que la résolution des données diminue, la performance des prévisions diminue, en particulier pour des horizons de prévision plus longs. Le modèle GWN s'est révélé particulièrement performant pour gérer les prévisions à long terme et les données à faible résolution par rapport aux auto-encodeurs. En outre, le temps de calcul requis pour le plus grand ensemble de données (données échantillonnées toutes les heures avec une prévision de 3 pas de temps en avant) a été enregistré pour tous les modèles. L'auto-encodeur GRU s'est avéré être le plus efficace sur le plan computationnel, nécessitant un peu plus de 4 minutes pour entraîner le plus grand ensemble de données. L'auto-encodeur LSTM a pris environ 10 minutes, tandis que le GWN a pris environ 28 minutes.

Les résultats révèlent que le GWN a systématiquement surpassé les modèles auto-encodeurs dans la plupart des scénarios, en particulier lorsque les données d'entrée avaient une faible résolution temporelle ou lorsque l'horizon de prévision s'étendait davantage dans le futur. Lorsque des données à haute résolution (avec des mesures horaires) ont été utilisées, les trois modèles ont obtenu d'excellentes performances, avec des  $R^2$  supérieurs à 0,98. Cependant, à mesure que la résolution des données diminuait, la précision des prévisions chutait de manière significative pour tous les modèles, le GWN maintenant de meilleures performances que les auto-encodeurs LSTM et GRU. L'étude met en évidence le fait que la résolution temporelle des données est un facteur plus important pour la précision des prévisions que la taille de la série temporelle. Cela ne signifie pas que la taille des données n'est pas importante pour des prévisions précises; un certain volume de données est nécessaire pour entraîner les modèles efficacement.

En conclusion, la résolution des données joue un rôle crucial dans la précision des prévisions de la dynamique des niveaux d'eau dans les systèmes karstiques. Bien que les trois modèles aient bien performé avec des données à haute résolution, le modèle GWN a montré une robustesse supérieure, en particulier lors de prévisions sur des horizons temporels plus longs ou avec des données à faible résolution temporelle. Dans les systèmes moins dynamiques, cependant, des données à résolution inférieure pourraient suffire pour des prévisions précises, ce qui permettrait potentiellement de réduire les exigences computationnelles.

Nous recommandons que les recherches futures se concentrent sur l'intégration d'estimations d'incertitude dans les prévisions des modèles en utilisant des méthodes telles que le dropout de type chaîne de Markov Monte Carlo. En outre, tester les modèles sur des ensembles de données plus volumineux pourrait fournir des informations sur l'effet de la taille des données sur la précision des prévisions à différentes résolutions. Nous suggérons également d'explorer des modèles hybrides combinant des connaissances basées sur les processus physiques avec des approches basées sur les données pour améliorer à la fois la précision des prévisions et l'interprétabilité dans des systèmes hydrogéologiques complexes. Dans l'ensemble, notre étude offre des perspectives sur la manière dont les modèles d'AP peuvent être efficacement appliqués pour prévoir les débits

---

des sources karstiques, avec des implications importantes pour la gestion de l'eau et les efforts de conservation.

### 1.4.3 Contributions de l'article 3

Le titre du troisième article est : Développement d'une fonction de perte pour réseaux de neurones profonds visant la prévision des événements extrêmes dans les séries temporelles. L'objectif est d'améliorer la gestion des ressources en eau souterraine en affinant les prévisions des événements extrêmes, tels que les inondations et les sécheresses, qui restent difficiles à anticiper avec les fonctions de perte traditionnelles comme l'EMA ou l'EMQ. Dans l'article 3, je propose une nouvelle fonction de perte appelée fonction de perte extrême (Extreme Loss Function), conçue spécifiquement pour les modèles RNP afin d'améliorer la prévision des événements extrêmes dans les séries temporelles. Deux ensembles de données, l'un provenant du système karstique de Milandre en Suisse et l'autre du bassin versant de la Yamaska au Québec, Canada, ont été utilisés pour évaluer l'efficacité de cette nouvelle fonction de perte.

Notre fonction de perte extrême améliore la capacité des modèles RNP à prédire les événements extrêmes en attribuant des pénalités plus sévères aux grandes erreurs de prédiction associées aux valeurs extrêmes. Lorsque les points de données se situent dans la plage normale, la fonction se comporte de manière similaire à une fonction de perte EMA standard. Cependant, pour les points de données qui s'écartent considérablement de la moyenne, la fonction applique des poids supplémentaires à ces erreurs, ce qui pousse le modèle à réévaluer ses prédictions via le processus de rétropropagation lors de l'optimisation du modèle. Ce mécanisme permet au modèle de se concentrer davantage sur la prédiction précise de ces événements rares, améliorant ainsi ses performances.

Les données manquantes dans les séries temporelles représentent un problème courant en géosciences. DeeSse (Mariethoz et al., 2010, 2015; Straubhaar et al., 2021) permet de générer des réalisations probabilistes des données manquantes respectant les fluctuations des séries temporelles mesurées. L'algorithme DeeSse repose sur une méthode de statistiques multipoints (Multiple Point Statistics; MPS). Au lieu de remplacer les données manquantes avec des moyennes simples ou des zéros, cette méthode génère des données probabilistes en empruntant des motifs au jeu de données d'entraînement tout en incorporant des variables auxiliaires telles que la température, les précipitations et les tendances saisonnières. Cette technique garantit que la série chronologique conserve sa variabilité naturelle et renforce la fiabilité des prévisions des modèles. Nous utilisons cette méthode de statistiques multipoints pour combler les lacunes dans nos données, et la série temporelle complète est ensuite utilisée pour l'entraînement du modèle.

Deux modèles de RNP ont été utilisés pour tester l'efficacité de la fonction de perte extrême. Le premier est le modèle GWN, un réseau de neurones par graphe qui combine un réseau de convolution temporelle avec un GCN. Il est particulièrement adapté aux données spatio-temporelles, car

---

il peut capturer à la fois les relations spatiales avec le graphe et les données temporelles grâce au réseau convolutif. Le second modèle consiste en un auto-encodeur GRU, qui est une structure de RNR conçue pour le traitement séquentiel des données de séries temporelles. Cette architecture utilise un encodeur-décodeur qui compresse la séquence d'entrée en un vecteur le plus creux possible et de longueur fixe, ce qui maximise le contenu en information pertinent du signal d'entrée.

Comme mentionné précédemment, cette étude a été menée à partir de données provenant de deux zones géographiques distinctes. Le système karstique de Milandre est un aquifère situé en Suisse. Il consiste en un total de quatre stations de surveillance : trois stations surveillant des sources (Saivu, Font et Bâme) et une autre station en amont d'un cours d'eau pérenne appelé Milandrine. Ces sources permettent d'inférer la dynamique du débit, influencée par les précipitations saisonnières et la fonte des neiges, ce qui en fait un cas idéal pour tester la capacité des modèles à prédire des événements de débit élevé. Dans le bassin versant de la rivière Yamaska, situé au Québec, Canada, dix-huit puits d'observation sont utilisés pour suivre les niveaux d'eau souterraine. Les données issues de ces puits, qui montrent plusieurs épisodes de baisse de la nappe phréatique, permettent d'évaluer la capacité des modèles à prévoir les périodes d'étiages.

Les performances des modèles ont été évaluées à l'aide de trois indicateurs clés : le  $R^2$ , l'EMQ et l'EMA. Le modèle GWN a montré des améliorations notables selon ces trois indicateurs lorsqu'il a été entraîné avec la fonction de perte extrême, en particulier pour les prévisions à court terme. En revanche, le modèle GRU a montré des gains minimes avec la fonction de perte extrême, car il est déjà capable de gérer efficacement la détection des valeurs extrêmes sans nécessiter l'aide de cette fonction spécialisée.

Les résultats ont démontré que le modèle GWN a obtenu des performances significativement meilleures lorsqu'il a été entraîné en utilisant la fonction de perte extrême, en particulier pour la prédiction des débits extrêmes dans le jeu de données du système de Milandre. Dans le jeu de données de Yamaska, le modèle a également montré de légères améliorations dans la prédiction des faibles niveaux d'eau souterraine. D'autre part, le modèle Auto-Encodeur GRU n'a montré que de légères améliorations lors de l'utilisation de la fonction de perte extrême pour les prévisions à court terme avec le jeu de données de Milandre. Pour le jeu de données de Yamaska, les performances du modèle GRU sont restées globalement inchangées, car il était déjà bien adapté à la gestion des prédictions de valeurs extrêmes grâce à sa structure auto-encodée.

En conclusion, la fonction de perte extrême permet d'améliorer la capacité des modèles RNP, en particulier le GWN, à prévoir les événements extrêmes dans les séries chronologiques, un facteur crucial pour une gestion efficace des ressources en eau souterraine. Bien que les améliorations pour l'auto-encodeur GRU soient marginales, en particulier sur le jeu de données de la Yamaska, ce résultat était attendu étant donné que le modèle GRU est intrinsèquement mieux adapté à la gestion des prédictions des valeurs extrêmes. Il est important de noter que la résolution des données est un facteur déterminant pour obtenir des prévisions précises; des données

---

de haute résolution réduisent considérablement le besoin de fonctions de perte spécialisées telles que la fonction de perte extrême développée dans cette étude.

## **1.5 Principales Contributions**

Les principales contributions scientifiques de ma thèse sont des codes en accès libre qui sont disponibles sur Github (<https://github.com/XiaoXia10>). Les codes développés pour l'Objectif 1 sont déjà en accès libre et peuvent être utilisés avec les données qui sont, elles, aussi en accès libre. De plus, la fonction de perte extrême ainsi que tous les modèles d'AA développés dans ce travail sont en accès libre, garantissant ainsi leur accessibilité et la reproductibilité de mes travaux.

La fonction de perte extrême est conçue pour être intégrée à tout modèle de réseau de neurones traitant des problèmes de régression mais dont la distribution des données contient des valeurs extrêmes mais très significatives comme pour la gestion de l'accessibilité de l'eau pour les populations ou pour le développement économique. Cette approche est non seulement utile pour l'hydrogéologie, mais aussi pour tout domaine scientifique nécessitant une meilleure prévision des séries temporelles avec des valeurs extrêmes. En mettant ces outils en accès libre, cette recherche contribue à l'avancement des applications de l'AA dans divers domaines scientifiques.

## **2 ARTICLE I : GRAPH NEURAL NETWORK FRAMEWORK FOR SPATIO-TEMPORAL GROUNDWATER LEVEL FORECASTING**

---

### **Cadre de réseau de neurones par graphe pour la prévision spatio-temporelle des niveaux de nappe phréatique**

Xiao Xia Liang<sup>1</sup>, Erwan Gloaguen<sup>1</sup>, Maxime Claprood<sup>2</sup>, Daniel Paradis<sup>1</sup>, Dany Lauzon<sup>3</sup>

<sup>1</sup> Centre Eau Terre Environnement, Institut national de la recherche scientifique, Québec, Canada

<sup>2</sup> Département des sciences appliquées, Université du Québec à Chicoutimi, Canada

<sup>3</sup> Département des génies civil, géologique et des mines, Polytechnique Montréal, Montréal, Canada

### **Contributions des auteurs**

Xiao Xia Liang : Rédaction du manuscrit, méthodologie, programmation, analyse des données et conceptualisation.

Erwan Gloaguen : Supervision, méthodologie, programmation et révision du manuscrit.

Maxime Claprood : Supervision, construction de modèle numérique hydrogéologique, conceptualisation et révision du manuscrit.

Daniel Paradis : Supervision et révision du manuscrit.

Dany Lauzon : Apport méthodologique en géostatistique et révision du manuscrit.

### **Publication ciblée**

Journal : Mathematical Geosciences - Accepté

Date de soumission : janvier 2024

### **Objectif de l'article**

L'objectif de cet article est de développer un RNG permettant de prédire les niveaux spatio-temporels des eaux souterraines dans un aquifère granulaire et de roc fracturé. L'objectif principal de ce cadre est de servir de substitut aux modèles hydrogéologiques numériques utilisés pour la gestion des eaux souterraines. Le modèle RNG intégré dans ce cadre peut être utilisé pour la prévision des niveaux d'eau souterraine après avoir été entraîné sur des données synthétiques par des modèles hydrogéologiques numériques. Ce faisant, nous pouvons atténuer certains des défis liés à l'utilisation des modèles physiques dans la gestion des eaux souterraines. Ce cadre est en accès libre, permettant à quiconque de l'utiliser ou de l'adapter librement à ses propres besoins.

---

## Résumé

La prévision spatio-temporelle des eaux souterraines joue un rôle essentiel dans la gestion et la planification des ressources en eau. Face aux pressions anthropiques et climatiques croissantes sur les eaux souterraines, il est urgent de développer des outils de prévision plus accessibles et efficaces afin d'assurer une gestion durable et une distribution équitable de cette ressource vitale. Les méthodes traditionnelles, qu'il s'agisse de relations empiriques simples ou de modèles numériques complexes, ont obtenu des résultats variables. Cependant, ces approches rencontrent souvent des difficultés à intégrer de nouvelles informations et/ou sont coûteuses en calcul, ce qui peut compromettre la précision et la fiabilité des prévisions à long terme. Pour pallier à ces limitations, nous proposons un cadre basé sur un réseau de neurones par graphes (RNG) qui intègre à la fois des méthodes géostatistiques explicites et des techniques d'apprentissage automatique implicites comme substitut aux modèles numériques afin de fournir des prévisions rapides et précises des niveaux d'eau souterraine. Ce cadre RNG peut également être utilisé comme un outil de couplage avec un modèle numérique pour faciliter la prévision, et ne se limite donc pas uniquement à une approche de substitution. Les RNG excellent dans le traitement de structures de données complexes, ce qui les rend particulièrement adaptés aux applications en géosciences. Notre cadre est entraîné sur des niveaux transitoires simulés d'eau souterraine et des débits de pompage générés à partir d'un modèle hydrogéologique complexe au Québec, Canada. Il a été testé sous divers scénarios de pompage, dont chacun nécessite plusieurs jours de calcul avec un simulateur d'écoulement dédié. En revanche, notre cadre fournit des réponses en quelques secondes, offrant ainsi un soutien précieux pour la gestion des eaux souterraines dans des conditions de stress hydrologique complexes. Les résultats démontrent sa capacité à produire des prévisions rapides et fiables des niveaux d'eau souterraine, constituant ainsi une alternative computationnellement efficace pour la gestion des eaux souterraines dans des contextes hydrogéologiques complexes.

## Abstract

Spatio-temporal groundwater (GW) forecasting plays an important role in water resources management and planning. Given the increasing anthropogenic and climatic pressures on GW, there is an urgent need for more efficient and accessible forecasting tools to ensure the sustainable management and distribution of this vital resource. Traditional methods, including simple empirical relationships and complex numerical simulations, have shown varying success. However, these approaches often struggle to incorporate new information and are computationally demanding, which can compromise the accuracy and reliability of forecasts over time. To address these limitations, we propose a graph neural network (GNN)-based framework that integrates explicit geostatistical methods with implicit machine learning (ML) techniques, serving as a surrogate for numerical models to provide fast and accurate GW level forecasts. This GNN framework can be used as a coupling tool to the numerical model to aid in forecasting; it is not limited to surrogacy. GNNs excel in handling complex data structures, making them well-suited for geosciences applications. Our framework is trained on simulated transient GW levels and pumping rates generated from a

---

complex hydrogeological model in Quebec, Canada. It has been tested under various pumping scenarios, each taking several days to solve using a dedicated flow simulator. In contrast, our approach delivers results in seconds, offering valuable support for GW management under complex stress conditions. The results highlight its ability to provide rapid and reliable GW forecasts, offering a computationally efficient alternative for managing GW in intricate hydrogeological settings.

**Keywords :** Machine Learning, Deep Neural Network, Graph Convolutional Network, Long Short-Term Memory, Groundwater Forecasting, Surrogate Models

## 2.1 Introduction

Groundwater (GW) is not only a vital resource that is essential to human health and the environment, but it also plays a key role in economic development. Given the growing impact of climate change and anthropogenic stresses on water resources, it is essential to have accurate tools for forecasting and managing this natural resource.

Physically based hydrogeological models are frequently used to evaluate and forecast GW levels and contaminant transport (Anderson et al., 2015; Delottier et al., 2022; Paradis et al., 2017). These complex hydrogeological models can be analyzed to determine how an aquifer system responds to various hydrogeological changes (whether climatic or anthropogenic), thereby aiding in aquifer management (Paradis et al., 2016). Due to the complexity of certain physical models such as the numerical model used in this study, running simulations will be computationally intensive, which hinders rapid decision-making and complicates GW management within the territory. The construction and operation of numerical models are inherently complex, limiting their widespread adoption and use. This is particularly problematic in areas like Québec, Canada, where GW is now integrated into land management. The rigidity and computational demands of traditional hydrogeological models often prevent rapid predictions, making it difficult to support management decisions that require a quick response.

Geostatistics is commonly used for spatio-temporal GW levels forecasting in wells, but traditional algorithms assume stationarity and require a high degree of correlation between variables, which does not typically reflect GW flow under different stresses (Pyrzcz et al., 2014). Additionally, geostatistical tools excel in interpolation but struggle with extrapolation, often leading to unrealistic predictions. To alleviate these issues, ensemble Kalman filters and ensemble Kalman smoothers (Evensen, 2003, 2009) are proposed to integrate realistic physical modeling while allowing ensemble members to be updated based on empirical covariance. While these tools are very powerful, the computation time is prohibitive and requires a highly trained person to run the models. Also, intricate geological conditions could further complicate the use of geostatistical methods, making explicit geostatistics challenging or even ineffective for forecasting active GW levels. To overcome these challenges, a promising approach could involve combining explicit geostatistics with implicit machine learning (ML) techniques, leveraging the strengths of both methodologies.

---

In recent years, hydrogeology researchers have increasingly employed various ML or deep neural network (DNN) techniques, such as recurrent neural networks (RNNs), to forecast GW levels (Li et al., 2021; Wunsch et al., 2021; Zhang et al., 2018). One of the RNNs used by researchers is the Long Short-Term Memory (LSTM) (Hochreiter et al., 1997). The LSTM recursive network is designed to work with long time series data for addressing the gradient vanishing problem. It works well with time series predictions in hydrology/hydrogeology, but it cannot assess the impact of spatial attributes such as subsurface hydraulic conductivity on GW levels. To address these limitations, some have explored alternative DNNs for GW levels forecasting. Notably, Yan et al. (2022), Wang et al. (2020a), and Tartakovsky et al. (2020) have leveraged physics-informed DNNs to constrain GW flow equations to obtain improved forecasting. The physics-informed DNNs are notably useful in GW levels forecasting, but they also do not address the spatial attributes of subsurface hydraulic conductivities and how they affect GW flow and levels.

Due to the limitations of the DNNs mentioned above, we believe that a spatio-temporal GNN model, which can incorporate spatial information along with time series data can be a solution. The GNN model was first introduced by Scarselli et al. (2008), which uses pre-existing neural network methods for processing information or data on a graph structure (i.e., propagating or learning information through the connective structure or edges of a graph). Unlike convolutional neural network (CNN) architectures (LeCun et al., 1998), GNNs do not require gridded data, such as perfectly aligned pixels in a picture, giving them the flexibility to process any type of data that can be represented by a graph. This flexibility has enabled various spatio-temporal studies in a wide range of domains. Some of these studies include traffic predictions (Bui et al., 2022; Diehl et al., 2019; Zhao et al., 2019), online recommendation systems (Ying et al., 2018), protein folding (Caflich, 2006; Strokach et al., 2020), and three-dimensional geological structural modeling (Hillier et al., 2021). More specific cases of researchers using GNN models in the field of hydrogeology and hydrology are Bai et al. (2023) and Sun et al. (2021); the former used GNNs to forecast GW levels in wells with field-measured data from British Columbia, Canada, while the latter used GNNs trained with measured stream data to enhance stream level forecasting in U.S. catchments.

However, training an ML model requires a large amount of data which is lacking in common hydrological projects. Consequently, the predictive power of such a trained model is limited. Recent research has shown that the use of synthetic data during training can generalize the predictive power of ML models. In hydrogeology, geostatistical simulations have proven to be very efficient in providing realistic geological settings with respect to GW flow (Lauzon et al., 2023; Neven et al., 2022). This approach is intended to forecast the impact of water usage on GW availability while circumventing the complexities associated with the simulation process of numerical models. Once an ML model is adequately trained, it can provide timely simulation results and easily adapt to various scenarios, effectively serving as a surrogate for the physical models. Using surrogate models in hydrogeology is not novel, as different types of surrogates have been employed for physical models (An et al., 2018; Previati et al., 2024; Yongkai et al., 2022)

---

In this paper, we present an implicit ML framework containing a graph neural network (GNN) model with 2 graph convolution network (GCN) (Kipf et al., 2016) layers, followed by 2 long short-term memory (LSTM) (Hochreiter et al., 1997) layers. For the rest of this paper, we will refer to this GNN model as GCN-LSTM. The GCN-LSTM network is inspired by the temporal GCN (T-GCN) model developed by Zhao et al. (2019). We have adapted the T-GCN network, originally developed for spatio-temporal traffic forecasting, for GW levels forecasting. Our ML framework incorporates the GCN-LSTM model because it can capture the spatial connectivity among wells and enables the temporal dynamic forecasting of GW levels under different anthropogenic pumping conditions. The adjacency matrix within the network provides spatial connectivity information among the nodes in a graph, representing the monitoring and pumping wells in our study. Each node is associated with multivariate time series data, which the network leverages for GW levels forecasting. The GW levels forecasting from all wells are made simultaneously, giving us a complete spatial distribution of GW levels under different pumping conditions over time. The main contribution of this project is a GNN framework that forecasts the spatio-temporal dynamics of GW levels in wells under anthropogenic pressures of aquifer pumping.

The paper is organized as follows. The basic background and theory of GNNs and the mathematical formulation for the GCN-LSTM network used in our framework (Fig 2.1) are explored in Section 2. In Section 3, we discuss the hydrogeological numerical model used for this project. The methodology for our framework is described in 5 main steps in Section 4, along with the experimental setup for training the GCN-LSTM network. The results of this study are presented in Section 5, and the discussion and concluding remarks are presented in Section 6.

## 2.2 Graph Neural Network

### 2.2.1 Graph Notations

A graph can be represented as  $G(V, E)$ , where  $V$  is a set of  $N$  nodes, and  $E$  is a set of edges. An edge,  $e_{ij} = (v_i, v_j) \in E$ , can be associated with a weight,  $w_{ij}$ , while in an unweighted adjacency matrix,  $w_{ij} = 1$ . Furthermore, these edges can be directed or undirected. Undirected edges are bidirectional, a pair of nodes  $u \in V$  and  $v \in V$  where  $(u, v) \in E \iff (v, u) \in E$ . In a one-hop neighborhood, all nodes connected directly to node  $v$  will form a neighborhood of  $N(v) = \{u \in V \mid (v, u) \in E\}$ .

An adjacency matrix,  $A$ , can be interpreted as a simplified way to view a graph, where  $A \in \mathbb{R}^{V \times V}$ . Each node indexes a specific row and column where entries in this matrix are edge connections between the nodes. In an unweighted edge situation, if  $e_{ij} \in E$ , then  $A_{ij} = 1$ , else  $A_{ij} = 0$ . In a weighted edge situation, if  $e_{ij} \in E$ , then  $A_{ij} = w_{ij}$ , else  $A_{ij} = 0$ .

---

## 2.2.2 Graph Convolution

There are 2 classes of graph convolution, spatial and spectral based. The spatial-based convolution works directly over a node’s local neighborhood. The spatial information can be aggregated from local neighborhoods and is used to update the node’s state (Bruna et al., 2013). The spectral-based convolution transforms nodes in the spectral domain, which initially solves for the Eigen decomposition of the graph’s adjacency matrix. Later, Defferrard et al. (2016) extended this idea with a fast localized convolution by using a Chebyshev expansion for eigenvalue approximation, called ChebNet. The GCN used in our GCN-LSTM network utilizes a spectral convolution that is further simplified from ChebNet to approximate the first-order neighbors (i.e., the one-hop neighborhood or the neighboring node that is directly connected) (Kipf et al., 2016). By simplifying the Chebyshev polynomial to a first-degree approximation, the GCN is more computationally efficient compared to the ChebNet; therefore, it can lead to faster convergence in the training process.

## 2.2.3 GCN-LSTM Theory

The GCN-LSTM network used in our framework is inspired by Zhao et al. (2019). This model consists of 2 GCN layers followed by 2 LSTM layers, followed by a dropout layer and a dense layer before the output prediction.

For the GCN layers, Kipf et al. (2016) designed the layers to aggregate the first-degree neighborhood information of the node to update its state. GCN considers the general equation of  $H$  layer-wise propagation as follows :

$$H^{(l+1)} = \tanh \left( \tilde{D}^{-1/2} \tilde{A} \tilde{D}^{-1/2} H^l W^l \right) \quad (1)$$

where  $l$  is the current layer,  $\tanh(\cdot)$  is the hyperbolic tangent activation function,  $\tilde{A} = A + I$ , where  $A$  is the adjacency matrix and  $I$  is the identity matrix.  $D$  is the degree matrix, where  $\tilde{D} = D + I$ , and  $W$  are the weights.

With 2 GCN layers used in this model, the forward propagation can be simplified to :

$$f(X, A) = \tanh \left( \hat{A} \tanh \left( \hat{A} X W^0 \right) W^1 \right) \quad (2)$$

where  $X$  is the input data, and  $\hat{A} = \tilde{D}^{-1/2} \tilde{A} \tilde{D}^{-1/2}$ .

Seo et al. (2018) formulated a graph convolution into an LSTM that operates on a non-gridded graph structure. This LSTM formulation is used in our GCN-LSTM network. Graph convolution-based LSTM formulation by Seo et al. (2018) can be written as :

$$C_t = f_t \circ C_{t-1} + i_t \circ \tanh (W_c x *_G X_t + W_c h *_G h_{t-1} + b_c)$$

$$\begin{aligned}
i_t &= \sigma(W_{ix} *_G X_t + W_{ih} *_G h_{t-1} + W_{ic} \circ C_{t-1} + b_i) \\
f_t &= \sigma(W_{fx} *_G X_t + W_{fh} *_G h_{t-1} + W_{fc} \circ C_{t-1} + b_f) \\
o_t &= \sigma(W_{ox} *_G X_t + W_{oh} *_G h_{t-1} + W_{oc} \circ C_t + b_o) \\
h_t &= o_t \circ \tanh(C_t)
\end{aligned} \tag{3}$$

where  $C$  is the cell state and  $h$  is the hidden state. The input, forget, and output gates are  $i$ ,  $f$ , and  $o$ , respectively. The subscript  $t$  is the current recurring iteration of the cell and  $t - 1$  is the previous iteration.  $W$  is the weight matrix for the respective gates and states. The  $\circ$  and  $*_G$  are the Hadamard product and the graph convolution, respectively. The  $\sigma(\cdot)$  is the sigmoid activation function and the  $\tanh(\cdot)$  is the hyperbolic tangent activation function.  $X$  is the input data. The  $b$  is the bias for the respective gates.

In the GNN context of this paper, wells represent nodes, and edges represent the spatial connectivity between the wells. The edges of the graph are undirected and unweighted. This is a graph-focused task that involves supervised training. Forecasting the GW levels in multiple wells through time is a spatio-temporal problem that can be represented as the learning of the mapping function  $f$  on the spatially connected graph of the adjacency matrix,  $A$ , and the input time-series data  $X$ . Hence, the mapping of  $f(X, A)$  can be applied to the forecasting of GW levels in wells.

## 2.3 The Numerical Model

The GNN framework is tested on the regional hydrogeological numerical model of Ville Mercier (Fig 2.2), located in southern Quebec, Canada. The Ville Mercier site has been investigated for several decades to control and monitor the operation of a pump and treat plant. A complete set of geological, hydrogeological, and geophysical data is available at the site to build and calibrate the numerical model and to monitor the hydraulic head variations (Claprood et al., 2023).

The 3D numerical model of Ville Mercier represents 6 major hydrostratigraphic units that control the GW flow in the region. From top to bottom, these units are recent organic sediments and filling materials, fine-grained marine sediments from the marine clay unit, coarse glaciofluvial sediments from the esker sand and gravel unit, regional till unit, fractured bedrock, and intact bedrock (Fig 2.3). The complexity of the model lies in the geological heterogeneity, as the local esker aquifer is present only in specific areas, with intricate lateral extents outlined by more than 30  $km$  of seismic reflection data (Claprood et al., 2022). Additionally, there is a known hydraulic connection between the esker aquifer and the regional fractured bedrock aquifer, but the volume and precise location of these connecting windows remain unknown. The numerical model covers an area of 226  $km^2$ , has a mean thickness of 57  $m$ , and contains 3,974,915 elements on 35 layers, with element size varying from 5  $m$  locally to 150  $m$  regionally. It is built with the FEFLOW software and has been calibrated

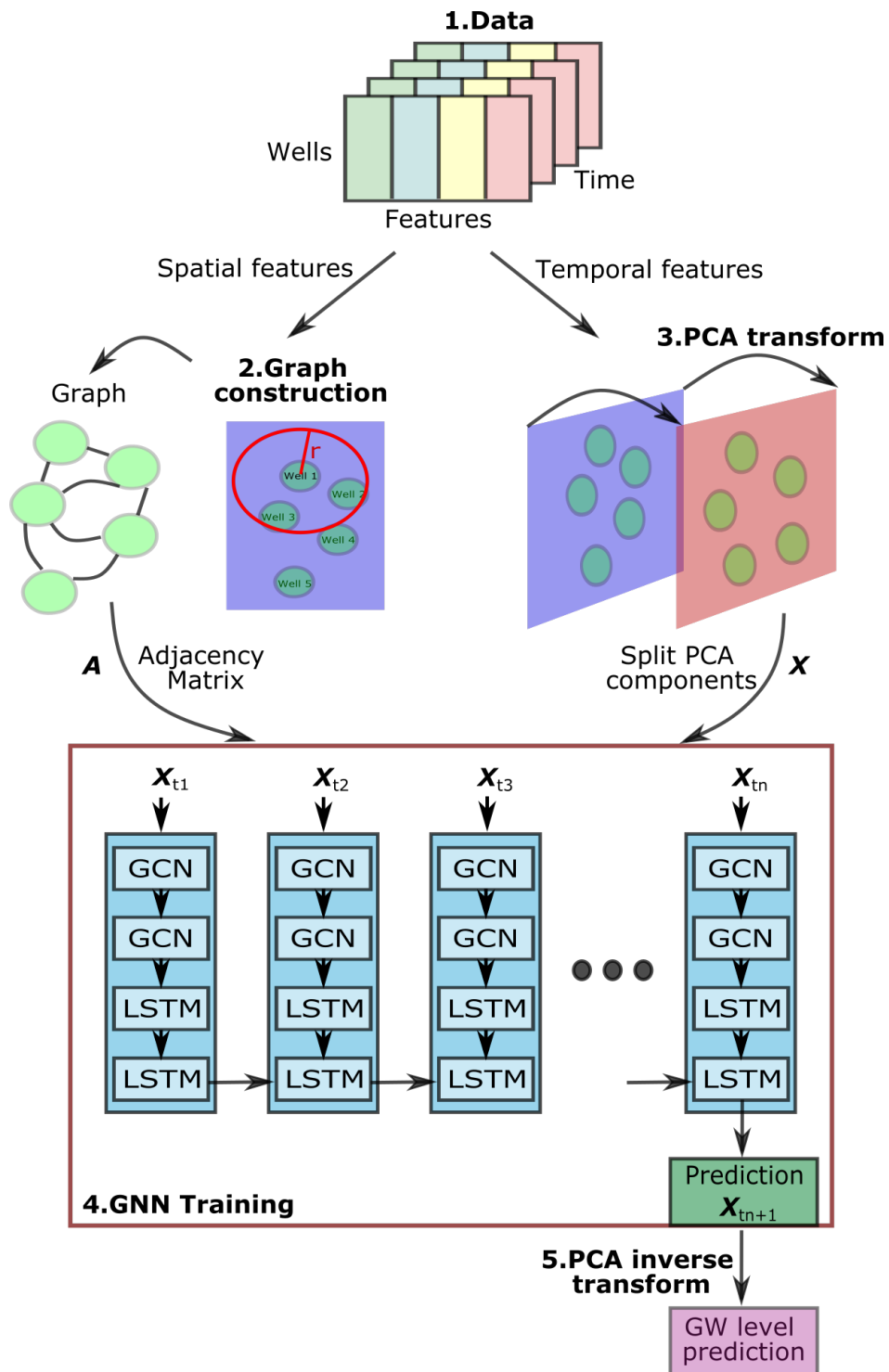
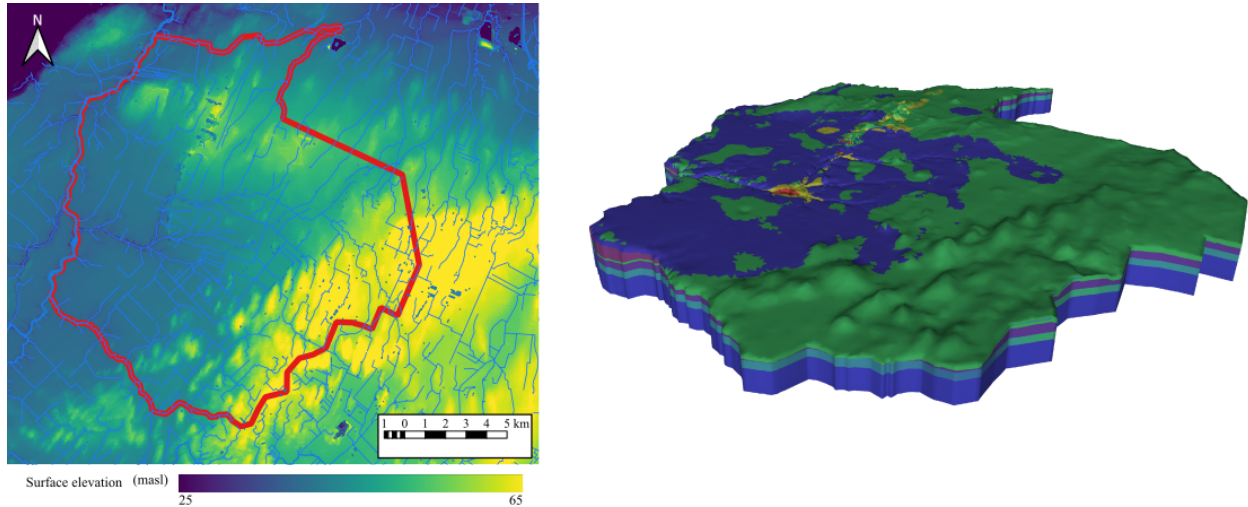
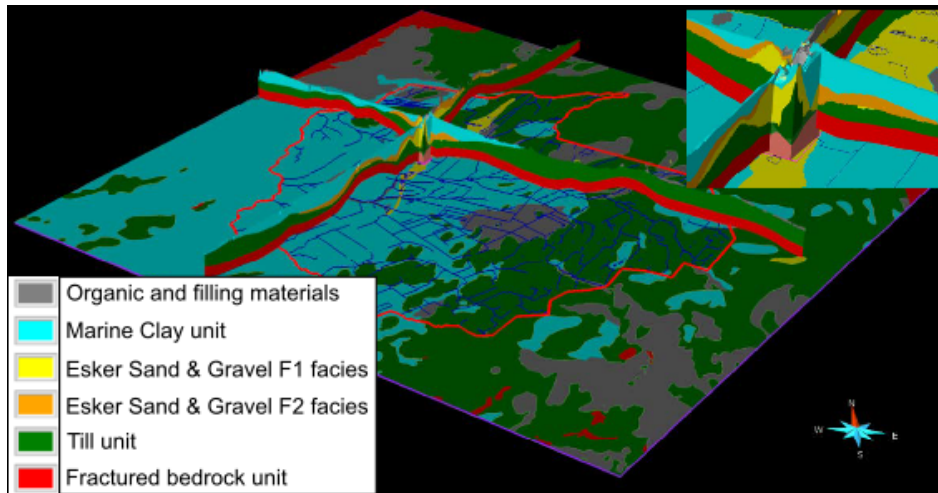


FIGURE 2.1 : The framework consists of 5 main steps. Step 1 : data resampling, normalization, and compilation. Step 2 : Graph construction using the spatial features. Step 3 : PCA transform and components splitting using the temporal features of the data. Step 4 : GNN model training by taking the adjacency matrix of the graph from step 2 and the split PCA component of the transient GW level from step 3. Step 5 : GW level predictions by inverse transform of GNN PCA components.



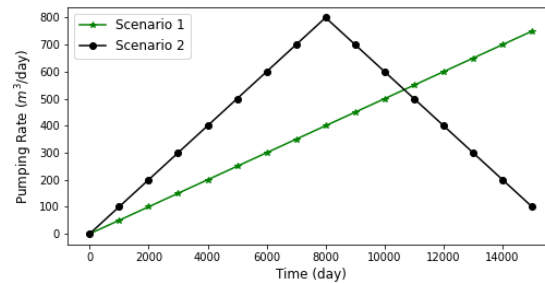
**FIGURE 2.2** : Left : Location of the numerical model on the south shore of the St. Lawrence River in Quebec (Canada). Thick red line is the outline of the numerical model. Right : 3D numerical model (vertical exaggeration of 15 to 1).



**FIGURE 2.3** : 3D geomodel of the hydrostratigraphic units at Ville Mercier, Quebec, Canada. Regional view has a vertical exaggeration of 40, and the local view has an exaggeration of 20. This figure is modified from (Claprood et al., 2023).

in steady-state and transient regimes using ensemble smoother methods (Claprood et al., 2023; Béraud et al., 2024).

The hydraulic properties of the esker aquifer are characterized by 123 in situ hydraulic conductivity measurements within the boundaries of the numerical model, while the hydraulic properties of the bedrock aquifer consist of 111 measurements from 15 wells. To represent the spatial heterogeneity and uncertainty of the hydraulic conductivity fields in the esker and in the fractured bedrock aquifers in the model, 200 realizations of hydraulic conductivity were computed using turning bands simulations. An important density of hydraulic head observations exists locally at the site (within a few hundred meters of the pump and treat plant), but very few observations are available at the regional scale. The measured hydraulic heads show great spatial variability, and their time series



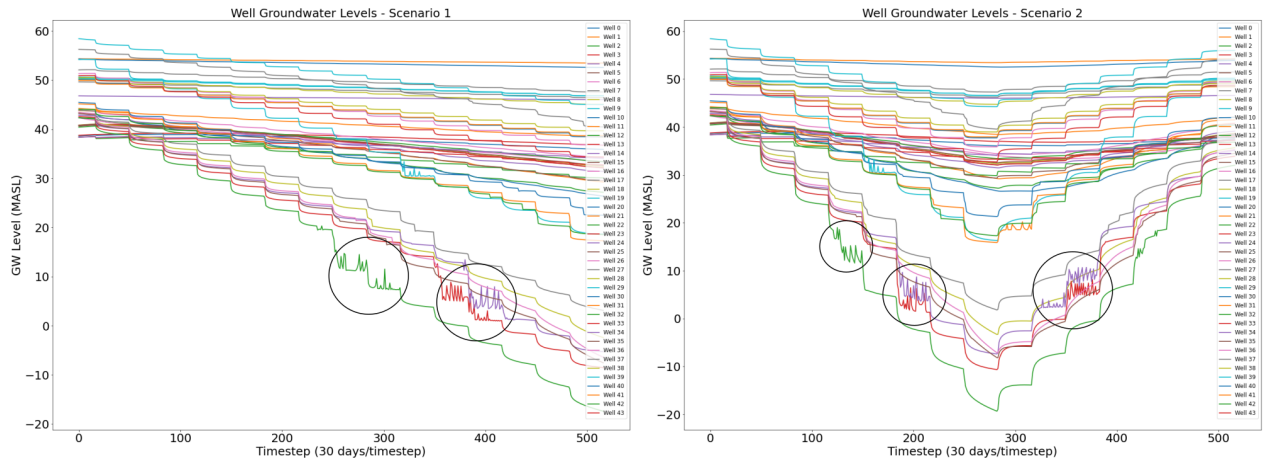
**FIGURE 2.4 : Pumping scenarios applied to the numerical hydrogeological model to carryout simulations.**

show different levels of sensitivity to pumping rate variations from the pump and treat plant. The spatial distribution of the observations and their complex temporal behavior complicate the calibration of the numerical model and introduce significant uncertainty when using it to predict future water level variations.

## 2.4 Methodology

### 2.4.1 Framework Construction

The ML framework presented here is designed as a surrogate of the numerical hydrogeological model to forecast GW levels in monitoring and pumping wells under varying pumping stresses. The two main components of this framework consist of a GCN-LSTM network for GW level forecasting and a principal component analysis (PCA) method for data dimension reduction. The framework consists of 5 main steps (Fig 2.1). Step 1 involves numerical model simulations to extract synthetic data for GCN-LSTM network training and adjacency matrix construction. We conducted transient simulations using two different pumping scenarios (Fig 2.4), resulting in dataset 1 and dataset 2. In step 2, an adjacency matrix is constructed to represent the spatial connectivity of the well. Well connectivity is established by applying a kernel radius cut-off distance to the calculated Euclidean pairwise distance. The best radius distance is obtained by testing in the GCN-LSTM network. Step 3 is for time series data dimension reduction; in this step, we use time series features of transient GW levels and pumping rates for PCA embedding to transform the two time series from Euclidean space to PCA space. Step 4 is the GCN-LSTM network optimization and training. The transient GW level component of the PCA embedding from step 3 is used for network training. Step 5 performs PCA inverse transformation to convert the GCN-LSTM network predictions from PCA space back to the original space.



**FIGURE 2.5 : Well GW levels from transient numerical model simulations. The black circles in both plots highlight some of the instabilities that emerged during numerical model simulations.**

### Step 1 : Numerical Model Simulation

To test the robustness of the GCN-LSTM network, two distinct water pumping scenarios are applied to the pumping wells for simulations (Fig 2.4). In scenario 1, the pumping rate increases steadily over 16,000 *days* from  $0 \text{ m}^3/\text{day}$  to  $800 \text{ m}^3/\text{day}$ . In scenario 2, the pumping rate increases gradually from  $0 \text{ m}^3/\text{day}$  to  $800 \text{ m}^3/\text{day}$  over 8000 *days*, and then gradually decreases to  $100 \text{ m}^3/\text{day}$  until the end of the simulation at 16,000 *days*.

A total of 44 wells are selected in the numerical model, comprising 22 pumping wells that were created originally for hydrogeological assessment in the numerical model and an additional 22 monitoring wells that are randomly selected in the numerical model. The pumping scenarios are applied only to the pumping wells, while the monitoring wells are used only to observe changes in GW levels over time. The monitoring wells are labeled 0 to 21, and the pumping wells are labeled 22 to 43 in our datasets. The transient GW data extracted from the numerical simulations of all the wells are plotted in Fig 2.5.

All the features extracted from the numerical model are the location of the wells in  $X$ ,  $Y$ , and  $Z$  coordinates, the hydraulic conductivities,  $K_x$ ,  $K_y$ , and  $K_z$  at the well opening, the steady-state head level at the well locations, the transient GW level at the well locations, and the pumping rates. These specific parameters are chosen on the basis that they are parameters that can be obtained in the field, and thereby securing that the GNN framework developed here could be used with measured field data (if a study area contains a large enough dataset to train the GCN-LSTM network). All the parameters extracted from numerical model simulations are scaled from 0 to 1 for training the GCN-LSTM network.

The parameters of  $X$ ,  $Y$ ,  $Z$ ,  $K_x$ ,  $K_y$ ,  $K_z$ , and the steady-state heads are used to construct the adjacency matrix (Table 2.1). These parameters are selected because they are static parameters, enabling the interpretation of spatial relationships between wells. The  $X$ ,  $Y$ ,  $Z$  coordinates provide

---

**TABLEAU 2.1 : Features used for graph construction and model training**

	<b>Features</b>
Graph	$X, Y, Z, K_x, K_y, K_z$ , steady-state head
GCN-LSTM training	Transient GW levels, Pumping rates

the location of the wells, while the hydraulic conductivities characterize the aquifer layer in which the well openings are located. Steady-state heads indicate the general flow direction and spatial distribution of the GW, allowing us to map the wells' positions in terms of the GW head gradient distribution. Dynamic parameters, including transient GW levels and pumping rates, are used for PCA embedding and subsequently GCN-LSTM network training to forecast GW levels (Table 2.1). These two time series are chosen to train the GCN-LSTM network to forecast GW levels under varying pumping stresses.

For the rest of the paper, we will use the word features to represent the data parameters extracted from numerical model simulations to avoid confusion between the parameters (features) and hyperparameters in the GCN-LSTM network.

## **Step 2 : Graph Construction**

Here, the graph nodes represent wells and the graph edges represent spatial connectivity between wells. The wells' features of  $X$ ,  $Y$ , and  $Z$  position, hydraulic conductivity of  $K_x$ ,  $K_y$ , and  $K_z$ , and steady-state head are used to construct the adjacency matrix (an adjacency matrix can be viewed as the numerical representation of a graph). As seen in Fig 2.3, the hydrogeological numerical model is complex with a local esker aquifer system and a regional fractured bedrock aquifer. The two aquifer systems are mostly separated by an impermeable aquitard clay layer. The proximity of two wells does not imply that these wells are hydraulically coupled, as one well could be in the local esker aquifer and the other located in the regional aquifer separated by an aquitard. Therefore, it is important to use these 7 features to be certain that the wells that are spatially connected are in close proximity but also in the same aquifer layer.

To establish the spatial connectivity, a pairwise Euclidean distance calculation is done for each well relative to all other wells. A radius filter is applied to the calculated distances of each well to trim the wells that are far away. If the calculated distances fall within or on this radius value, we declare these wells to be spatially connected; otherwise, they are not connected. In terms of a graph, if two wells are spatially connected, then there exists an edge between these well nodes (i.e., if an edge exists, it is represented by a 1 in the adjacency matrix, and no connection is represented by a 0). Furthermore, the nodes are made to be self-looped; this can be viewed as  $\tilde{A} = A + I$ , where  $\tilde{A}$  is the self-looped adjacency matrix, which is the result of adding the identity matrix  $I$  to the original adjacency matrix  $A$ . If a well does not have a neighbor within the radius cut-off distance, the closest neighbor is selected.

---

To determine the optimal graph radius for hydraulic connectivity in the wells, various radii (0.10, 0.15, 0.20, 0.25, 0.30, 0.35, 0.40, 0.50) are tested to ensure that the adjacency matrix is constructed correctly, in that wells are spatially connected only if they are close in location and are in the same aquifer layer. A homophilic projection of the graphs constructed with cut-off distance values of 0.15, 0.20, 0.25, and 0.30 are shown in Fig 2.6. The radius distance is scaled from 0 to 1. With a distance of 0, the well only considers its own position; so the only spatial connection is to itself, and with a distance of 1, the well is connected to all other wells.

The construction of the adjacency matrix can be represented mathematically as follows.

$$A^{(i,j)} = \begin{cases} 1, & \text{if } \text{dist}(V_i, V_j) \leq \text{radius}, \\ 0, & \text{otherwise,} \end{cases} \quad (4)$$

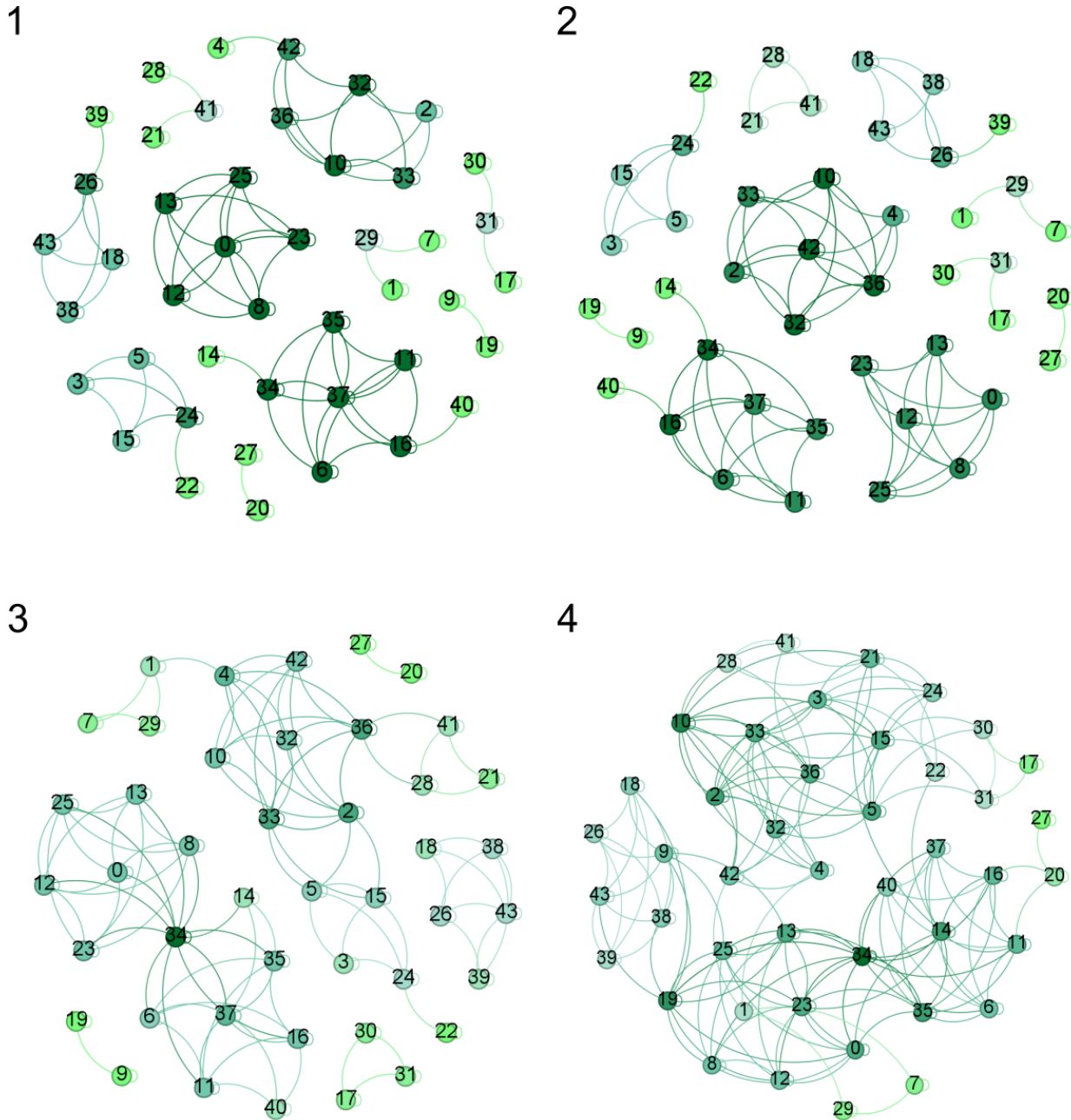
where  $i$  and  $j$  are the row and column indices of the adjacency matrix  $A$ , and  $\text{dist}(V_i, V_j)$  denotes the Euclidean distance between the node vectors  $V_i$  and  $V_j$ . Each node vector contains the 7 features of  $K_x, K_y, K_z$ , steady-state head, and the well opening positions  $X, Y$ , and  $Z$ .

### Step 3 : PCA Dimension Reduction

The PCA method is used for dimension reduction. We have 2 time series : the transient GW level in all wells and the pumping rates from the pumping wells. The GCN-LSTM is a univariate network that takes 3-dimensional data, but we have 4 dimensions (the dimensions are : mini-batch size, number of nodes, number of time steps, number of time series) in our datasets. Therefore, dimension reduction is needed.

We used PCA to embed the 2 time series into 2 principal components to take advantage of the orthogonalization property of the PCA method. PCA is performed per time step along the time series data. As the pumping rates vary over time, their relationship with GW level, as well as the extent of the sphere of influence, will also change. To accurately capture the relationship between these variables at each point in time, we applied PCA for each time step. The connection between the pumping rate and GW levels in wells, particularly with hydraulic conductivity varying across five orders of magnitude, remains poorly understood. By conducting PCA at each time step, we ensured that the relationships among the variables were orthonormal at that specific moment, enabling us to map these relationships in a clearer and more comprehensible manner.

Since the 2 components are orthonormal in the PCA space, we take the PCA-embedded data matrix and split the components into 2 vectors instead of reducing the 2 components into 1. The reduction of the pumping rate component or the transient GW component is not desirable. We want to forecast the transient GW level and want to use this component for GCN-LSTM network training. The calculated variance in the pumping rate component can be high at certain time steps; therefore, the reduction of this component will lead to the loss of information from the pumping



**FIGURE 2.6 : Graph homophily projection of well nodes numbered from 0 to 43. The graphs are constructed with a radius cut-off of 0.15, 0.20, 0.25, and 0.30 for plots 1, 2, 3, and 4, respectively. The radius cut-off is shown to influence the degree of complexity of the relationships between the wells (Bastian et al., 2009a).**

rate variable during network training (i.e., we want the anthropogenic pumping stress in aquifers to be learned in the GCN-LSTM training process, and therefore be able to forecast GW levels more accurately). However, we did test the PCA dimension reduction method to reduce the two components into one. The pumping rate component is reduced, and the transient GW component is used for training the GCN-LSTM network. The results were poor, and the dimension reduction is undesirable in this case. As mentioned above, the variance in the pumping rate in specific time

---

steps can be large; when this component is reduced, we lose important information. We opted to split the components instead of component reduction. Using this splitting method, one could include any number of temporal features for forecasting; it is not limited to the 2 features shown here. This allows multiple time series to be transformed into PCA space, and the principal component of interest can be used for model training.

#### **Step 4 : GCN-LSTM Model**

The GCN-LSTM model takes the compiled timeseries data and the adjacency matrix for training. In a general sense, the GCN takes the adjacency matrix, which gives the spatial connectivity of the nodes, and does a first-order approximation graph convolution on the nodes. The LSTM takes the transient GW level component from the PCA embedding and learns the temporal dynamics for GW level forecasting.

#### **Step 5 : PCA Inverse Transform**

In this step, the forecasts from the GCN-LSTM network are transformed back into the original space. The PCA forecast is paired with the corresponding component of the pumping rate vector in step 3. After combining the 2 components, an inverse PCA transformation is performed to return to the original space.

### **2.4.2 GCN-LSTM Model Training and Optimization**

The split PCA transient GW component vector from step 3 of our framework is compiled into  $X$  as the input data with a shape of  $N$  rows (representing the number of wells) and  $T$  columns (representing the number of time steps), given  $X \in \mathbb{R}^{N \times T}$ . Here, we have chosen each time step to be per 30 days; therefore  $T = [t_0, t_1, t_2, \dots, t_{k-1}]$ , where  $t_0 = 30 \text{ days}$ ,  $t_1 = 60 \text{ days}$ ,  $\dots$ , and  $k$  is the number of time steps sampled at 30 days per time step. The data  $X$  is divided into sections for the GCN-LSTM network training (70%), validation (15%), and testing (15%). The optimizer used is the adaptive moment estimation (Adam) (Kingma et al., 2014) and employs the mean square error (MSE) as a loss function. To avoid overfitting, we implement early stopping during model training and apply a 50% data dropout rate. The GCN-LSTM hyperparameters (Table 2.2) are optimized using a coarse-to-fine grid search.

For model training, the input training data  $X$  is resampled to 30-day time steps. We train the GNN model with four 30-day time steps ( $h = 4$ ) and forecast a single 30-day time step ( $n = 1$ ). We use the sliding window technique for the sequence-to-sequence forecasting. The training sequence

TABLEAU 2.2 : Optimized Hyperparameters for GCN-LSTM Model

Parameter	Value
Activation Functions	Tanh
Learning Rate	0.001
Nodes for GCN Layers	[5, 5]
Nodes for LSTM Layers	[600, 600]
Dropout Rate	0.5
Loss Function	MSE
Optimizer	Adam

is given by  $X = \left[ [t_0, t_{0+h}), [t_1, t_{1+h}), [t_2, t_{2+h}), \dots, [t_{(k-1)-(h+n)}, t_{(k-1)-n}] \right]$ , and the forecast sequence is  $Y = \left[ t_{0+(h+n)}, t_{1+(h+n)}, t_{2+(h+n)}, \dots, t_{(k-1)-(h+n)} \right]$ .

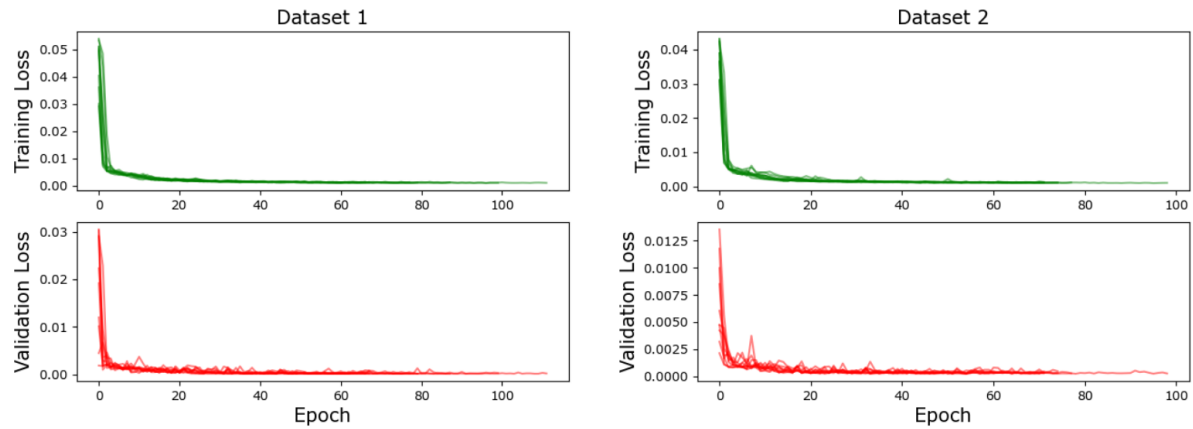
We test the adjacency matrices constructed using the range of radii (0.10, 0.15, 0.20, 0.25, 0.30, 0.35, 0.40, 0.50) as input to test the GCN-LSTM training to identify the most suitable radius for our datasets. This is an ad hoc exercise that is data dependent. The suitability of the radii is assessed through the MSE loss function value calculated during model optimization. The root mean squared error (RMSE) and mean absolute error (MAE) are used as metrics.

We trained the GCN-LSTM network 10 times for each dataset to determine the sensitivity range of the weights optimized by the Adam optimizer in the network. The optimizer is stochastic by nature; therefore, each training of the network optimization could converge to a different value, resulting in a range of possible predictions for GW forecasting. We want to show this range of values, which is a more honest representation of the forecasting power of the network. By training the model multiple times, we can also assess more accurately the robustness of the network (i.e., does the network converge during each training and over how many epochs). The metrics MAE, RMSE, and the coefficient of determination ( $R^2$ ) are used to evaluate the overall performance of the model prediction.

## 2.5 Results

The optimization of hyperparameters is dataset-dependent. Since there are 2 different datasets from 2 simulation scenarios, a range of hyperparameter values is found for both datasets. From this range, the hyperparameters are fine-tuned to fit both datasets. The optimized hyperparameters for the 2 datasets are shown in Table 2.2.

The adjacency matrices derived from our tested range of radii are used to train our GCN-LSTM network. The metrics from the network training and validation helped to determine the optimal radius for the 2 datasets. For dataset 1, the optimal range is 0.20 to 0.30 (Table 2.3), and for dataset 2, this range is 0.10 to 0.30 (Table 2.4). Based on our results, a radius of 0.25 is chosen, which falls within the optimal ranges of both datasets. Herein, we use the adjacency matrix constructed using the 0.25 radius cut-off as the adjacency matrix for model training.



**FIGURE 2.7 : Training and validation loss of dataset 1 and 2. The 10-model training and validation loss are plotted for each dataset.**

Using the hyperparameters in Table 2.2, the adjacency matrix constructed using the radius cut-off value of 0.25, and the PCA embedded component of the transient GW level, the GCN-LSTM network is trained. The 10-model optimizations for the two datasets are plotted over the number of epochs in Fig 2.7. The loss values decrease rapidly and reach a plateau, indicating the model has converged. Given a computer with one CPU, each model training took a few minutes to complete.

**TABLEAU 2.3 : Testing radius cut-off values for graph construction on dataset 1.**

<b>Radii Range</b>	<b>0.1</b>	<b>0.15</b>	<b>0.2</b>	<b>0.25</b>	<b>0.3</b>	<b>0.35</b>	<b>0.4</b>	<b>0.45</b>	<b>0.5</b>
Train Loss MSE	9.91E-4	1.34E-3	9.98E-4	1.00E-3	8.68E-4	1.06E-3	8.75E-4	3.16E-3	1.80E-2
Validation Loss MSE	1.33E-4	2.34E-4	1.53E-4	1.57E-4	1.21E-4	1.58E-4	1.38E-4	8.84E-4	9.91E-4
Test Loss MSE	1.44E-4	2.58E-4	1.56E-4	1.66E-4	1.45E-4	2.01E-4	1.57E-4	1.37E-3	1.50E-3
Error of entire timeseries (RMSE)	1.22E-2	1.87E-2	1.37E-2	1.28E-2	9.65E-3	1.24E-2	9.78E-3	3.72E-2	0.23
Error of entire timeseries (MAE)	4.19E-2	8.67E-2	5.85E-2	5.49E-2	2.50E-2	6.23E-2	2.95E-2	8.87E-2	0.26

**TABLEAU 2.4 : Testing radius cut-off values for graph construction on dataset 2.**

<b>Radii Range</b>	<b>0.1</b>	<b>0.15</b>	<b>0.2</b>	<b>0.25</b>	<b>0.3</b>	<b>0.35</b>	<b>0.4</b>	<b>0.45</b>	<b>0.5</b>
Train Loss MSE	8.44E-4	8.96E-4	1.08E-3	1.23E-3	1.13E-3	3.27E-3	3.01E-3	2.77E-3	2.86E-3
Validation Loss MSE	2.07E-4	2.08E-4	2.58E-4	2.28E-4	2.47E-4	4.79E-4	6.56E-4	6.74E-4	5.68E-4
Test Loss MSE	3.18E-3	6.54E-3	3.37E-3	3.37E-3	7.56E-3	9.16E-3	1.08E-2	4.62E-3	1.04E-2
Error of entire timeseries (RMSE)	1.80E-2	2.35E-2	2.04E-2	2.21E-2	3.06E-2	5.05E-2	4.60E-2	3.71E-2	4.13E-2
Error of entire timeseries (MAE)	6.79E-2	6.92E-2	7.31E-2	8.52E-2	8.69E-2	0.11	0.12	8.31E-2	0.12

**TABLEAU 2.5 : Mean error metrics and loss values of 10 GNN optimizations for the 2 datasets in meters.**

<b>Datasets</b>	<b>Training MSE loss (m)</b>	<b>Validation MSE loss (m)</b>	<b>Testing MSE loss (m)</b>	<b>RMSE-entire timeseries (m)</b>	<b>MAE-entire timeseries (m)</b>	<b>RMSE-test dataset (m)</b>	<b>MAE-test dataset (m)</b>	<b>R2-entire timeseries</b>
1	0.112	0.016	0.018	0.038	1.22	0.018	1.04	0.960
2	0.099	0.492	0.082	0.061	2.26	0.082	2.50	0.926

---

Four wells are randomly selected to present our results. Of these four wells, two are monitoring wells (wells 2 and 11), and the other two are pumping wells (wells 30 and 37). Results from dataset 1 and 2 of the four selected wells are shown in Fig 2.8 and Fig 2.9, respectively. The black dashed line is the PCA transient GW levels component scaled from 0 to 1, while the red lines are the 10 GCN-LSTM network predictions. Generally, for both datasets, there is good agreement between the simulated results from the numerical model and the GCN-LSTM predicted values. In Fig 2.8 and Fig 2.9, plots 1, 3, 5, and 7 are the GW levels on the normalized PCA space. The PCA inverse transform of the simulated and predicted in the original space measured in meters above sea level (MASL) are shown in plots 2, 4, 6, and 8. Instabilities in the inverted predicted PCA transient GW levels component in plots 2, 4, 6, and 8 can be seen in both datasets (sharp jumps, as seen in the red lines). These instabilities arise from step 5 of our framework, when the split PCA pumping rate component is brought together with the GCN-LSTM prediction. Each time step is 30 days; with such a coarse sampling, the GCN-LSTM is not able to accurately predict the exact time when the pumping rates change. Therefore, a mismatch is induced when combining the PCA pumping rate component and the predicted PCA GW levels from GCN-LSTM predictions in step 5. As a result, instabilities occur during the inverse transformation in this step, as seen by the sharp jumps in the model predictions (red lines in Fig 2.8 and Fig 2.9).

The loss and error metrics in meters are present in Table 2.5. The mean MSE loss values for training, validation, and testing of the GNN model for dataset 1 are 0.112 *m*, 0.016 *m*, and 0.018 *m*, respectively. For dataset 2, the mean MSE loss values for training, validation, and testing datasets are 0.099 *m*, 0.492 *m*, and 0.082 *m*, respectively.

The RMSE and the MAE are metrics used to evaluate the GCN-LSTM predictions on numerically simulated data. The RMSE and MAE values are calculated for the testing portion of the time series and for the whole time series that includes the training, validation, and testing. For dataset 1, the mean RMSE and MAE for the whole time series are 0.038 *m* and 1.22 *m*, respectively. As for dataset 2, the mean values are 0.061 *m* for RMSE and 2.26 *m* for MAE (Table 2.5).

To assess the GCN-LSTM generalization performance, the RMSE and MAE of the testing portion of the time series were calculated, since the model had never been trained or seen this portion of the data. For dataset 1, the mean RMSE is 0.018 *m* and the mean MAE is 1.04 *m*. The mean RMSE and MAE values for dataset 2 are 0.082 *m* and 2.50 *m*, respectively.

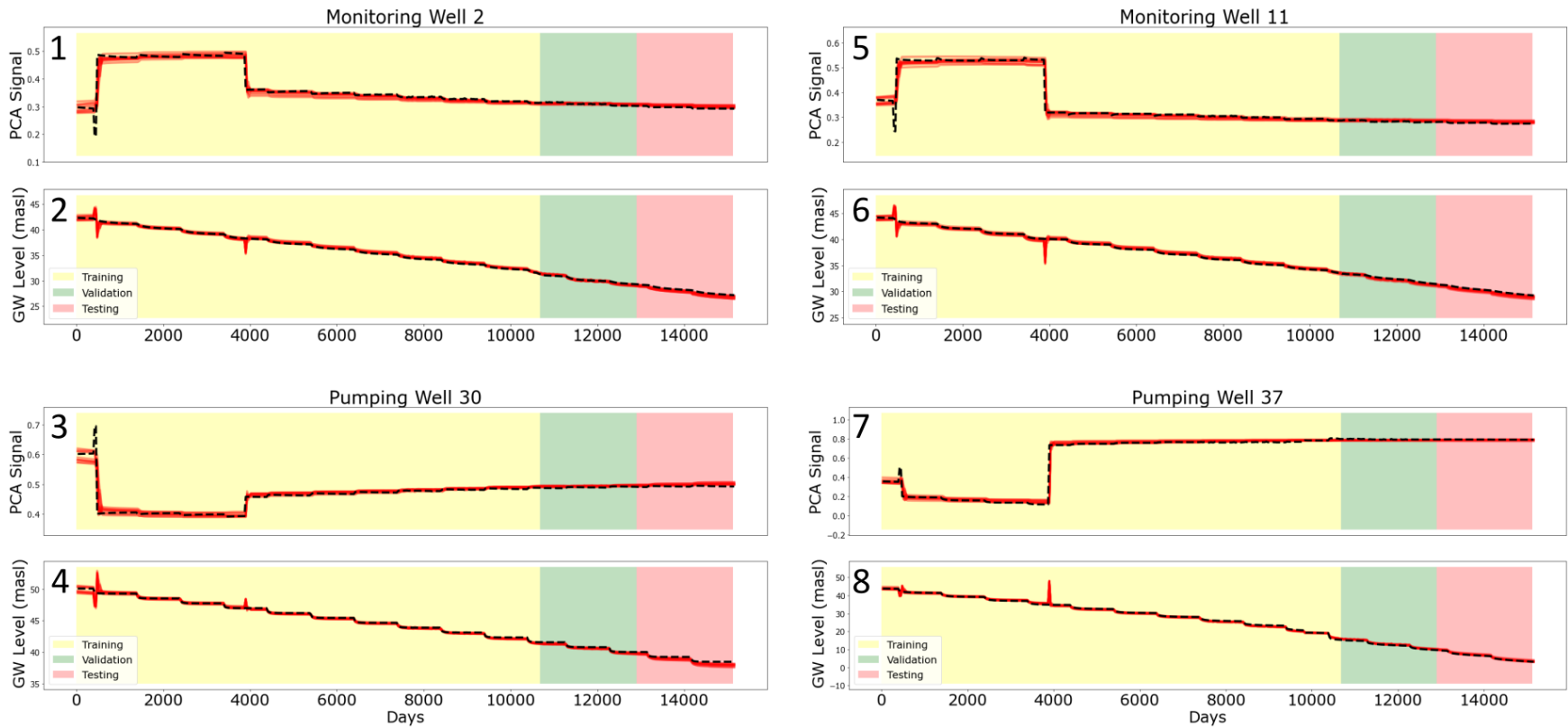
The error calculations for the testing portion of the time series indicate that the GCN-LSTM is capable of forecasting GW levels with high accuracy and is comparable to the simulated data from the numerical model. The error values indicate that the optimized GNN model is better suited for dataset 1 as the calculated error values for dataset 1 are smaller than the calculated error values for dataset 2 (Table 2.5).

The  $R^2$  values for both datasets are calculated at 0.960 for dataset 1 and 0.926 for dataset 2 (Table 2.5). The  $R^2$  values for both datasets are good, and results indicate the model performs better on dataset 1. One source of error is the uncertainty in the physical modeling and the small-scale

---

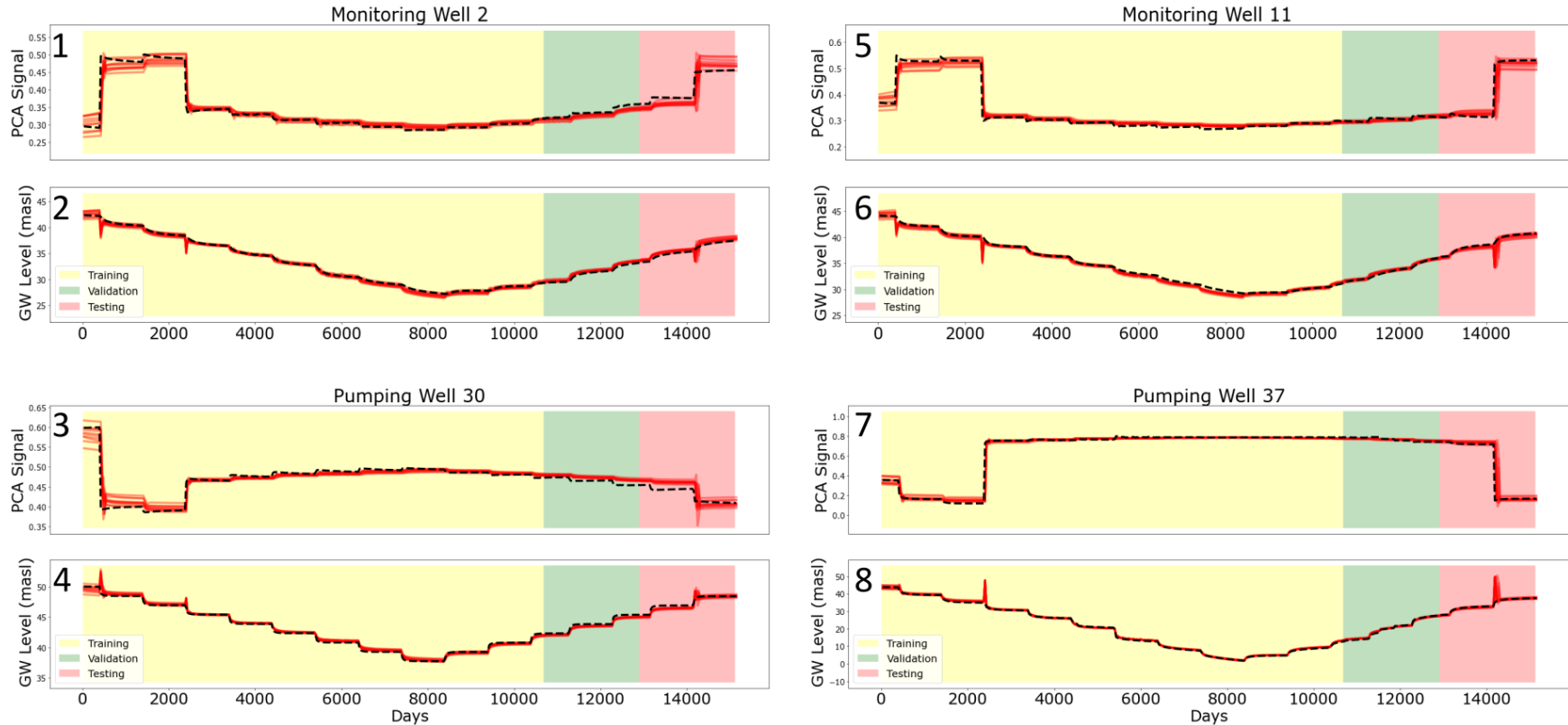
heterogeneity in the hydrogeological model, which locally changes the GW flow depending on the pumping scenarios. Another source of error comes from the hyperparameters, as only one set of best-fit hyperparameters is used to train both datasets. This could lead to one dataset performing better than the other. Hyperparameter tuning is a data-dependent and ad hoc exercise.

# Dataset 1



**FIGURE 2.8 : Dataset 1 of training, validation, and testing of the GNN model. Monitoring wells 2 and 11, and pumping wells 30 and 37 are shown for dataset 1. Plots 1, 3, 5, and 7 are the PCA GW signal. Plots 2, 4, 6, and 8 are the PCA inverse transformed GW levels measured in meters above sea level. The black dash line in all plots are the simulated GW levels and red lines are the predicted values from GCN-LSTM network.**

## Dataset 2



**FIGURE 2.9 :** Dataset 2 of training, validation, and testing of the GNN model. Monitoring wells 2 and 11, and pumping wells 30 and 37 are shown for dataset 1. Plots 1, 3, 5, and 7 are the PCA GW signal. Plots 2, 4, 6, and 8 are the PCA inverse transformed GW levels measured in meters above sea level. The black dash line in all plots are the simulated GW levels and red lines are the predicted values from GCN-LSTM network.

---

## 2.6 Discussion and Conclusion

In conclusion, the framework utilizing a GCN-LSTM network demonstrates robustness in forecasting GW levels across various pumping scenarios by incorporating spatial and temporal data from monitoring and pumping wells. For graph construction, spatial features extracted from the numerical model were employed to establish the spatial connectivity between wells. A radius cut-off was determined and applied to the calculated Euclidean distances between wells to construct this spatial connectivity. To address the dimensionality constraints of the network, a PCA transformation was applied to our temporal features to reduce dimensionality by splitting the components. Although the PCA transformation technique can introduce instability during inverse transformations, as evidenced by sharp jumps in the red lines in Fig 2.8 and Fig 2.9, the network still successfully reproduces trends in GW levels under different pumping conditions.

This framework can also be applied to field-measured data for GW level forecasting, as all features used for network training and adjacency matrix construction are measurable in the field. Field data, such as well locations and estimated hydraulic conductivity, can be used to construct the adjacency matrix, and measured GW levels can be employed for network training, eliminating the need for numerically simulated data. However, this approach is contingent upon the availability of time series data for network training.

Implicit ML models can be difficult to interpret and may intimidate practitioners due to their "black box" nature, particularly in the context of environmental modeling and resource management. Therefore, explicit geostatistical methods play a crucial role in our model training. They provide a framework that accounts for spatial variability and correlation within the data, enabling GCN-LSTM to accurately capture the underlying spatial patterns and relationships. This approach enhances the foundational understanding of the models and reassures practitioners in their decision-making while offering significant computational efficiency. It also allows the models to make more reliable predictions and generate insights that are grounded in the physical realities of the system under study. Incorporating geostatistical methods into the training process of ML models can improve their robustness, interpretability, and generalization capabilities, ultimately leading to more informed decision-making in complex geospatial contexts.

While the future of using complex numerical models remains uncertain, the process is often time-consuming and resource intensive. Running each ensemble simulation can take several days to several weeks, and requires a highly qualified modeler, which hinders the democratization and affordability of hydrogeological models. This challenge is particularly relevant in regions like Quebec, Canada, where GW management is increasingly integrated into land management and legislative frameworks. Our proposed methodology leverages an explicit geostatistical framework and involves training a ML model on synthetic data from ensembles of stochastic hydrogeological realizations. This approach aims to forecast the effects of pumping on water levels at a semi-regional scale, providing a more efficient and accessible solution for ongoing model updates and legisla-

---

tive compliance. More specifically, our surrogate model can be used, for example, to inform land use planning in small municipalities to understand the effect of changing pumping patterns, or to understand the impacts of a new pumping well on GW sustainability. It eliminates the need for a highly skilled modeler to re-run simulations, significantly reducing the time and cost, as our trained GCN-LSTM network can instantly forecast GW levels (depending on available computational power, forecasting with our trained GCN-LSTM network can be instantaneous to a few minutes). An alternative approach to using our GNN framework would be to couple it with the numerical model, effectively creating a digital twin. This training strategy would leverage numerical simulations of the GW flow to train the network for future forecasting. Such an approach would aim to reduce computational demands and time requirements by avoiding the need to perform extensive transient simulations.

The framework in its current form has several limitations. The network is not designed for mass transport, which could be changed to forecast the advective component of mass transport, and it has only been trained with realistic pumping scenarios and not with more complex unrealistic pumping scenarios to further test the generalization capabilities of the network.

## **Acknowledgments**

We thank Natural Resources Canada for funding this project and DHI for providing a student license for the numerical model simulations. The authors would like to thank Natural Resources Canada for funding this project and DHI (FEFLOW) for providing a student license to carry out the numerical model simulations. The authors would also like to thank the reviewers and the guest editor for their thoughtful feedback.

## **Availability of data and material**

Data and other materials are available for download on Github. [https://github.com/XiaoXia10/GWforecast\\_GNNFramework](https://github.com/XiaoXia10/GWforecast_GNNFramework)

## **Statements and Declarations**

### **Conflicts of Interest**

To the best of the authors' knowledge, there are no conflicts of interest or competing interests in this project.

---

## **Funding**

Funding for this study is provided by Natural Resources Canada.



### **3 ARTICLE II : COMPARISON OF FORECASTING CAPABILITIES OF DEEP LEARNING MODELS WITH VARYING TIME SERIES RESOLUTION FOR ACCURATE KARST SPRING DISCHARGE PREDICTIONS**

---

**Comparaison des performances prédictives des modèles d'apprentissage profond selon la résolution des séries temporelles pour une estimation précise du débit des sources karstiques**

Xiao Xia Liang<sup>1</sup>, Philippe Renard<sup>2</sup>, Erwan Gloaguen<sup>1</sup>, Julien Straubhaar<sup>2</sup>, and Maxime Claprood<sup>3</sup>

<sup>1</sup> Centre Eau Terre Environnement, Institut National de la Recherche Scientifique, Québec, Canada

<sup>2</sup> Centre d'Hydrogéologie et de Géothermie, Université de Neuchâtel, Neuchâtel, Suisse

<sup>3</sup> Département des sciences appliquées, Université du Québec à Chicoutimi, Chicoutimi, Canada

#### **Contributions des auteurs**

Xiao Xia Liang : Rédaction du manuscrit, méthodologie, programmation, nettoyage et compilation des données et conceptualisation.

Philippe Renard : Supervision, méthodologie, programmation et révision du manuscrit.

Erwan Gloaguen : Supervision, programmation et révision du manuscrit.

Julien Straubhaar : Programmation et révision du manuscrit.

Maxime Claprood : Supervision et révision du manuscrit.

#### **Publication ciblée**

Journal : Groundwater - Soumis

Date de soumission : mars 2025

#### **Objectif de l'article**

Puisque nous avons réussi à utiliser ce cadre comme substitut aux modèles physiques, et que notre modèle RNG peut être entraîné sur des données synthétiques pour prévoir avec précision les niveaux des eaux souterraines, les questions essentielles demeurent : peut-on se passer d'un modèle physique pour la prévision des niveaux d'eau souterraine si nous avons les données spatio-temporelles adéquates? Peut-on entraîner des modèles RNG sur des données de terrain et obtenir des prévisions précises des niveaux d'eau souterraine? Forts du succès du premier objectif, nous

---

poursuivons en évaluant les capacités de prévision spatio-temporelle des modèles GNN entraînés sur des données mesurées sur le terrain.

Les principaux objectifs de ce chapitre sont d'examiner la capacité des modèles d'AA, tels que les modèles RNG, à prévoir avec précision les niveaux des eaux souterraines et d'identifier les critères clés pour obtenir des résultats précis. Ce faisant, nous pouvons évaluer s'il est possible de se passer de la construction de modèles physiques pour la prévision des niveaux d'eau souterraine.

**Résumé** Les aquifères karstiques constituent des sources d'eau essentielles dans de nombreuses régions du monde. Ces systèmes présentent une forte hétérogénéité et anisotropie, caractérisés par un écoulement rapide de l'eau à travers des conduits et une infiltration plus lente à travers les matrices rocheuses et les fractures. Par conséquent, la modélisation physique et la prévision précise des débits des sources pour une gestion durable de l'eau restent des défis majeurs. Dans cette étude, nous exploitons les capacités des réseaux neuronaux basés sur les données pour prédire les débits des sources sans recourir à une modélisation physique du système karstique. Nous évaluons trois architectures d'apprentissage profond : un auto-encodeur LSTM, un auto-encodeur GRU, et le modèle GWN. Ces modèles sont testés sur un jeu de données karstique provenant de Suisse, en intégrant des séries temporelles à différentes résolutions afin d'évaluer leurs performances de prévision sur divers horizons temporels.

Les objectifs de cette étude sont d'évaluer la précision prédictive de chaque modèle et de déterminer la résolution des données nécessaire pour obtenir des prévisions fiables. Les résultats montrent que tous les modèles testés permettent de prévoir efficacement les débits en vue de la gestion des ressources en eau. Toutefois, le modèle GWN surpasse les modèles auto-encodeurs pour toutes les durées de prévision lorsque des données haute résolution sont disponibles. De plus, en présence de données à faible résolution, le modèle GWN dépasse significativement les performances des modèles auto-encodeurs.

### **Abstract**

Karst aquifers serve as critical water sources in many parts of the world. These systems exhibit high heterogeneity and anisotropy, characterized by rapid water flow through conduits and slower seepage through rock matrices and fractures. Consequently, physically modeling and accurately forecasting spring discharges for sustainable water management remain challenging. In this study, we harness the capabilities of data-driven neural networks to forecast spring discharges without relying on physical modeling of the karst system. We evaluate three deep learning architectures : a Long Short-Term Memory (LSTM) auto-encoder, a Gated Recurrent Unit (GRU) auto-encoder, and the Graph WaveNet (GWN) model. These models are tested on a karst dataset from Switzerland, incorporating time series data at different resolutions to assess forecasting performance over varying prediction horizons. The objectives are to evaluate the predictive accuracy of each model and to determine the data resolution required for reliable forecasts. The results demonstrate that all tested models effectively predict discharge for water resource management. However, the GWN model outperforms the auto-encoder models across all forecast durations when high-resolution

---

data is available. Furthermore, under low-resolution data conditions, the GWN model significantly surpasses the performance of the auto-encoder models.

**Keywords :** Data Resolution, Karst Spring Forecast, Machine Learning, Deep Neural Network, Auto-encoder Model, Graph WaveNet Model, Time Series Data

### 3.1 Introduction

Approximately 10% of the world's population relies on karst aquifers for drinking water (Stevanović, 2018), making this a critical source of potable water worldwide (Stevanović, 2019). Considering the importance of this resource, it is crucial to have tools with accurate spring discharge forecasting capabilities for effective resource management. Here, a comparison of data-driven models is presented with a focus on the recent advancements of deep neural network (DNN) or deep learning (DL) models. Our work explores the analysis and forecasting of karst spring discharge as a data-driven problem.

Various approaches to modeling karst spring discharges have been compared in Jeannin et al. (2021). Traditional methods include constructing fully distributed hydrogeological numerical models based on solving process-based governing equations (Doummar et al., 2018, 2012; Vuilleumier et al., 2019). Semi-distributed pipe flow models, such as epikarst and conduits, are also used (Sezen et al., 2019; Chang et al., 2015). Karst aquifers present significant challenges for physical modeling due to their high spatial heterogeneity and anisotropic flow characteristics. This heterogeneity, inherent to the formation of karst conduits, is evident in the multi-scale porosity of carbonate rocks that is influenced by geological features such as bedrock composition and strata orientation. These intricate conduit networks lead to complex flow dynamics, characterized by fast conduit flow and slow seepage through fractures. Collecting data across all flow phases and accurately mapping the spatial heterogeneity of these systems is challenging; therefore, it complicates the construction of reliable physical models (Chen et al., 2018).

As more data becomes available through increases in groundwater monitoring networks (e.g., United States Geological Survey, Groundwater Watch - Active Water Level Network), data-driven models should be considered for time series forecasting. As suggested by Shapiro et al. (2022), groundwater hydrology could be reframed as a data-driven science due to the increasing availability of groundwater data. In line with this school of thought, Bakker et al. (2019) have suggested that time series analysis, specifically groundwater heads, is a data-driven approach to find a relationship between the input and output signal. However, Bakker et al. (2019) argued that data-driven gray-box models, which they classified as semi-physical models that apply algorithms that have some physical basis, should be used instead of the black-box, neural network models. There exist neural networks, such as graph neural networks (GNNs), that incorporate some physical basis of an aquifer system into the neural network. In this case, the GNNs could be considered as a gray-box model. In the opposite school of thought, Siegel et al. (2019) had argued that using artificial

---

intelligence models such as neural networks will neglect centuries of scientific understanding of physical processes to empirically look for relationship among data. To overcome some of these challenges, physics-informed neural networks (PINNs) have been invented and can include equations of physical processes such as contaminant advection-dispersion, saturated, and unsaturated flows in neural networks to constrain the results of the neural network (He et al., 2020, 2021a; Wu et al., 2021; Tartakovsky et al., 2020). Siegel et al. (2019) cites Aristotle to not to push scientific analysis further than the precision and accuracy of data allow. In this study, we will precisely test the temporal resolution of time series data to observe where the accuracy of predictions lies by using these data to train data-driven models.

The most widely used DNN algorithms to forecast spring discharge are recurrent neural networks (RNNs), such as LSTM (Cheng et al., 2020; Ma et al., 2024a; Opoku et al., 2024; Zhang et al., 2024; An et al., 2020; Fang et al., 2022). The popularity of LSTM for spring discharge is understandable, as it is developed to regulate the gradient explosion problem in long time series data (Hochreiter et al., 1997). Since the LSTM is a popular network for time series forecasting, we consider a type of LSTM as a model for comparison.

Recently, transformer models have become the go-to network for training large language models (Xue, 2020; Devlin, 2018; Radford, 2018). Pölz et al. (2024) used a transformer model to forecast the discharges of three different karst springs located in Austria. Using the transformer model, their Nash-Sutcliffe Efficiency (NSE) metric for the 3 springs with different forecasting lead times ranges from 0.27 to 0.99. This is a wide range of accuracy with no consistency in between the 3 springs and lead times. The transformer models are certainly powerful, but they require a large amount of data to train. For example, to train a language model, the minimum amount of samples needed is in the range of millions (Popel et al., 2018). At this moment, extra large datasets are still difficult to obtain in hydrogeology.

Gai et al. (2023) used graph neural networks (GNNs) to forecast spring discharges in northern China; the GNNs used in their study are the Chebnet (Defferrard et al., 2016), and the Graph Convolutional Network (GCN) (Kipf et al., 2016). The GCN can be considered to be a more computationally efficient model of the ChebNet. This is done by simplifying the Chebnet to a first-degree approximation of the Chebyshev polynomial while applying a Kipf normalization to each of the model layers. Luckily, unlike the transformer models, GNN models do not require a large amount of data to train. However, these two chosen GNNs from Gai et al. (2023) are not spatio-temporal networks designed to forecast sequential data, but are used for spatial classifications of nodes.

Another common deep neural network is Convolutional Neural Networks (CNNs). It is currently the backbone of computer vision, which is applied to technologies such as facial recognition and self-driving cars (Khan et al., 2018; Lou et al., 2020; Do et al., 2018; Kanagaraj et al., 2021). Wunsch et al. (2022) used a coupled model with a stacked 2D, then 1D CNN architecture to forecast 3 different karst spring discharges throughout Europe. They trained their 2D-1D CNN model with spatially disturbed meteorological data to forecast spring discharge. The NSE metric for spring

---

discharge predictions ranges from 0.73 to 0.87. However, this method requires using meteorological data that are not necessarily easily correlated with spring discharge. Advancements in CNNs have seen their adaptation into more complex architectures designed for temporal prediction (Van Den Oord et al., 2016; Wu et al., 2019). Van Den Oord et al. (2016) developed the basis of the WaveNet model on a gated dilated causal convolution method (Yu et al., 2016) for sequence-to-sequence prediction that is used in text-to-speech applications. Inspired by the WaveNet model, Wu et al. (2019) developed the Graph WaveNet (GWN) model for spatio-temporal traffic predictions. Effectively, Wu et al. (2019) changed the gated dilated causal convolution to a gated temporal convolutional network (TCN) (Dauphin et al., 2017) and changed the CNN framework of the WaveNet model into a graph framework calling their model Graph WaveNet (GWN).

In this study, we chose three DNN models for comparison, an LSTM and GRU auto-encoder model (Cho et al., 2014), and the GWN model (Wu et al., 2019). The 3 models are chosen for their proven capacity to carry out temporal forecasting. The main criteria for model selection are 1, the model is a DNN, 2, the model is intended and/or designed to forecast sequential data, and 3, it has the capability to do sequence-to-sequence forecasting. We are interested in testing the temporal forecasting lengths of differing resolutions of data; therefore, the models must have the capacity to forecast different lengths of time.

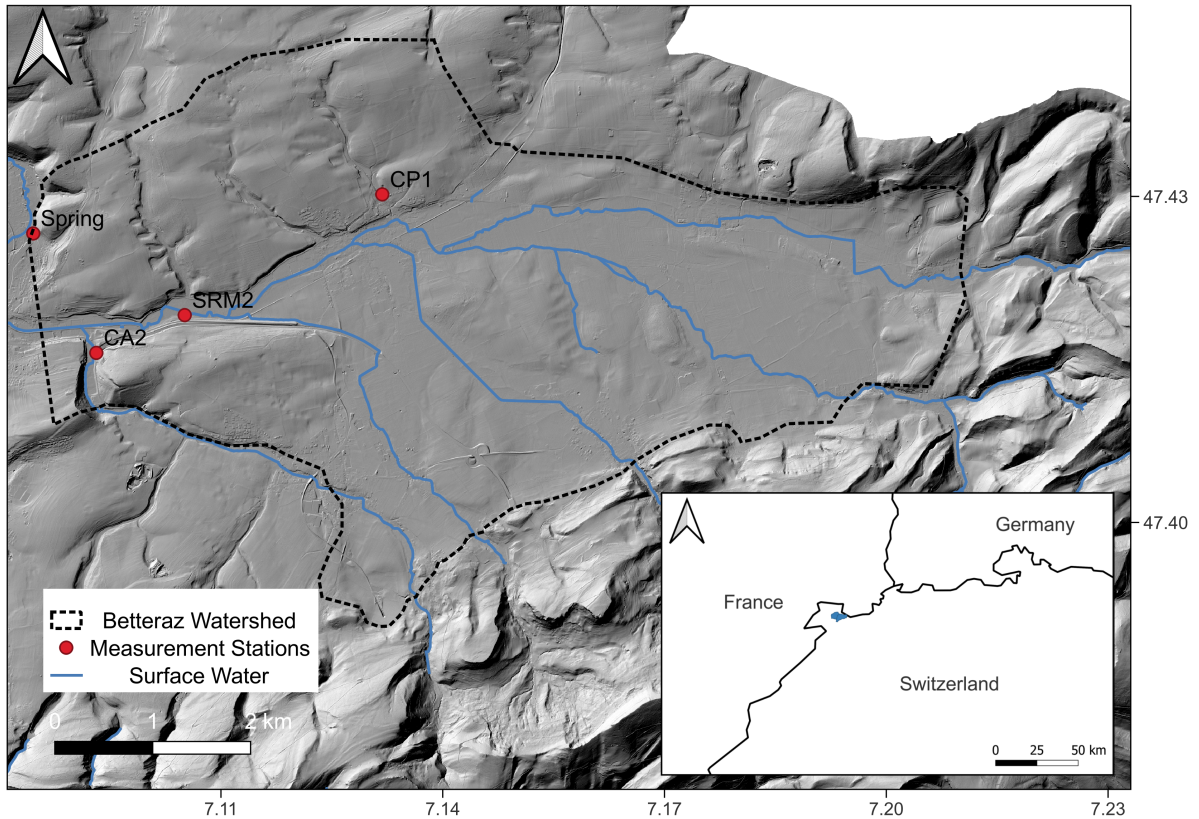
This article is structured as follows. The next section provides an overview of the karst system from which the time series data were collected. This is followed by a brief description of the data and an explanation of the DNN models selected for this study. Next, the experimental setup for training these models is presented. The second-to-last section discusses the forecasting results from all models. Finally, the last section includes a discussion and the overall conclusions of our paper.

## **3.2 Study Area**

The Betteraz spring is located in the northwest corner of Switzerland in the Jura mountains that border France. This study area has been the subject of several studies in recent years (Trunz, 2017; Miville, 2020; Zuercher, 2018). The Malm karst aquifer feeds the Betteraz spring; the aquifer is interspersed with limestone and marl. The limestone is the transmissive medium, while the marl acts as an aquiclude. This system is located under the Quaternary alluvial deposit and on top of an impermeable clay layer that separates the deep Dogger aquifer from this karst system (Trunz, 2017).

A total of 9 monitoring stations were installed to collect groundwater level data in this study area; one station is installed in the spring catchment basin, and the other 8 stations are installed upstream of the spring. For this study, we used data from 4 monitoring stations. We did not use data from all the stations as some of the stations are close in proximity. Therefore, two stations are measuring the same time series signal, and other stations have large missing data gaps greater than 30% of the total time series. The monitoring stations used are the Betteraz spring station,

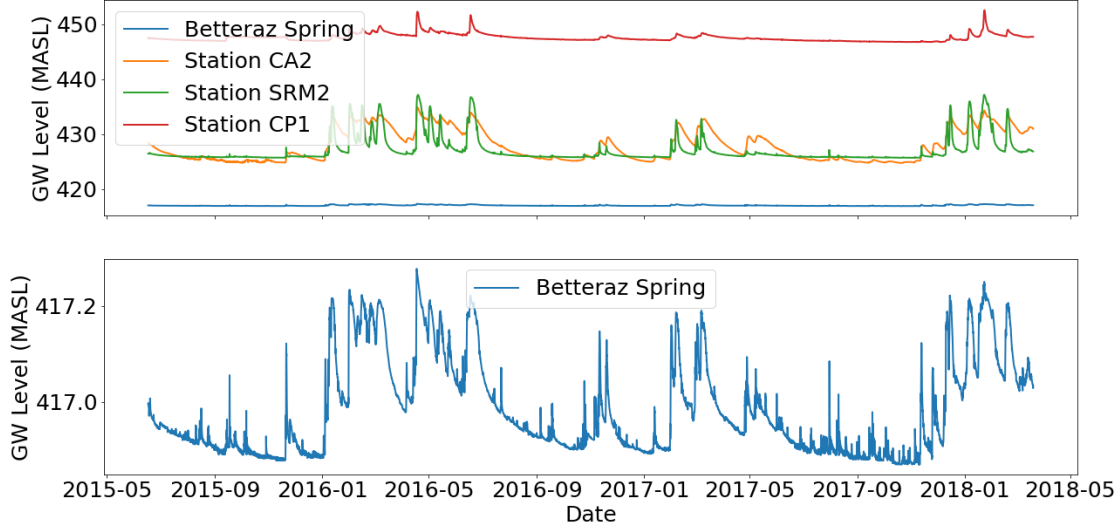
the CA2 station, the SRM2 station, and the CP1 station. The locations of these stations can be seen in Figure 3.1. The spring station is installed inside a spring water catchment basin, where a bunker-like structure was constructed around the spring to collect water. The CA2 and SRM2 stations are located southeast of the spring, while the CP1 station is located east of the spring and is also the furthest upstream of all the stations (Figure 3.1).



**FIGURE 3.1 : Betteraz spring study area. Data collected from the 4 stations, Betteraz Spring, CA2, SRM2 and CP1 are used to train the DNN models**

### 3.2.1 Time Series Data

The data used in this study have been collected between June 17, 2015 and March 19, 2018, with hourly measurements (Trunz, 2017; Zuercher, 2018). This is a relatively small dataset that is representative of typical hydrological or hydrogeological studies. The complete set of data used in this study can be seen in Figure 3.2.



**FIGURE 3.2 :** The top plot contains all time series data from the 4 monitoring stations. The bottom plot is data from the Betteraz Spring station.

### 3.3 DNN Models

In this section, the three DNN models selected to be used in study are introduced.

#### 3.3.1 GWN model

The GWN is considered in this study because it is a multivariate spatio-temporal DNN designed to forecast the temporal vehicle velocities on spatially connected roadways. We apply these spatio-temporal traffic forecasting philosophies to the karst aquifers to forecast their spring discharges. The GWN model fulfills the 3 model selection criteria and also uses some hydrogeological information such as the locations of the monitoring stations and the general groundwater flow directions to construct the required graph for model training. This model can be considered as a gray-box model, since it uses some physical information of the aquifer system.

The GWN model was developed by Wu et al. (2019) and later improved by Shleifer et al. (2019). In this study, we adapted the improved GWN model from Shleifer et al. (2019) to forecast karst spring discharges. Refer to Wu et al. (2019) and Shleifer et al. (2019) for a detailed description of the GWN model. The GWN contains two main components, a gated Temporal Convolutional Network (TCN) and a Graph Convolutional Network (GCN) (Figure 3.3). The TCN employs a causal dilated convolutional method (Yu et al., 2016) to learn temporal patterns. These learned temporal patterns are sent to the GCN layer, where the spatial connections of the measuring stations are learned through a graph. Through stacking the spatio-temporal layers, GWN is able to tackle long-term

temporal patterns at the top layers and short-term temporal patterns at the bottom layers. The skip connections are used to regulate the network. The long and short-term information is summed from the spatio-temporal layers, then sent through two fully connected layers to get the output or forecast.

A GNN requires its data to be in a graph form. A graph consists of nodes and edges and can be presented mathematically by an adjacency matrix (Table 3.1). Here, the nodes represent our measuring stations, while the edges represent the groundwater spatial connections between the stations (Figure 3.4). There are many ways to construct a graph; we constructed our graph by adapting the Gaussian covariance function (Li et al., 2017) to do a pairwise distance calculation and applying a groundwater flow direction filter to make a directed graph. A node in a graph indexes a row and a column in the adjacency matrix, and the entries in the matrix are the edges of the graph (Table 3.1). If an adjacency matrix entry is zero, then no connection exists between the nodes. The adjacency matrix is constructed by

$$A_{i,j} = \exp\left(-\left(\frac{\text{dist}(v_i, v_j)}{\alpha}\right)^2\right) \cdot D_{i,j}, \quad (3.1)$$

where  $i$  and  $j$  are the rows and columns of the adjacency matrix,  $v$  is a node vector,  $\text{dist}()$  is the distance calculation,  $\alpha$  is a scaling factor, and  $D$  is the directional matrix constructed by calculating the difference in elevation between stations; if the difference is greater than or equal to zero, the entry in  $D$  is 1; otherwise, it is 0.

**TABLEAU 3.1 : The weighted and directed adjacency matrix used for GWN model training.**

	Spring	CA2	SRM2	CP1
Spring	1.0	0.0	0.0	0.0
CA2	0.862	1.0	0.0	0.0
SRM2	0.787	0.928	1.0	0.0
CP1	0.368	0.420	0.647	1.0

In the GWN framework, the TCN can be represented as

$$Z_{TCN} = \tanh(U_a \star X + a) \odot \sigma(U_b \star X + b), \quad (3.2)$$

where  $U_a$ ,  $U_b$ ,  $a$ , and  $b$  are model learned parameters,  $X$  is the input data,  $\tanh(\cdot)$  is the tangent hyperbolic activation function,  $\sigma(\cdot)$  is the sigmoid activation function,  $\odot$  is the element-wise product, and  $\star$  is the dilated causal convolution operator.

For the GCN layer in this framework, a diffusion convolution is used as it is proven to be more effective at spatio-temporal modeling (Li et al., 2017). The graph convolution layer with forward and backward diffusions over  $K$  finite steps is written as

$$Z_{GCN} = \sum_{k=0}^K \left( X \cdot P_f^k \cdot W_{f_k} + X \cdot P_b^k \cdot W_{b_k} \right). \quad (3.3)$$

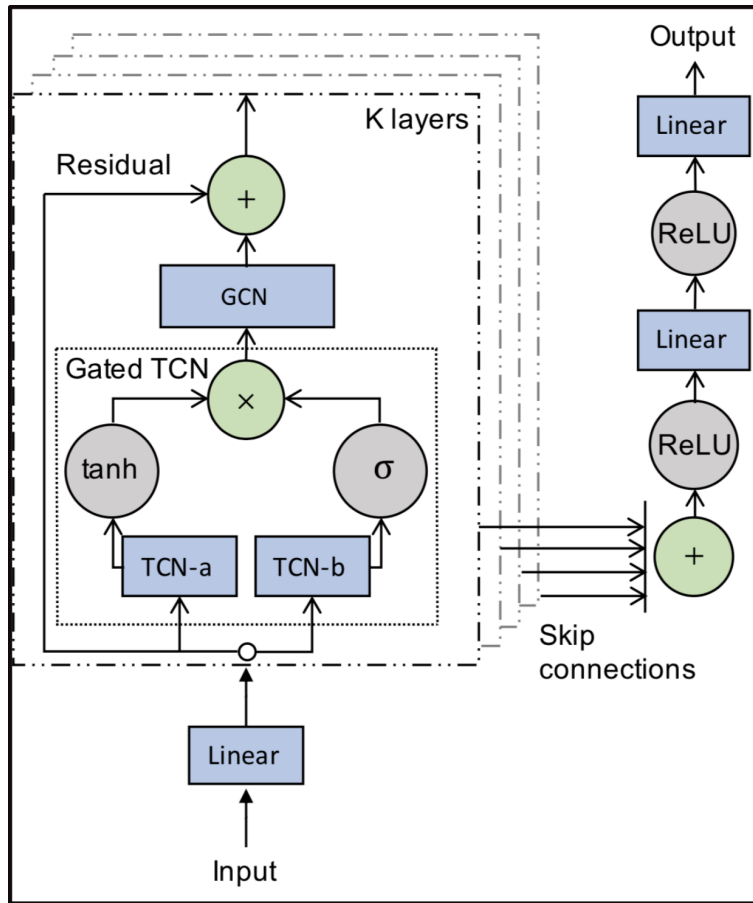


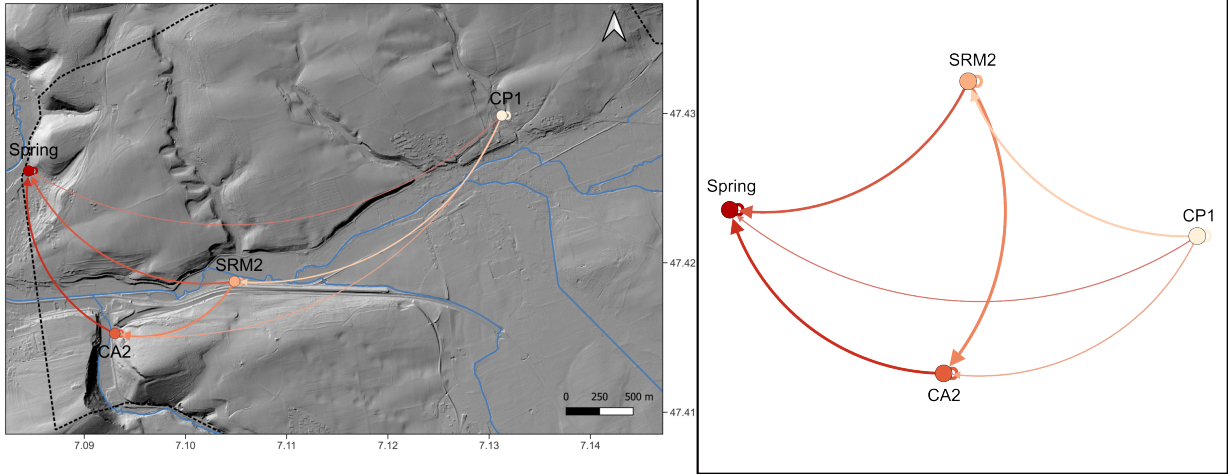
FIGURE 3.3 : The Graph WaveNet framework. This figure is taken from Wu et al. (2019).

In this equation, each column in the input data  $X$  corresponds to a node of the graph (i.e. a station). The forward transition (or diffusion) matrix  $P_f$  is defined from the adjacency matrix  $A$  by  $P_{f_{i,j}} = A_{i,j} / \sum_l A_{i,l}$ , where the  $i$ -th row represents the diffusion from the node (station)  $i$ . The backward transition matrix  $P_b$  is defined in the same manner from  $A^T$ . The power of these transition matrices allows us to propagate the diffusion over several steps, and for each step  $k$ , the matrices  $W_{f_k}$  and  $W_{b_k}$  are model-learned parameters.

### 3.3.2 GRU and LSTM Auto-Encoder Models

The auto-encoder models are chosen for their ability to learn the data characteristics through the encoding and decoding process with the benefits of gated recursive properties of an LSTM and GRU unit to prevent gradient explosion in long time series forecasting (Hochreiter et al., 1997). The auto-encoder models also satisfy the 3 model selection criteria listed in the Introduction section.

The GRU unit (Figure 3.5) can be viewed as a simplified LSTM cell with only two gates and one state (Cho et al., 2014). This makes the GRU unit more computationally efficient than the LSTM because the LSTM cell contains three gates and two states (Figure 3.6). The GRU unit contains a



**FIGURE 3.4 : Illustrations of the directed graph used for GWN model training. On the left, the physical location of the stations are shown as nodes with directed edges that follow the general flow direction of groundwater. On the right, a graph projection of nodes and edges are shown. This diagram is a projection, therefore, the node stations are not based on physical locations, but rather a random projection in that data space.**

reset and update gate, along with a hidden state. The reset gate decides whether the last hidden state should be kept or disregarded; then the result is combined with the current input data, and a candidate hidden state for the next step is computed. The update gate decides whether the last hidden state will be updated with this new candidate for the hidden state.

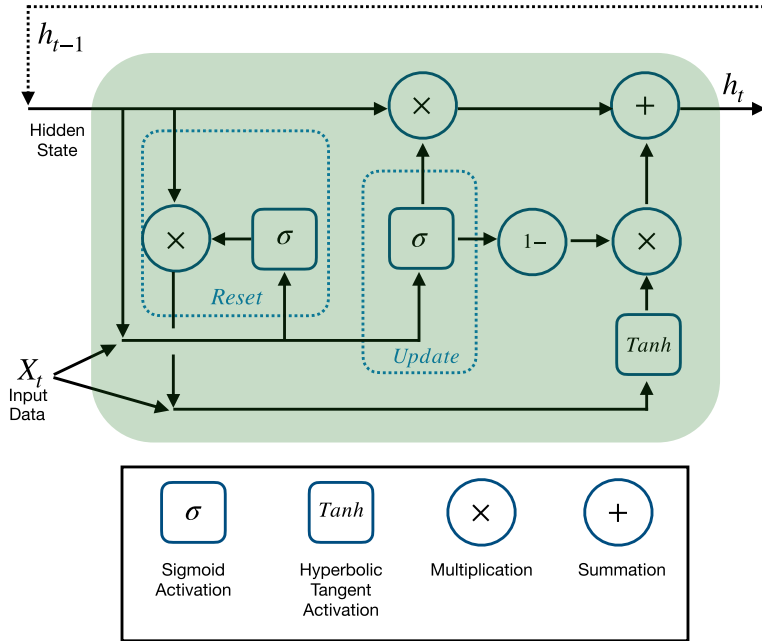
The LSTM unit contains a cell with 3 gates and 2 states (Figure 3.6). The gates are input, output, and forget. The gates regulate the in and out flow of information in the cell, while the cell remembers information over a time interval. The input gate decides which new input data should be stored in the cell state. The output gate decides which information or data in the current cell state should be outputted, considering the current and previous of the two states. The forget gate decides which information should be forgotten or discarded from the previous cell state.

An auto-encoder model contains an encoder and a decoder pair (Cho et al., 2014). The encoder maps an input time series to a fixed-length vector,  $c$ , and the decoder maps the vector representation back to a target time series (Figure 3.7). The encoder and decoder are mapped with the encoding function,  $f_e$ , and the decoding function,  $f_d$ . In our case, the mapping functions for the LSTM auto-encoder model are the LSTM unit functions (Hochreiter et al., 1997). While the mapping functions for the GRU auto-encoder model are the GRU unit functions (Cho et al., 2014).

For the GRU auto-encoder model, there is a hidden state that is recursively looped back into the unit (Figure 3.5). In the encoder, the hidden state can be written as

$$h_t = f_e(h_{t-1}, X_t), \quad (3.4)$$

where  $h_t$  is the hidden state of the current time step,  $h_{t-1}$  is the previous time step hidden state,  $f(\cdot)$  is the mapping of the GRU unit functions, and  $X_t$  is the current input data.



**FIGURE 3.5 : GRU unit used in the GRU auto-encoder (Cho et al., 2014). This unit contains 2 gates : the reset and the update. It also contains a hidden state that loops back into the next recursive step.**

In the decoder, the hidden state depends on the encoded vector  $c$  and the target  $Y$ . The decoder hidden state can be written as

$$h_t = f_d(h_{t-1}, Y_{t-1}, c). \quad (3.5)$$

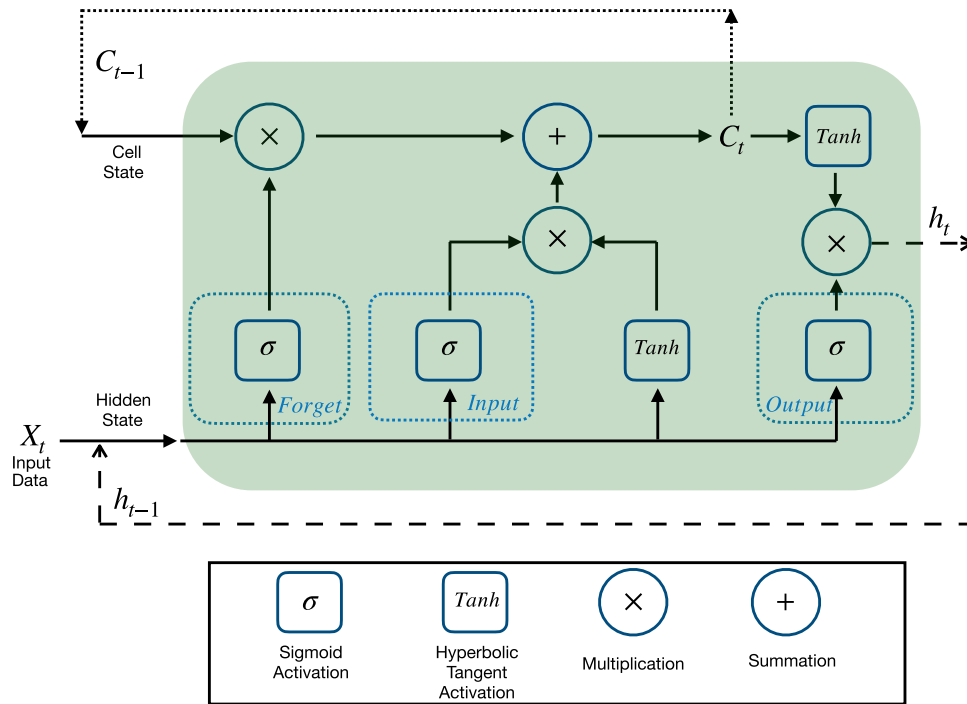
The LSTM unit contains two states, the hidden state and the cell state. The output of those 2 states is needed for the next recursive step in the LSTM auto-encoder. The hidden state of the encoder and decoder can be written as Equations 3.4 and 3.5, respectively. The difference is in the  $f_e$  and  $f_d$  mapping functions; they are the LSTM unit functions instead of the GRU unit functions. The encoder mapping function for the LSTM's cell state can be written as

$$C_t = f_e(C_{t-1}, X_t), \quad (3.6)$$

where  $C_t$  is the current cell state and  $C_{t-1}$  is the past time step of the cell state.

The decoder LSTM cell state can be written as

$$C_t = f_d(C_{t-1}, Y_{t-1}, c). \quad (3.7)$$

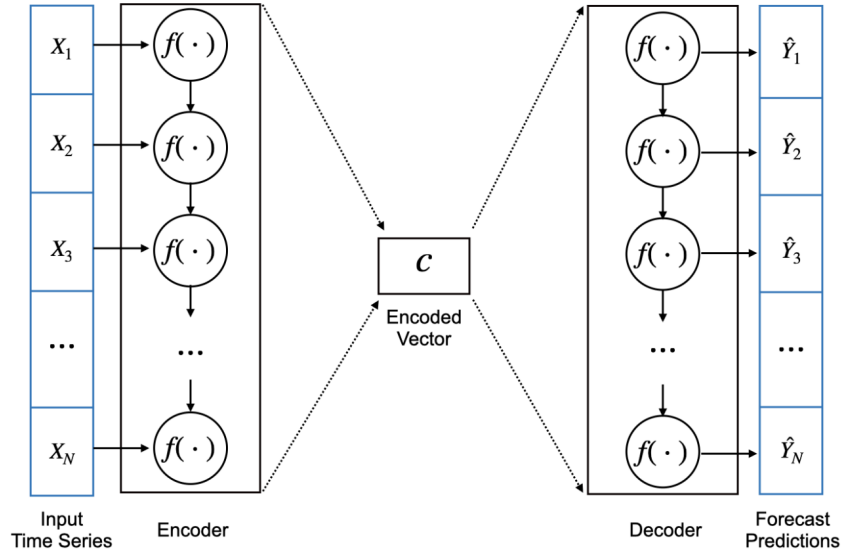


**FIGURE 3.6 :** The LSTM unit contains 3 gates : the forget, the input and the output. There are 2 states, the hidden and the cell, that loops back into the unit for the next recursive step.

### 3.4 Experiment Setup

In order to test the temporal resolution of the data, we resampled the hourly data. The hourly data are resampled into 4, 8, 12, and 24-hour time steps. For example, to obtain the resampled 4-hour time step measurements, the mean value of the 4 hourly measured data is used. The resampled datasets are further compiled into a sequence-to-sequence format for model training to test the forecasting capabilities of the DNN models. All data compilation for model training can be found in Table 3.2.

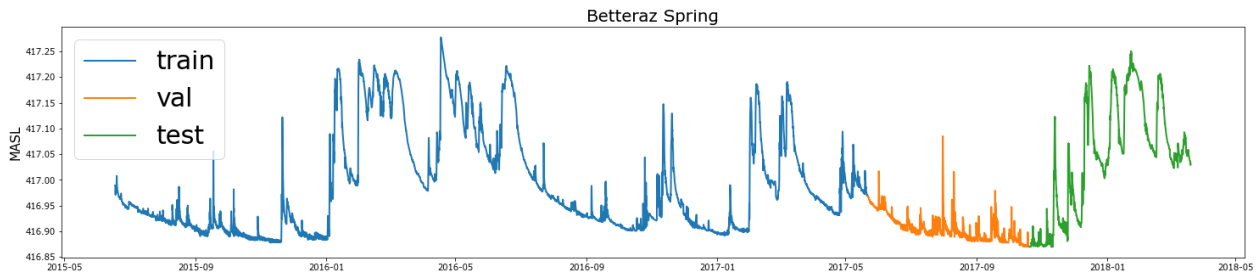
The sequence-to-sequence method is a sliding window technique (Figure 3.9). Figure 3.9 demonstrates sequence-to-sequence ( $\mathbf{X}$  length = 3,  $\mathbf{Y}$  length = 3, shift length = 3) data preparation for model training. The training sequence of  $\mathbf{X}$  can be written as  $[x_0, x_{h-1}], [x_{shift}, x_{shift+h-1}], [x_{2shift+h-1}], \dots, [x_{T-h-n}, x_{T-n-1}]$ , and the  $\mathbf{Y}$  forecasting sequence can be written as  $[x_h, x_{h+n-1}], [x_{shift+h}, x_{shift+h+n-1}], [x_{2shift+h}, x_{2shift+h+n-1}], \dots, [x_{T-n}, x_{T-1}]$ . Here  $x$  are the measured data,  $h$  is the number of training lengths,  $n$  is the length of forecast, shift is the sliding window length while sliding over the data, and  $T$  is the length of the time series data. Each input set of  $\mathbf{X}$  contains  $N$  number of training sets,  $h$  time step length for training by  $s$



**FIGURE 3.7 : An illustration of an auto-encoder model. The input data  $X$  is inputted into the encoder mapping function and a fixed length encoded vector,  $c$ , is obtained and forwarded to the decoder. The decoder then decodes the encoded information in the mapping function and makes a prediction output,  $Y$ .**

number of monitoring stations, given  $\mathbf{X} \in \mathbb{R}^{N \times h \times s}$ . While  $\mathbf{Y}$  contains  $N$  number of training sets,  $n$  number of forecast time steps by the  $s$  number of monitoring wells, given  $\mathbf{Y} \in \mathbb{R}^{N \times n \times s}$ . However, the GWN model is a multivariate model, where each node in the graph contains 2 time series : the groundwater level time series data and yearly scaled time series. The yearly scaled time series is from 0 to 1, where the first day of the year is 0 and the last day of the year is 1. Therefore,  $X$  has the tensor shape of  $\mathbf{X} \in \mathbb{R}^{N \times h \times s \times 2}$  and  $Y$  has the tensor shape of  $\mathbf{Y} \in \mathbb{R}^{N \times n \times s \times 2}$ .

The available datasets of  $\mathbf{X}$  and  $\mathbf{Y}$  for DNN model training for each data resolution can be found in Table 3.2. This value does not include the model validation or the testing datasets. All time series data are divided into segments for model training (70%), validation (15%), and testing (15%) (Figure 3.8).



**FIGURE 3.8 : Data split for DNN model training, validation, and testing are 70%, 15%, and 15% respectively.**

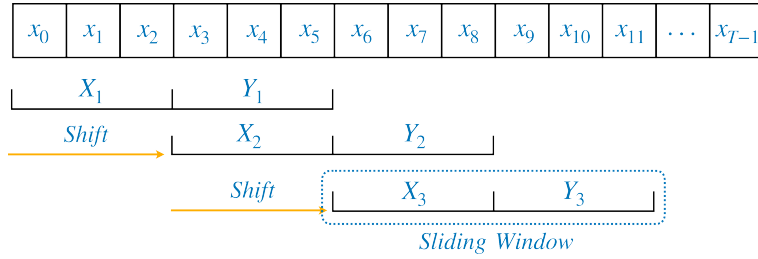
The groundwater levels of all monitoring stations are forecast at once with the 3 selected models, although only one of the stations is the spring, while the other 3 are conduit monitoring stations. We are interested in the spatio-temporal forecasting capabilities of the DNN models; therefore, we

---

developed the models to forecast all 4 stations' groundwater levels at once. The DNN models can be adapted to forecast the spring discharge only.

**TABLEAU 3.2 : Data compilation for DNN model training.**

Time per measurement	X (Time step length)	Y (Time step length)	Shift (Time step length)	Auto-Encoders (Num of training sets)	GWN (Num of training sets)
1-hour	3	3	3	5638	5638
	6	6	6	2818	2819
	12	12	12	1408	1409
4-hour	3	3	3	1408	1409
	6	6	6	703	704
	12	12	12	351	351
8-hour	3	3	3	703	704
	6	6	6	351	351
	12	12	12	175	175
12-hour	3	3	3	468	469
	6	6	6	233	234
	12	12	12	116	116
24-hour	3	3	3	234	234
	6	6	6	116	116
	12	12	12	57	57



**FIGURE 3.9 : The sequence-to-sequence data preparation for DNN model training**

For model training and optimization, we used the ADAM optimizer (Kingma et al., 2017) and the mean absolute error (MAE) loss function. Each model’s hyperparameters are optimized per dataset as hyperparameter tuning is an ad hoc exercise that is data-dependent. Node dropout and early stopping are implemented to regularize the models to prevent overfitting.

### 3.5 Results

The performance of the deep neural network (DNN) models is evaluated using four key metrics : root mean square error (RMSE), mean absolute error (MAE), mean absolute percentage error (MAPE), and the coefficient of determination ( $R^2$ ). The results for all three models are summarized in Table 3.3.

The forecast results from the testing datasets are illustrated in Figures 3.10 to 3.14. Each figure corresponds to a different temporal resolution (1-hour, 4-hour, 8-hour, 12-hour, and 24-hour) and presents forecasting results for prediction horizons of 3, 6, and 12 time steps.

Figure 3.10 displays the results for the hourly dataset. The models demonstrate strong predictive performance, achieving  $R^2$  values exceeding 0.98. All three DNN models accurately predict all forecast lengths.

Figure 3.11 presents the results for the 4-hour resampled dataset. Forecasts for 3 and 6 time step horizons yield satisfactory results across all models. However, for the 12-time-step horizon, all models exhibit lagging, where the predicted signal is shifted to the right relative to the measured signal. This suggests that the available training data may be insufficient to enable accurate long-term forecasting.

In Figure 3.12, the results for the 8-hour resampled dataset are shown. The 3 time step forecasts remain acceptable, but for the 6 time step forecasts, model predictions exhibit noticeable lag, further indicating a lack of sufficient training data. Comparatively, the 12 time step forecast for the 4-hour dataset (Figure 3.11) and the 6 time step forecast for the 8-hour dataset (Figure 3.12) are both trained with 351 datasets (Table 3.2). This suggests that achieving reliable forecasts at these data resolutions would require more than 351 training instances. Additionally, forecasts for the CP1 station (last row) indicate that the dataset may lack the necessary resolution to accurately capture

---

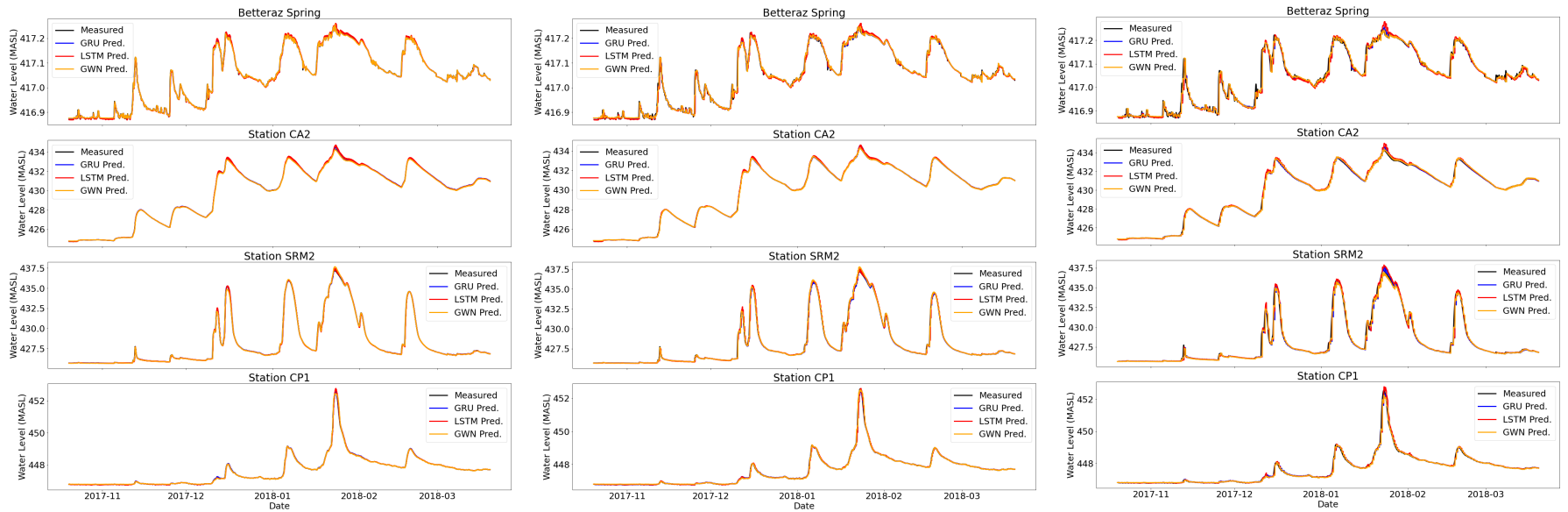
peak or extreme water levels in this dynamic aquifer system, as all models fail to predict peak water amplitudes accurately.

Figure 3.13 displays the results for the 12-hour resampled dataset. Forecasts for all models and time steps exhibit significant lag, further reinforcing that the available training data is insufficient to generate accurate predictions.

The performance metrics for all models, trained with different data resolutions and forecast lengths, are summarized in Table 3.3. Among the tested models, GWN consistently outperforms the auto-encoder based models. While the difference in performance is marginal at high data resolutions, the GWN model demonstrates a substantial advantage when working with low-resolution datasets. As expected, the best results for all models are obtained using the hourly dataset, which provides the highest resolution and the most extensive training data. The  $R^2$  values exceed 0.99 for forecasts of 3 and 6 time steps and remain above 0.98 for 12 time step forecasts. The results also indicate that forecast accuracy is preserved as long as the temporal data resolution remains high, even when training data size decreases or forecasting horizons increase. These findings highlight that while the quantity of training data is an important factor in data-driven models, temporal resolution is the key determinant of forecasting accuracy in highly dynamic karst systems such as the Betteraz spring system.

The interpolated performance metrics ( $R^2$ , RMSE, MAPE, and MAE) for all models are visualized in Figure 3.15. In these plots, scatter points indicate the locations of calculated metrics, point size represents the number of available training datasets, and color denotes metric values. By interpolating the metric values, we provide an overview of the sensitivity of these metrics in relation to data resolution, forecast length, and training data size.

### Hourly Measured Data



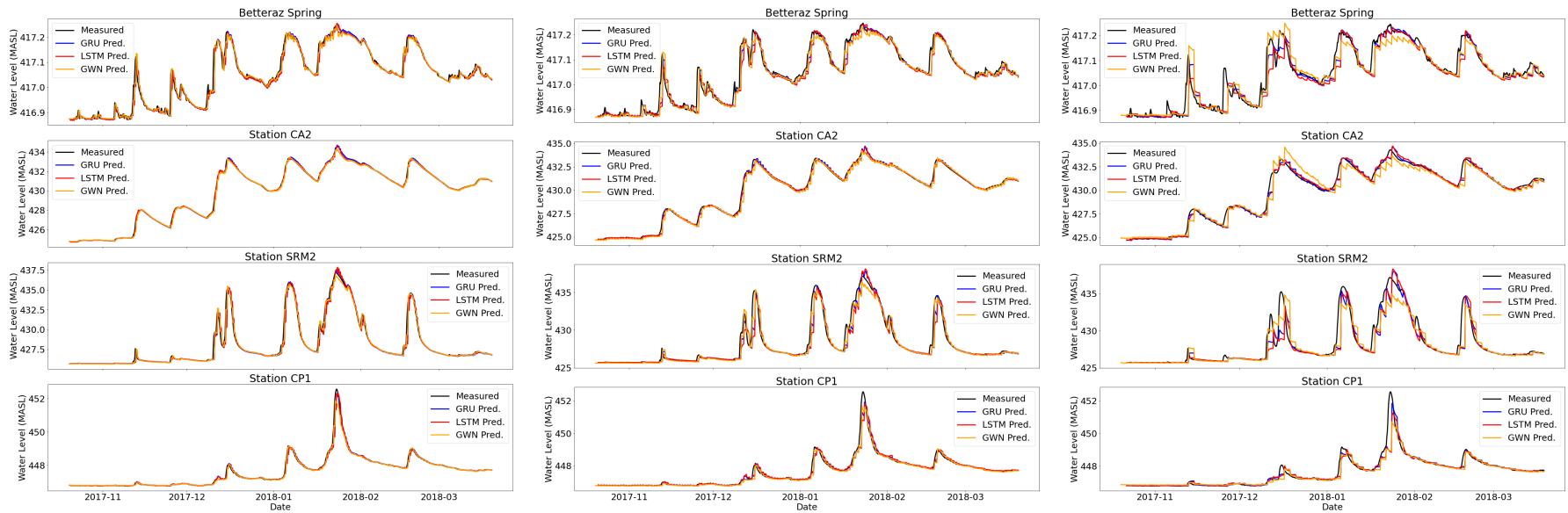
3 Time Steps Forecast

6 Time Steps Forecast

12 Time Steps Forecast

**FIGURE 3.10 : Hourly measured groundwater level data. The forecasting of 3, 6 and 12 time steps are plotted.**

### Resampled Data: 4-Hour



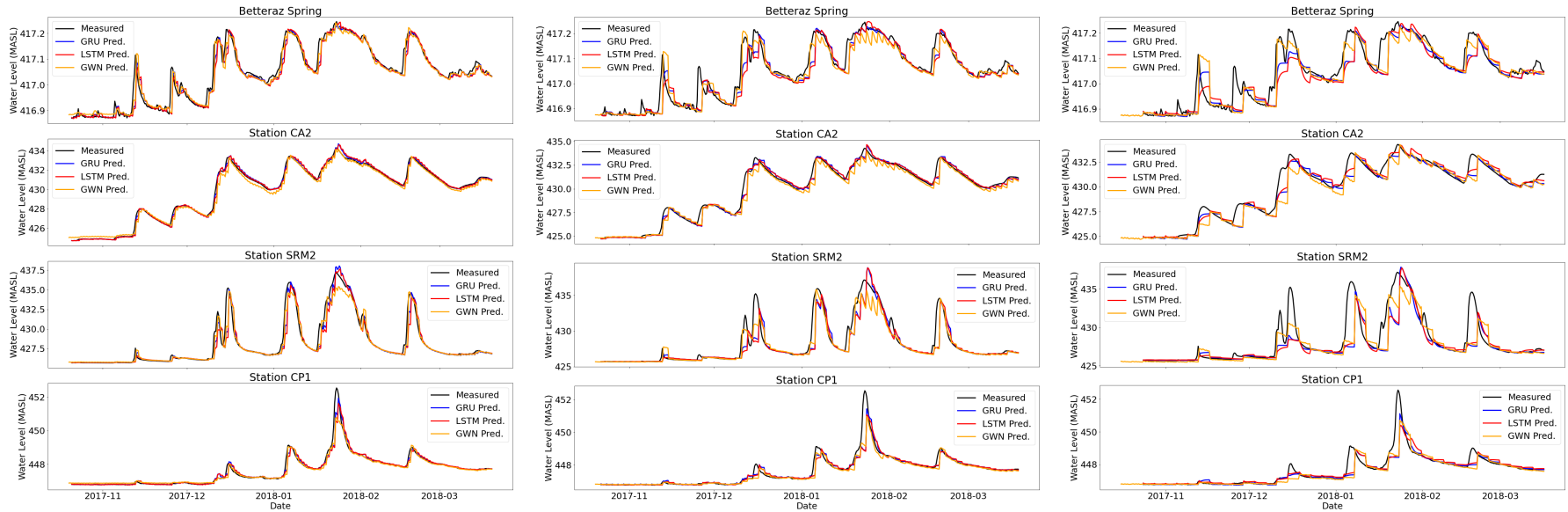
3 Time Steps Forecast

6 Time Steps Forecast

12 Time Steps Forecast

**FIGURE 3.11 : The resolution of the data is at 4-hour per time step. The forecasting of 3, 6 and 12 time steps are plotted.**

### Resampled Data: 8-Hour



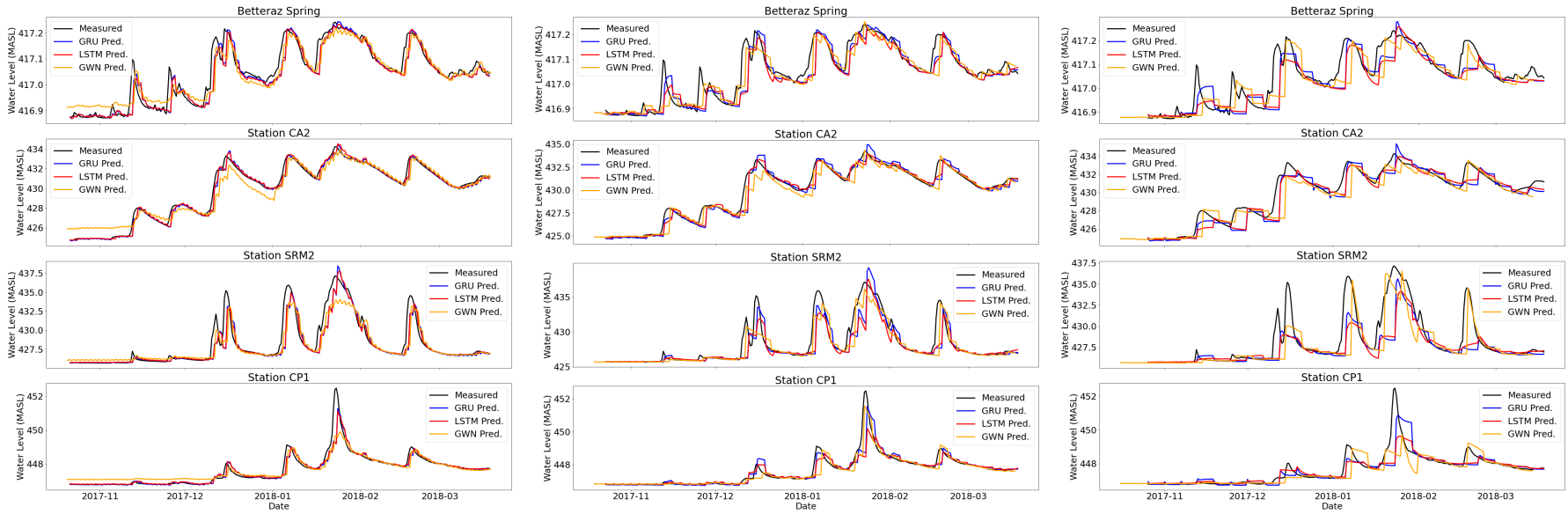
3 Time Steps Forecast

6 Time Steps Forecast

12 Time Steps Forecast

**FIGURE 3.12 : The resolution of the data is at 8-hour per time step. The forecasting lengths of 3, 6 and 12 are plotted.**

### Resampled Data: 12-Hour



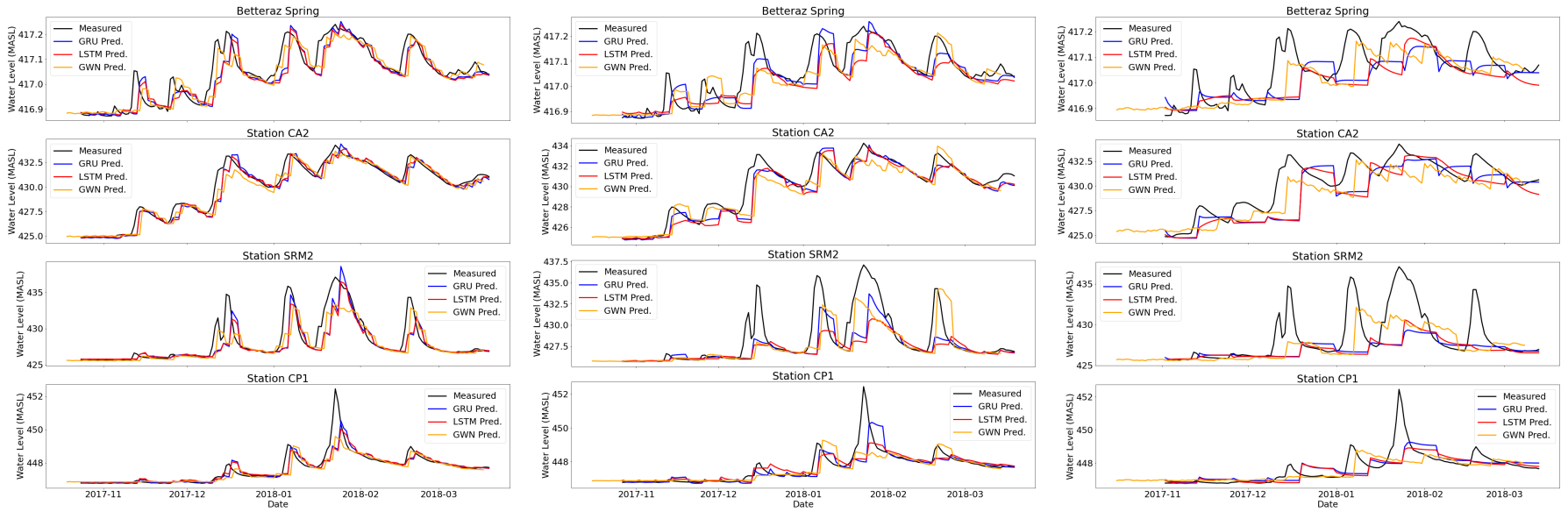
3 Time Steps Forecast

6 Time Steps Forecast

12 Time Steps Forecast

**FIGURE 3.13 : The resolution of the data is at 12-hour per time step. The forecasting lengths of 3, 6 and 12 are plotted.**

### Resampled Data: 24-hour



3 Time Steps Forecast

6 Time Steps Forecast

12 Time Steps Forecast

**FIGURE 3.14 : The resolution of the data is at 24-hour per time step. The forecasting lengths of 3, 6 and 12 are plotted.**

---

### 3.6 Conclusions and Discussions

In this study, the temporal data resolution required to obtain accurate forecasting with DNN models is tested. Furthermore, the forecasting capabilities of three DNN models are evaluated. Our findings indicate that data resolution is the most important factor in obtaining accurate forecasts. As shown in Figure 3.15, with high-resolution data, even smaller training sets enable all models to achieve good performance. Conversely, when high-resolution data is unavailable, the size of the training set becomes critical for achieving accurate results.

The three models tested include the LSTM and GRU auto-encoders, as well as the GWN. Generally, the GWN demonstrates incrementally better predictive capabilities than the auto-encoder models when using high-resolution data. However, it performs significantly better with low-resolution data and smaller training datasets. The two auto-encoders perform similarly, though the GRU auto-encoder is less computationally demanding. For instance, with the largest compiled dataset (i.e., 1-hour frequency measurements forecasting three time steps ahead) used to train the auto-encoders, the GRU model training takes 4 minutes and 28 seconds, while the LSTM requires 10 minutes and 6 seconds.

Some karst systems are highly dynamic and, therefore, require higher-resolution data to train data-driven models for more accurate forecasting results. However, in less dynamic systems, high-resolution data may not be as necessary for accurate forecasts. We recommend using a larger dataset to further test the effects of data size and resolution by systematically varying the data resolution while maintaining a constant data size. Additionally, larger training datasets are recommended to further test 12-hour and 24-hour frequency measurements, as good results might be achievable for shorter forecast lengths.

Incorporating uncertainty estimations into model predictions would also be of great interest. Models could be adapted using techniques such as Markov Chain Monte Carlo dropout (Gal et al., 2016) to provide uncertainty in predictions.

The computational requirements are not extensively considered in this study due to the use of a small dataset. However, we recorded the computation times required to train each model using the largest dataset (i.e., 1-hour frequency measurements forecasting 3 time steps ahead). The dataset contains 5,638 training sets. The required training times for the GRU auto-encoder, LSTM auto-encoder, and GWN are 4 minutes and 28 seconds, 10 minutes and 6 seconds, and 28 minutes and 22 seconds, respectively. Model training was conducted on a computer equipped with an NVIDIA GeForce GTX 1650 graphics processing unit.

As demonstrated, forecasting karst spring discharges can be reframed as a data-driven problem. The approach to data preprocessing and model selection are critical factors in achieving accurate forecasts. While some ML models are undeniably powerful and useful, they are not universally applicable.

---

## **3.7 Acknowledgments**

The funding to carry out this study is supported by Natural Resources Canada through the PAR grant. The authors would like to thank Dr. Celia Trunz (Université de Neuchâtel) for providing the data used in this study.

### **3.7.1 Authors' Note**

To the best of the authors' knowledge, there are no conflicts of interest.

### **3.7.2 Code and Data Availability**

All DNN model codes and data used in this study are available on GitHub : [https://github.com/XiaoXia10/DNN\\_forecastKarstSpring](https://github.com/XiaoXia10/DNN_forecastKarstSpring).

**TABLEAU 3.3 : Metrics calculated from the testing dataset from each of the DNN models.**

Model	Resampled data (hour)	Forecast Length	RMSE (m)	$R^2$	MAPE	MAE (m)
GRU	24	12	1.496	0.232	2.01E-03	0.904
LSTM	24	12	1.521	0.196	2.15E-03	0.933
GWN	24	12	1.236	0.512	1.80E-03	0.778
GRU	24	6	1.023	0.594	1.19E-03	0.518
LSTM	24	6	1.122	0.541	1.40E-03	0.609
GWN	24	6	1.023	0.637	1.27E-03	0.551
GRU	24	3	0.784	0.758	9.10E-04	0.395
LSTM	24	3	0.769	0.756	8.91E-04	0.386
GWN	24	3	0.791	0.772	9.75E-04	0.422
GRU	12	12	0.96	0.652	1.16E-03	0.504
LSTM	12	12	0.936	0.668	1.15E-03	0.501
GWN	12	12	0.928	0.689	1.05E-03	0.453
GRU	12	6	0.714	0.806	7.90E-04	0.346
LSTM	12	6	0.695	0.803	7.57E-04	0.328
GWN	12	6	0.625	0.862	7.10E-04	0.307
GRU	12	3	0.52	0.883	5.42E-04	0.235
LSTM	12	3	0.507	0.885	5.34E-04	0.232
GWN	12	3	0.655	0.844	9.07E-04	0.393
GRU	8	12	0.835	0.745	9.11E-04	0.395
LSTM	8	12	0.811	0.745	9.62E-04	0.417
GWN	8	12	0.823	0.77	9.70E-04	0.42
GRU	8	6	0.561	0.8681	5.81E-04	0.252
LSTM	8	6	0.573	0.857	5.96E-04	0.259
GWN	8	6	0.542	0.885	5.72E-04	0.248
GRU	8	3	0.333	0.947	3.40E-04	0.15
LSTM	8	3	0.393	0.93	4.06E-04	0.176
GWN	8	3	0.362	0.946	4.23E-04	0.183
GRU	4	12	0.528	0.88	5.23E-04	0.227
LSTM	4	12	0.545	0.869	5.49E-04	0.238
GWN	4	12	0.613	0.857	7.07E-04	0.306
GRU	4	6	0.323	0.95	3.27E-04	0.142
LSTM	4	6	0.366	0.936	3.74E-04	0.162
GWN	4	6	0.314	0.956	3.39E-04	0.147
GRU	4	3	0.205	0.979	1.99E-04	0.086
LSTM	4	3	0.22	0.975	2.07E-04	0.09
GWN	4	3	0.175	0.984	1.57E-04	0.068
GRU	1	12	0.182	0.983	1.69E-04	0.074
LSTM	1	12	0.18	0.982	1.82E-04	0.079
GWN	1	12	0.176	0.985	1.76E-04	0.076
GRU	1	6	0.102	0.994	9.43E-05	0.041
LSTM	1	6	0.115	0.992	1.14E-04	0.05
GWN	1	6	0.0873	0.996	7.88E-05	0.034
GRU	1	3	0.065	0.997	6.98E-05	0.03
LSTM	1	3	0.076	0.997	7.52E-05	0.033
GWN	1	3	0.056	0.998	5.82E-05	0.025

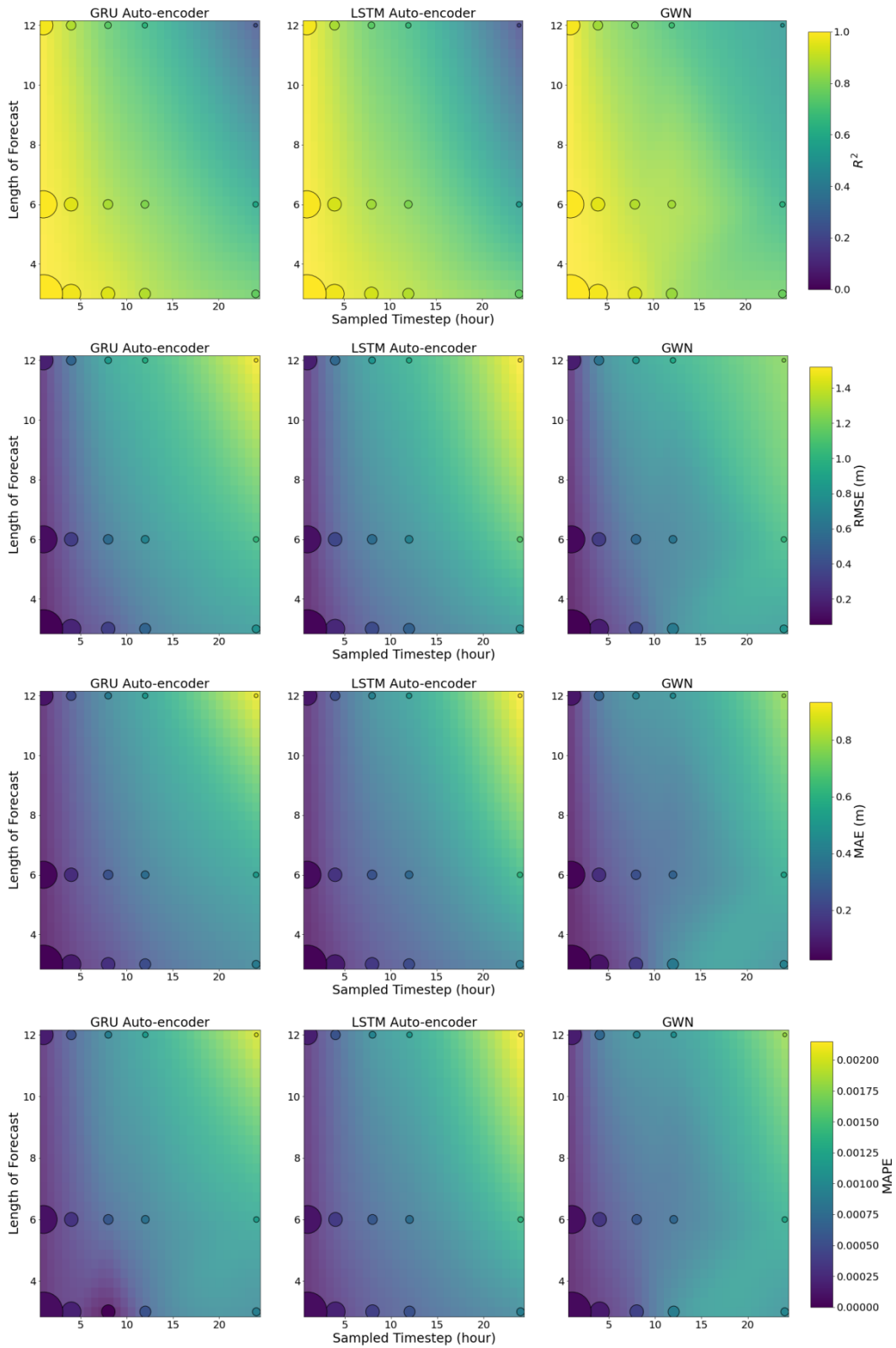


FIGURE 3.15 : The interpolation of the  $R^2$ , RMSE, MAE and MAPE metric results of all models. The scatter points are the sampled locations and the size of the points are the data size available for model training.

## **4 ARTICLE III : DEEP NEURAL NETWORK LOSS FUNCTION DEVELOPMENT FOR FORECASTING EXTREME EVENTS IN TIME SERIES DATA**

---

**Développement d'une fonction de perte pour réseaux neuronaux profonds visant la prévision des événements extrêmes dans les séries temporelles**

Xiao Xia Liang<sup>1</sup>, Julien Straubhaar<sup>2</sup>, Erwan Gloaguen<sup>1</sup>, Maxime Claprood<sup>3</sup>, and Philippe Renard<sup>2</sup>

<sup>1</sup> Centre Eau Terre Environnement, Institut National de la Recherche Scientifique, Québec, Canada

<sup>2</sup> Centre d'Hydrogéologie et de Géothermie, Université de Neuchâtel, Neuchâtel, Suisse

<sup>3</sup> Département des sciences appliquées, Université du Québec à Chicoutimi, Chicoutimi, Canada

### **Contributions des auteurs**

Xiao Xia Liang : Rédaction du manuscrit, méthodologie, programmation, nettoyage et compilation des données et conceptualisation.

Julien Straubhaar : Simulations de Statistiques à points multiples, programmation et révision du manuscrit.

Erwan Gloaguen : Supervision, méthodologie et révision du manuscrit.

Maxime Claprood : Supervision et révision du manuscrit.

Philippe Renard : Supervision, méthodologie et révision du manuscrit.

### **Publication ciblée**

Journal : Journal of Hydrology - Soumis

Date de soumission : mars 2025

### **Objectif de l'article**

Dans le deuxième article, nous avons déterminé que les modèles d'AA, tels que les modèles RNG, peuvent être utilisés pour prévoir avec précision les niveaux des eaux souterraines à partir de données spatio-temporelles mesurées sur le terrain. L'objectif de cet article est de développer une nouvelle fonction de perte afin d'améliorer les prévisions du modèle pour les événements extrêmes. Le principal objectif est d'étendre et de tester davantage les capacités de prévision des modèles d'AA de l'Article 2 pour prédire les événements extrêmes grâce à notre nouvelle fonction de perte.

---

**Résumé** Les eaux souterraines constituent une ressource essentielle pour la consommation humaine et la croissance économique. Développer des outils permettant de prévoir les niveaux des eaux souterraines avec une plus grande précision est crucial pour une gestion durable de cette ressource. Avec le développement rapide des méthodes d'apprentissage profond (AP) basées sur les données, les hydrogéologues ont tiré parti de cette tendance en utilisant des modèles AP pour la gestion prédictive des ressources en eaux souterraines. Dans cette étude, nous avons développé une nouvelle fonction de perte, appelée Extreme Loss, destinée aux problèmes de régression et pouvant être couplée à n'importe quel modèle AP afin d'améliorer la prévision des événements extrêmes dans les séries temporelles. Nous avons utilisé le jeu de données sur le débit des sources karstiques du système karstique de Milandre en Suisse ainsi qu'un jeu de données de surveillance des eaux souterraines du bassin versant de la Yamaska au Canada, afin d'évaluer les performances de cette fonction. Ces deux ensembles de données, très différents, ont été exploités pour tester la capacité de la fonction de perte extrême (Extreme Loss Function; ELF) à améliorer la prédiction des événements extrêmes, tels que les inondations et les sécheresses, par les modèles DL. Nous avons employé une méthode d'échantillonnage direct basée sur les statistiques multi-points (Multiple Point Statistics, MPS) pour combler les lacunes dans les séries temporelles segmentées. Cette méthode génère des réalisations statistiquement plausibles des données manquantes en intégrant les données sur les eaux souterraines ainsi que d'autres variables temporelles complémentaires, telles que les précipitations, la température et les tendances saisonnières. Les séries temporelles complétées par la méthode MPS ont ensuite été utilisées pour tester l'ELF en l'intégrant à un modèle auto-encodeur Gated Recurrent Unit (GRU) ainsi qu'au modèle Graph WaveNet (GWN). Ces deux modèles ont été entraînés en utilisant notre ELF ainsi que la fonction de perte erreur quadratique moyenne (EQM) pour comparaison. Chaque modèle a effectué des prévisions sur deux horizons temporels différents afin d'évaluer l'efficacité de notre ELF. Nos résultats montrent que l'ELF améliore significativement les prévisions du modèle GWN pour les événements hors distribution. Pour le modèle auto-encodeur GRU, de légères améliorations ont été observées. Ces résultats indiquent que la fonction de perte extrême remplit son objectif, à savoir améliorer la prévision des événements extrêmes en contraignant l'apprentissage du modèle lors de l'entraînement.

**Abstract** Groundwater is a critical resource for human consumption and economic growth. Developing tools that can forecast groundwater levels more accurately is imperative for sustainable groundwater management. With the recent rapid development of data-driven, deep learning (DL) methods, hydrologists have leveraged this trend by employing DL models for predictive groundwater resource management. In this study, we developed a novel loss function, called extreme loss, for regression problems, which can be coupled with any DL model to enhance the prediction of extreme events in time series data. We used the karst spring discharge dataset from the Milandre Karst system in Switzerland and a monitoring well dataset from the Yamaska watershed in Canada to evaluate its performance. These two very different datasets were used to test the capability of the extreme loss function (ELF) in improving the DL model's prediction of extreme events, such

---

as floods and droughts. We employed a direct sampling Multiple Point Statistics (MPS) method to fill gaps in the segmented time series datasets. This MPS method generates statistically plausible realizations of the missing data by incorporating groundwater data and other complementary time series inputs, such as precipitation, temperature, and seasonal trends. The completed time series generated by the MPS method are used to test the ELF by coupling it with a Gated Recurrent Unit (GRU) auto-encoder model and the Graph WaveNet (GWN) model. Both models were trained using our ELF and the Mean Absolute Error (MAE) loss function for comparison. Each model conducted forecasts with two different lengths to further evaluate the effectiveness of our ELF. Our results demonstrate that the ELF significantly improves the GWN model's forecasts for out-of-distribution events. For the GRU auto-encoder model, slight improvements were observed. These findings indicate that the extreme loss function performs as intended, which is to improve predictions of extreme events by constraining the model during training.

**Keywords :** Extreme Event Forecasting, Time Series Data, Karst Spring Forecast, Groundwater Forecast, Loss Function, Machine Learning, Deep Neural Network

## 4.1 Introduction

Extreme climate events, such as floods and droughts, are complex phenomena driven by multivariate physical and biological processes. These events often have significant impacts on the economy, ecology, and sustainable development of a region. With climate change accelerating, extreme climate events are becoming increasingly prevalent worldwide (Dai, 2013; Li et al., 2024; Cook et al., 2018; Jasechko et al., 2024). Changing precipitation patterns and rising temperatures are placing greater pressure on aquifers. In response to these challenges, we aim to develop simple yet effective tools that we can use in the case of sustainable groundwater management. This study focuses on developing a loss function, extreme loss function (ELF), that can be coupled to any DL model for solving regression-type problems to improve the forecasting performance of the models on extreme events. Here, we define extreme or outlier events as measured values that fall outside the mean range, typically more than two standard deviations from the mean. In terms of hydrology, measured events such as high flow and low flow rates, which are out of the mean data distribution, are considered to be extreme events.

Many data-driven methods, such as neural networks (Clark et al., 2022; Solgi et al., 2021; Adamowski et al., 2011; Daliakopoulos et al., 2005; Bowes et al., 2019; Patra et al., 2023; Wunsch et al., 2021), time series analysis (Vonk et al., 2024; Brakenhoff et al., 2022; Sakizadeh et al., 2019), and geostatistics (Ma et al., 2024b; Takafuji et al., 2019; Yoon et al., 2016), have been employed to forecast groundwater levels. While many researchers have attempted to use data-driven methods for groundwater level forecasting, few have focused specifically on improving the accuracy of forecasting extreme events.

---

Using deep learning (DL) or deep neural network (DNN) models, Sahoo et al. (2019) tested a Long Short-Term Memory (LSTM) network to evaluate its suitability for forecasting low flow in discharge data collected from a gauging station in India. They found that the LSTM model is suitable for low-flow forecasting when predicting one time step ahead. However, no efforts were made to improve the forecast specifically during low-flow periods. On the other hand, Li et al. (2023) did probabilistically enhance three different LSTM models trained in parallel to better predict reservoir water levels during extreme periods of drought in California, USA.

Similarly, others have designed DNN models to detect or predict outlier events in time series data. (Naidoo et al., 2022) employed an LSTM auto-encoder coupled with a 3-layer fully connected feed-forward network to improve the prediction of anomaly outlier events in stock market data. (Kieu et al., 2018) developed a DNN framework to detect outliers in multivariate time series data. This framework involves two steps : first, enriching the time series data with derived features and statistical features to capture temporal changes, thereby creating enriched time series data; and second, feeding these enriched time series data into two auto-encoder models to reconstruct the data. The original enriched data are then used as indicators of outlier events.

The existing literature on outlier detection and prediction often involves complex frameworks with multiple DNN models. In this study, we do not rely on complex models to forecast extreme events. Instead, we focus on developing a loss function that is simple to implement and independent of specific models or frameworks. In regression problems, the loss function quantifies the discrepancy between the model's predictions and the target values. The loss function's primary objective is to minimize this error, thereby improving the accuracy of the model. During model training, the process focuses on reducing the loss to optimize performance. Most regression problems utilize mean-based loss functions, such as the Mean Absolute Error (MAE) and the Mean Squared Error (MSE) (Wang et al., 2020b; Jadon et al., 2024). These mean-based loss functions penalize predictions that deviate from the mean, resulting in smoothed predictive outputs. However, applying mean loss functions in DNN models for time series forecasting with extreme events can be ineffective, as they hinder the model's ability to predict values outside the mean range. Here, we propose to test two different DNN models, the Gated Recurrent Unit (GRU) auto-encoder (Cho et al., 2014) and the Graph WaveNet (GWN) model (Wu et al., 2019), coupled with our ELF to forecast extreme events in two different time series datasets. One of the datasets used in this study consists of measured karst spring flow from Switzerland, and the other is measured well groundwater levels from Canada. A comparison is done with the MAE loss function using these two models forecasting on the two datasets.

---

## 4.2 Study Areas

In this section, we briefly describe the hydrogeological context of both datasets used to test our approach, located in the Milandre Karst system in Switzerland and the Yamaska watershed in Canada.

### 4.2.1 Milandre Karst System

The Milandre catchment is located in the northwestern part of Switzerland, in the Jura Mountains bordering France (Figure 4.1). This catchment hosts a karst system of significant interest to speleologists, as 12 km of passages have been explored since the 1960s. The cave passages are predominantly subhorizontal and organized into several levels (Vuilleumier et al., 2019). Subhorizontal Jurassic limestone is underlain by Oxfordian marls (Kovács et al., 2003). Since the Oxfordian marls act as an aquitard beneath the active conduits, there is no deep phreatic zone. This aquifer system is classified as a shallow karst system (Perrin, 2003).

The karst system has a recharge area of  $13.3 \text{ km}^2$ , which includes a perennial spring, Saivu, and an overflow spring, Bâme (Favre, 2001). A flow loss from the Milandre catchment to the Font spring catchment has been identified, though the exact paths of this flow loss remain unknown. There is also a perennial stream flowing through the main karst drain, often referred to as the Milandrine (Grasso et al., 1994). The mapped conduits of the Milandre karst system are depicted in Figure 4.1 in blue.

### 4.2.2 Yamaska Watershed

The Yamaska watershed is located on the south shore of the Saint Lawrence River, east of Montréal, Canada (Figure 4.2). The Yamaska River flows northwest from its headwaters in Brome Lake, covering a length of more than 160 km (Remmal et al., 2017). This watershed encompasses an area of  $4,794 \text{ km}^2$ , consisting mainly of flat farmlands known as the Saint Lawrence Lowlands. To the southeast of the Lowlands lies the Piedmont region, and the farthest southeastern corner corresponds to the Appalachian Uplands (Beaudry et al., 2018).

The Lowlands are underlain by the sedimentary Saint Lawrence Platform, which consists of Cambrian and Ordovician black and red shales, limestones, and dolostones. This platform is covered by till and fluvio-glacial sediments deposited during the last two glacial periods. These glacial sediments are, in turn, overlain by marine clay and silt deposited by the Champlain Sea (Beaudry et al., 2018; Globensky, 1987). The fractured bedrock aquifer in the Lowlands region is confined by marine clay that is thicker than 10 meters and contains brackish water.

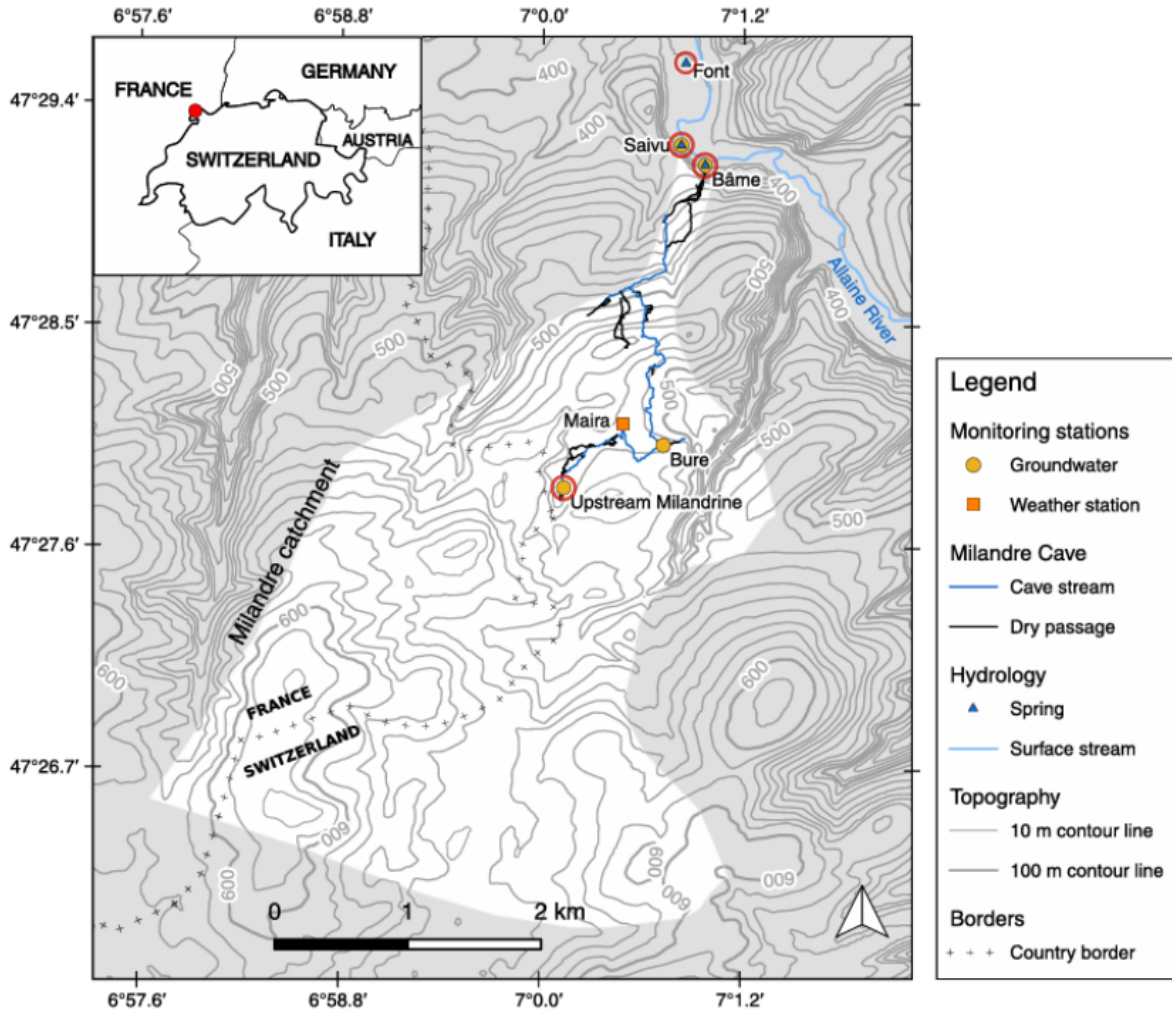


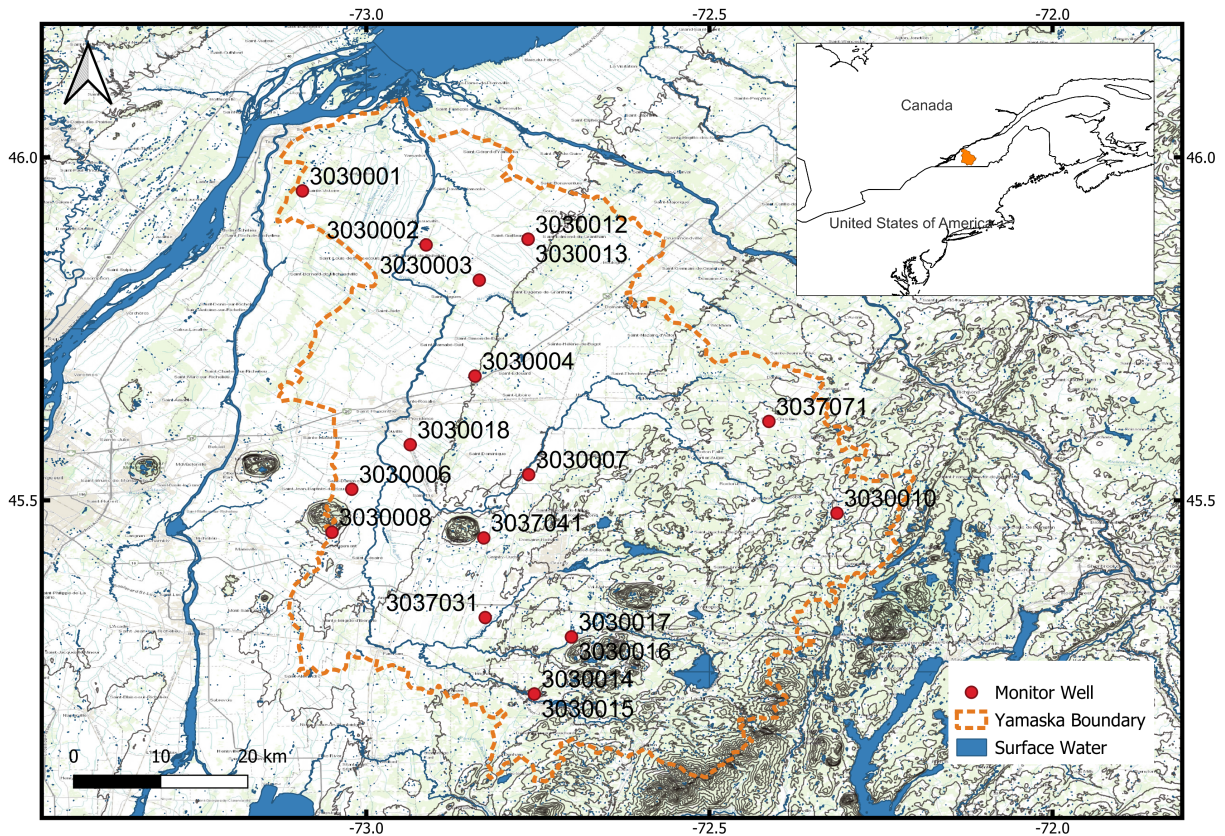
FIGURE 4.1 : Milandre catchment with locations of the monitoring and spring measuring stations. Data used in this study is from stations (Milandrine, Bâme, Saivu, and Font) encircled in red. This map is modified from Vuilleumier et al. (2021)

Moving south toward the Piedmont region, the thickness of marine clay and till varies, resulting in differing levels of aquifer confinement. In the Uplands, which correspond to the Humber Zone of the Appalachians, the region is composed of highly deformed Devonian continental margin rocks. The aquifer in this region is generally unconfined, with varying sediment thicknesses and discontinuous till layers deposited during the last glacial period (Beaudry et al., 2018).

### 4.3 Time Series Data

#### 4.3.1 Milandre karst system data

The time series of the Swiss karst system used in this study comes from three springs named Font, Bâme, and Saivu, as well as an upstream station in the perennial Milandrine stream, referred



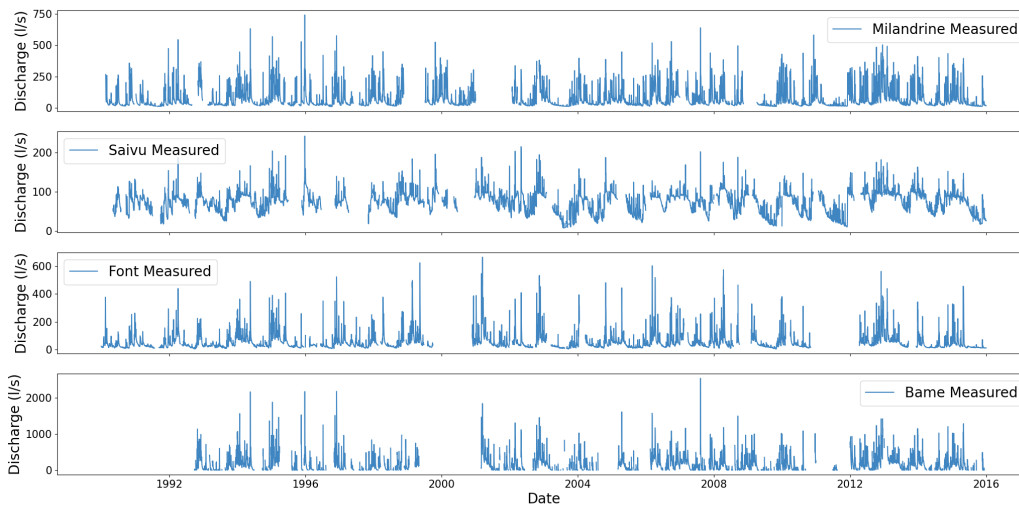
**FIGURE 4.2 : The Yamaska watershed is located on the south shore of the Saint-Lawrence river. The locations and labels of the monitoring wells are shown.**

to as Milandrine. In total, fragmented data from four monitoring stations are available, covering the period from the beginning of 1990 to the end of 2015 (Figure 4.3). The measurement frequency is hourly, but the data have been resampled to a 4-hour time step to achieve a lower data resolution, allowing the effects of the extreme loss function on the forecasting capabilities of the selected models to be tested.

The data is divided into two parts : from 1990 to 2014 for model training and validation, and the year 2015 for testing. This dataset was chosen due to the dynamic flow behavior of the springs. Using this dataset, the study aims to test the capability of the selected models, coupled with the extreme loss function, to forecast high and low spring flow rates.

### 4.3.2 Yamaska Monitoring Well Data

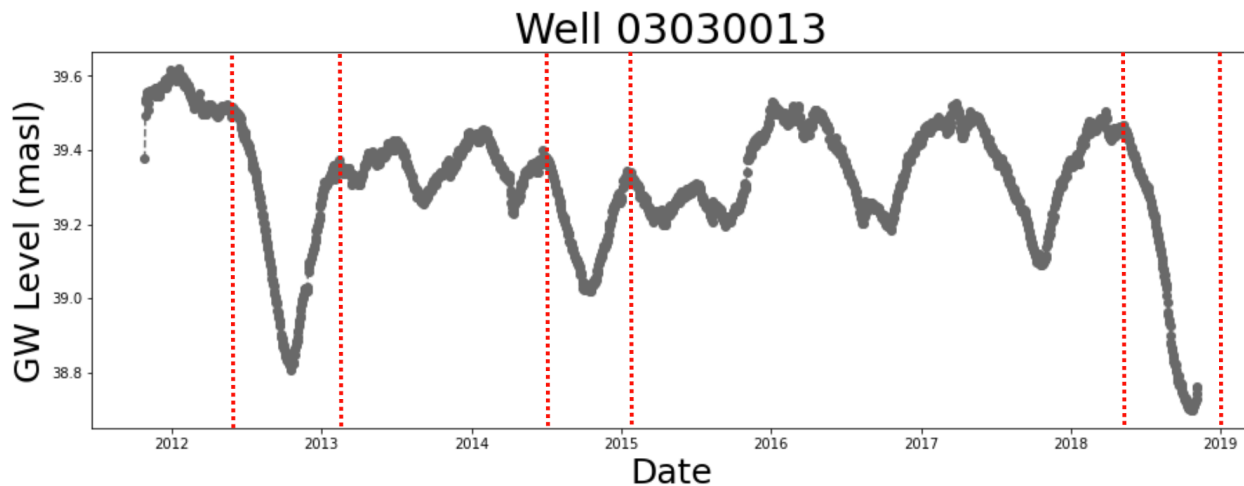
This dataset contains measured piezometric groundwater levels from 18 monitoring wells throughout the Yamaska watershed (Figure 4.2). The data sampling frequency is daily, covering the period from January 2011 to December 2018. However, the dataset contains missing measure-



**FIGURE 4.3 : Measured discharge from stations Milandrine, Saivu, Font and Bâme.**

ments in most of the time series. The data is divided into two parts : from 2011 to 2017 for model training and validation, and the year 2018 for testing.

This dataset includes three distinct periods of measured groundwater table decline which have a potential impact on the intense agricultural activity in the area. The first measured decline occurred from the middle of 2012 to the beginning of 2013, the second from the middle of 2014 to early 2015, and the third from early 2018 to the end of 2018 (Figure 4.4). This dataset was selected due to its recorded groundwater declines, making it suitable for testing the capability of the extreme loss function in aiding low groundwater table forecasting.



**FIGURE 4.4 : Measured decline in the Yamaska groundwater table. The 3 decline periods are bounded by red dotted lines.**

---

### 4.3.3 MPS Time Series Gap Filling

Incomplete data sets are common in geosciences, as seen in the karst spring dataset (Figure 4.5). Many data imputation methods exist in DL; common methods to handle missing data are to fill the missing data with zeros or the mean of the time series. A more elaborate method is to mask the data with ones in place of data and zeros in place of missing data; this mask is then used in the DL model training. This masking method indicates where there is data for the DL model to carry out training or data imputation. For example, Che et al. (2018) uses RNN models with a data mask to impute missing multivariate time series data.

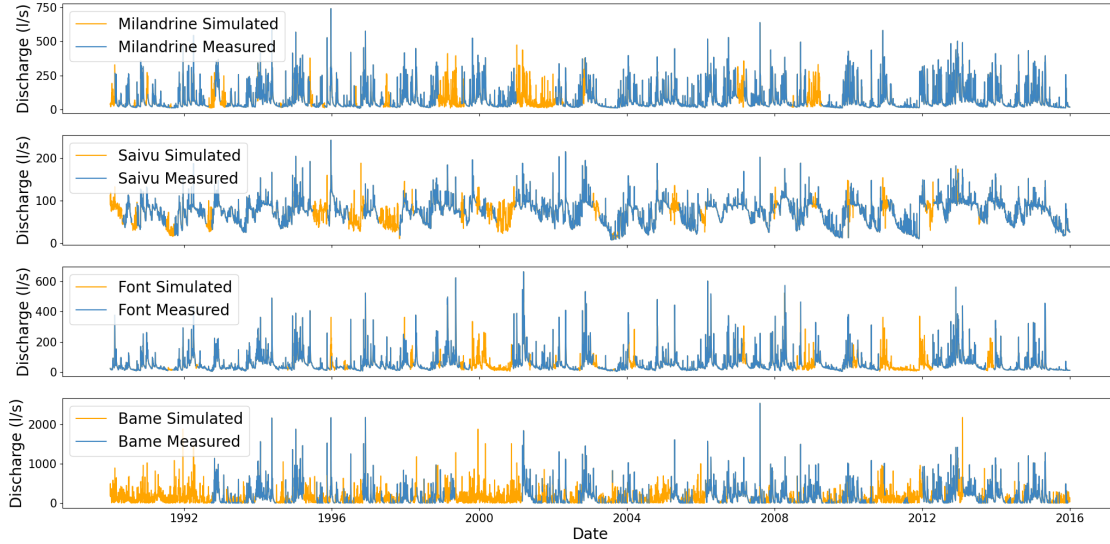
Here, instead of using a DL model for data imputation, we use a proven algorithm called DeeSse (Mariethoz et al., 2010, 2015; Straubhaar et al., 2021) that uses direct sampling Multiple Point Statistics (MPS) to generate stochastic realizations of the missing data. This is a non-parametric method that randomly populates the missing time series by borrowing patterns from the training dataset. To generate more equiprobable realizations, we also used auxiliary temporal data (precipitation, temperature, and seasonal trend) (Oriani et al., 2014). All realizations generated comply with the measured data statistics, making these realizations stochastically probable. The time series data completed by the MPS method is used for model training, validation, and testing. The readers are referred to the Appendix section for more information on MPS simulations. As an illustration, one realization of each of the stations is plotted in Figure 4.5 along with the measured data.

Stochastically probable realizations of the missing data in the Milandre and Yamaska datasets are generated using this MPS method. One generated realization from each of the datasets is randomly selected for DL model training and testing.

## 4.4 Extreme Loss Function

The ELF developed in this study is applicable to regression-type problems aimed at improving a DL model's performance on predicting extreme or outlier events. Our loss function leverages the simplicity of the MAE loss function while applying additional weights to the errors associated with outlier events. The extreme loss function employs a hyperparameter,  $\alpha$ , which is multiplied by the standard deviation of the training data to set a threshold for identifying outlier values. When outliers are detected, the function creates two masks to isolate them, one for minimum values and the other for maximum values of the data distribution. Extra weights are applied to these two masks in the loss function. The ELF calculated loss is then fed back into the DL model for backpropagation. The ELF method can be divided into 4 main steps :

Step 1. The mean and standard deviation of an input subset of target training data (mini-batch of input target data),  $y_{true}$ , are calculated.



**FIGURE 4.5 : Milandre measured and simulated missing data. The simulated missing time series data is generated from MPS simulations.**

Step 2. The minimum and maximum range of the target training data that is outside of the mean is calculated. The minimum range is computed as  $\gamma = \text{abs}(\text{mean} - \text{min})/\text{std}$ , while the maximum is computed as  $\beta = \text{abs}(\text{mean} - \text{max})/\text{std}$ .

Step 3. Masks are created by checking the conditions : if  $\text{abs}(y_{\text{true}} - \text{mean}) > \alpha * \text{std}$  and if  $\text{abs}(y_{\text{true}} - \text{mean}) < \alpha * \text{std}$ , where  $\alpha$  is a hyperparameter that is greater than or equal to 2, and  $\text{std}$  is the standard deviation. The former condition is referred to as the *extreme\_mask\_max*, and the latter condition is referred to as the *extreme\_mask\_min*.

Step 4. Using the masks from step 3 and the min and max ranges calculated from step 2, we add weights to the calculated MEA loss from the true target data,  $y_{\text{true}}$  and DL model predicted,  $y_{\text{pred}}$ . This is done in two parts by separating the min extremes and max extremes. We add weight to the MEA loss where the *extreme\_mask\_max* is true by multiplying by the scalar  $\beta$ . Separately, we add weight to the MEA loss where the *extreme\_mask\_min* is true by multiplying by  $\gamma$ . The 2 weighted MEA losses are merged together to get the extreme loss.

Since the extreme loss function is based on the absolute error of model predictions, it behaves as a standard MAE loss function when no extreme events are present. By using training subsets instead of the entire training time series, the function mitigates potential non-stationarity issues in the data. Additionally, by employing two masks, it allows the method to bypass challenges associated with non-Gaussian distribution in the data.

The extreme loss function constrains DL models to find a more optimal minimum through the calculated loss but does not alter the inherent predictive capacity of the models. If a DL model is already well-suited for forecasting outliers or extreme events, the ELF does not interfere with this

capability. However, when the model struggles with predicting extremes, the ELF constrains the loss to enhance the accuracy. These aspects will be demonstrated through the results obtained using the two selected DL models in this study.

## 4.5 Deep Learning Models

Two DL models, GWN (Wu et al., 2019) and GRU auto-encoder (Cho et al., 2014), are selected to test the extreme loss function in regards to its architecture. The GWN is a graph neural network that integrates a stacked temporal convolutional network (TCN) with a graph convolutional network (GCN) to make forecasting predictions (Wu et al., 2019; Shleifer et al., 2019). Meanwhile, the GRU auto-encoder is a recurrent neural network (RNN) structured with an encoder and a decoder. The encoder model, as the name suggests, encodes the learned input information into a fixed-length vector, which is then fed to the decoder for decoding (Cho et al., 2014). The code for both models can be found on GitHub (Section ).

### 4.5.1 GWN Model

The GWN model adapted for this study is the improved version described in Shleifer et al. (2019). All formulations for this model can be found in Wu et al. (2019) and Shleifer et al. (2019).

A graph neural network, such as the GWN, requires the data to be represented as a graph. A graph consists of nodes and edges. In this context, the nodes represent the monitoring stations/wells, and the edges represent the hydraulic connectivity between these stations/wells. An adjacency matrix serves as the mathematical representation of a graph. Since the two datasets used in this study are fundamentally different, distinct methods are applied to construct the graphs for network training.

For the Milandre karst spring data, a weighted and directed graph was constructed using the latitude, longitude, and elevation parameters of the monitoring stations (Figure 4.6). The graph is directed as we assumed that the karst water flow is unidirectional in the karst conduits. The graph was constructed by adapting the Gaussian covariance function (Li et al., 2017),

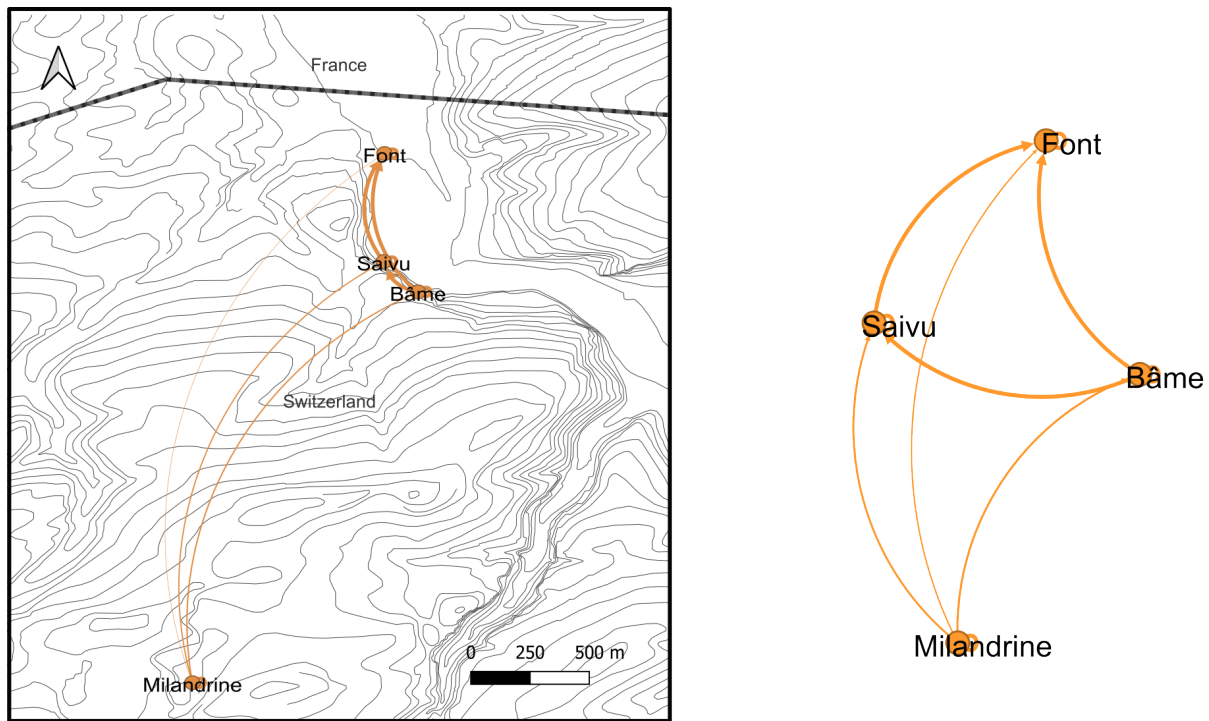
$$A_{i,j} = \exp\left(-\left(\frac{\text{dist}(\mathbf{v}_i, \mathbf{v}_j)}{\epsilon}\right)^2\right) \cdot D_{i,j}, \quad (4.1)$$

where  $A$  is the adjacency matrix,  $i$  and  $j$  denote the rows and columns within the matrix,  $\text{dist}(\cdot, \cdot)$  is the distance function between two nodes accounting for the latitude, the longitude, and the elevation,  $\epsilon$  is a scaling factor, represented by the standard deviation of the distances, and  $D$  is the directional matrix, calculated based on the elevation differences between stations : the entry at row index  $i$  and column index  $j$  is 1 if the elevation of the station at node  $\mathbf{v}_i$  is greater than or equal to the elevation of the station at node  $\mathbf{v}_j$ , and 0 otherwise.

For the Yamaska groundwater monitoring wells, a weighted, undirected graph was constructed by calculating the Gaussian covariance distances between the monitoring wells as the flow direction is not calculated a priori. The parameters used include latitude, longitude, elevation, depth of well opening, aquifer confinement type, and aquifer sediment type (Figure 4.7). Using these six parameters, the graph was constructed as follows,

$$A_{i,j} = \exp \left( - \left( \frac{\text{dist}(\mathbf{v}_i, \mathbf{v}_j)}{\epsilon} \right)^2 \right), \quad (4.2)$$

where  $\mathbf{v}$  is a vector containing the 6 parameters listed above.



**FIGURE 4.6 : Weighted and directed graph of the Milandre monitoring stations. The map on the left shows the physical locations of the stations with their spatial connections calculated using Equation 4.1. The right illustration shows the weighted and directed graph of the stations projected in data space. The edge thickness represents the weighted spatial connections of the stations (Bastian et al., 2009b).**

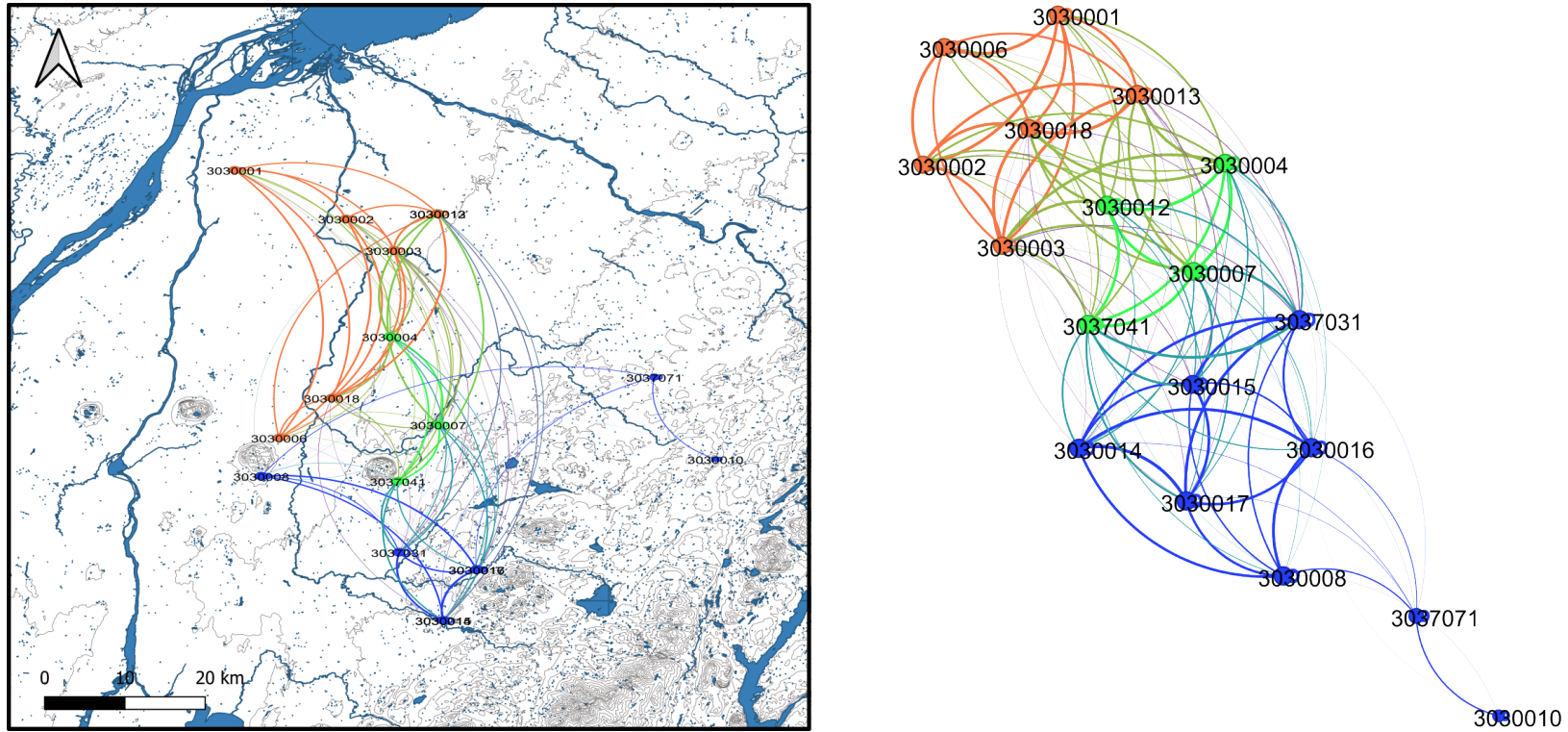


FIGURE 4.7 : The left map shows the Yamaska monitoring well graph fixed in physical location. The right graph shows the Yamaska monitoring wells projected in data space. The graph is calculated using Equation 4.2. The color of node indicates the nodes to be in the same neighborhood as they are the closet neighbors calculated using the 6 parameters. The edge thickness represents the weights of spatial connections between the wells. The size of the nodes represents the degree of connections (Bastian et al., 2009b).

---

## 4.5.2 GRU Auto-Encoder Model

The GRU auto-encoder, introduced by Cho et al. (2014), was originally developed for statistical machine translation. It consists of a recurrent neural network (RNN) encoder and decoder that are jointly trained to maximize the conditional probability of an output sequence given an input sequence. The encoder processes the input sequence and compresses its information into a fixed-length vector, which represents the probability distribution of the sequence. The decoder then uses this vector to generate probabilistic sequential predictions. Following these principles, we developed a simple GRU auto-encoder model; refer to Cho et al. (2014) for the detailed formulation of this auto-encoder.

## 4.5.3 Experimental Setup

The sequence-to-sequence method is used for model training and forecasting; this method is also known as the sliding window method. This method involves training a model on a data sequence to forecast another sequence. Here, the time series data is compiled into  $\mathbf{X}$  and  $\mathbf{Y}$  pair, where  $\mathbf{X}$  is the training sequence and  $\mathbf{Y}$  is the target sequence. The sequence  $\mathbf{X}$  can be written as  $[x_0, x_{h-1}], [x_{shift}, x_{shift+h-1}], [x_{2shift+h-1}], \dots, [x_{T-h-n}, x_{T-n-1}]$ , and the sequence  $\mathbf{Y}$  as  $[x_h, x_{h+n-1}], [x_{shift+h}, x_{shift+h+n-1}], [x_{2shift+h}, x_{2shift+h+n-1}], \dots, [x_{T-n}, x_{T-1}]$ . Here,  $x$  is the measured data,  $h$  is the training sequence length,  $n$  is the target sequence length,  $shift$  is the shifting length to get the next  $\mathbf{X}$  and  $\mathbf{Y}$  pair, and  $T$  is the length of the time series data.

Denoting  $s$  the number of monitoring stations or wells (i.e. the dimension of any  $x_k$ ), and  $N$  the number of training sets (i.e. the number of sequences within  $\mathbf{X}$  and  $\mathbf{Y}$ ), the sequences are rearranged in array form as  $\mathbf{X} \in \mathbb{R}^{N \times h \times s}$  and  $\mathbf{Y} \in \mathbb{R}^{N \times n \times s}$ . This is true for the auto-encoder model as the data tensors are in 3 dimensions, but the GWN model is a multivariate model that requires 4-dimensional data tensors. The GWN model requires that each node in the graph contains a minimum of 2 time series. In this case, the nodes contain the measured time series data and the yearly scaled time series. The yearly scaled time series ranges from 0 to 1, where the first day of the year is 0 and the last day of the year is 1. Therefore, for the GWN model,  $\mathbf{X} \in \mathbb{R}^{N \times h \times s \times 2}$  and  $\mathbf{Y} \in \mathbb{R}^{N \times n \times s \times 2}$ .

With the two datasets, 2 different forecasting time steps are tested on both DL models (i.e., 2 different  $h$  and  $n$  lengths are tested on each model). All the data compilation information for model training can be found in table 4.1.

The tunings of hyperparameters of each model are done per dataset, as this is an ad hoc exercise that is data dependent. Once the hyperparameters are tuned, they are kept the same for model training with the ELF and the MAE loss function to obtain a fair comparison of each of the loss

functions. Early stopping for model training and node dropout are implemented to regulate the models to prevent overfitting.

**TABLEAU 4.1 : Data compilation for DL model training**

Dataset	Hours per time step	Training time steps ( $h$ )	Forecast time steps ( $n$ )
Milandre	4	6	6
	4	9	9
Yamaska	24	3	3
	24	4	4

## 4.6 Results

The metrics used to evaluate the models' performances are the coefficient of determination ( $R^2$ ), the Root Mean Square Error (RMSE), and the Mean Absolute Error (MAE). The GWN test dataset metrics can be found in Table 4.2, and the GRU auto-encoder test dataset metrics can be found in Table 4.3. All plotted results shown in this section are the test data from both datasets.

### 4.6.1 GWN Model Results

The test data metrics for the GWN model trained with Milandre data show improvements using the extreme loss function compared to using the MAE loss function. In both forecast lengths, clear improvements can be seen in all three metrics. The results for the GWN model trained with the Yamaska data also show small improvements in all three metrics when extreme loss is used (Table 4.2).

The test data forecasts for Milandre can be found in Figure 4.8 for the 6 forecast lengths, while the cross plots of measured data and prediction for the MAE loss and the extreme loss results are plotted in Figure 4.9. Differences in the forecast of extreme highs and lows are observed in Figure 4.8, showing that the extreme loss function is working to help improve the forecast of outlier or extreme events. The results for the 9 forecast lengths (Figure 4.10) and its cross plots (Figure 4.11) show similar results to the 6 forecast length results. However, the improvements are smaller since the forecast length has increased.

The results of the Yamaska 3-day ahead forecast are plotted in Figure 4.12 and the corresponding cross plot is Figure 4.13. The 4-day ahead forecasts are plotted in Figure 4.14 and the corresponding cross plot is Figure 4.15. The metrics results from using the extreme loss for both forecast lengths are slightly better than using the MAE loss. In Figure 4.12 and Figure 4.14 the extreme loss can be seen working as intended. The results of using the extreme loss are more evident in wells 3030012 and 3030013 as the recorded low groundwater levels between September and November are better predicted. This is also clearly visible for low values in the cross plots (Figures 4.13

and 4.15). In all the cases, the extreme values predicted using ELF, both low and high, are less biased compared to MAE.

**TABLEAU 4.2 : GWN test dataset metrics**

Dataset	Forecast length	GWN - MAE Loss			GWN - EXT Loss		
		$R^2$	RMSE	MAE	$R^2$	RMSE	MAE
Milandre	6 (4-hour)	0.73	54.20 l/s	21.37 l/s	0.78	49.37 l/s	17.63 l/s
	9 (4-hour)	0.66	61.14 l/s	23.44 l/s	0.69	59.03 l/s	21.60 l/s
Yamaska	3 (24-hour)	0.95	0.057 m	0.033 m	0.95	0.055 m	0.031 m
	4 (24-hour)	0.95	0.062 m	0.038 m	0.95	0.060 m	0.035 m

#### 4.6.2 GRU Model Results

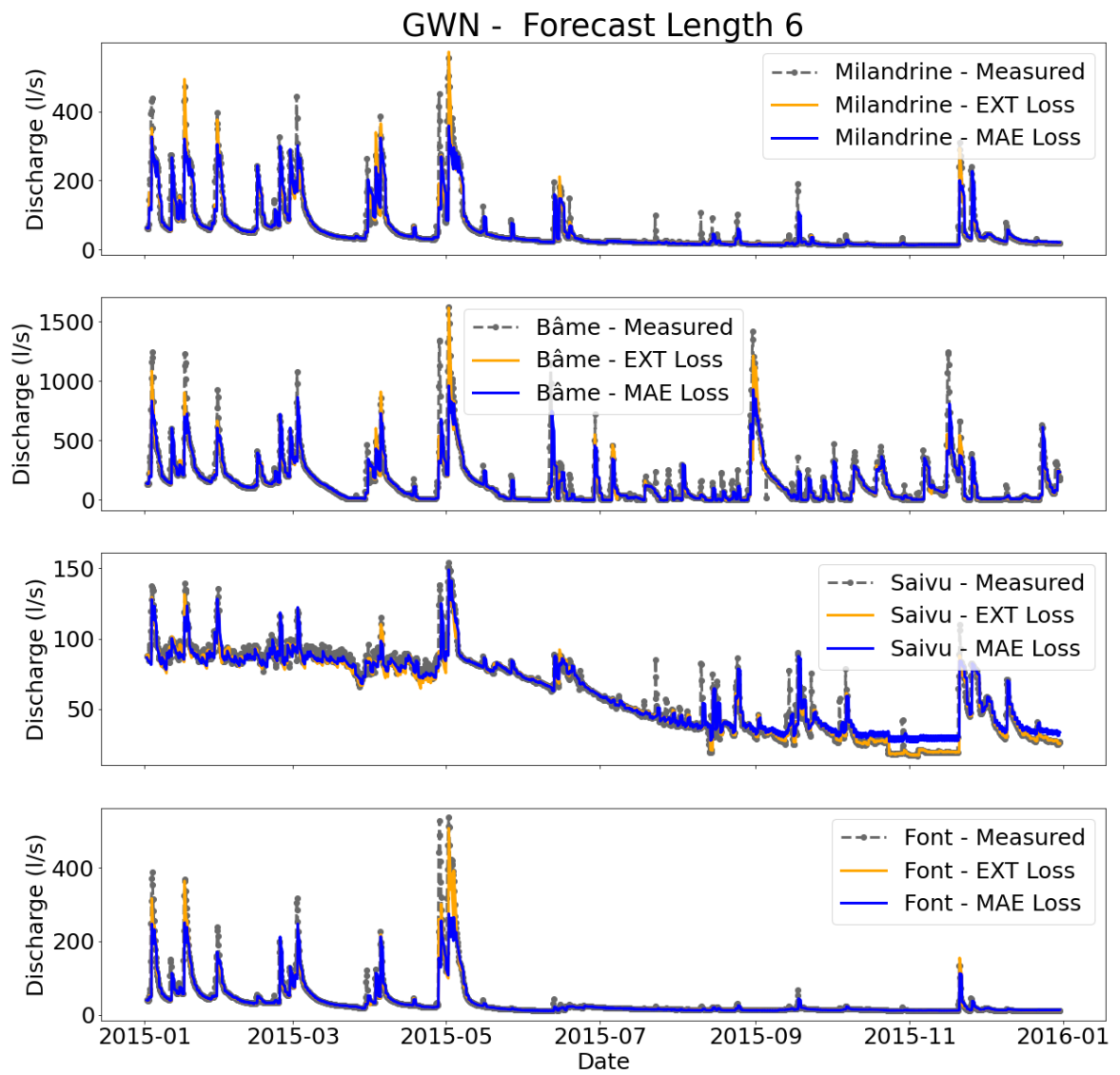
All test data metrics' results from the GRU model can be found in Table 4.3. For the Yamaska data, slight improvements are observed in all metrics with the extreme loss function. As for the Milandre data, improvements are observed in the 6 forecast length in the  $R^2$  and RMSE metrics, while the MAE metric is marginally better with the MAE loss function. For the Milandre 9 forecast length metrics, the  $R^2$  is the same for both loss functions, while the RMSE and MSE metrics are moderately better with the MAE loss function. These metrics are very close in value and do not have any significant differences between the 2 loss functions.

The results trained on the Milandre data are plotted in Figure 4.16 for 6 forecast lengths, and the corresponding cross plot is Figure 4.17. The results for the 9 forecast lengths are plotted in Figure 4.18 and the corresponding cross plot is Figure 4.19. The plots of using the extreme loss and the MAE loss are not evidently different. However, small differences can be seen in the high flows between the extreme loss and the MAE loss.

The results trained on the Yamaska data for the 3 days ahead forecast are plotted in Figure 4.20 and the corresponding cross plot is Figure 4.21. The results of the forecast for the 4 days ahead are plotted in Figure 4.22 and the corresponding cross plot is Figure 4.23. The plots exhibit very little difference between the extreme loss and the MAE loss. The results using the MAE loss are good; therefore, using the extreme loss will not largely improve the results.

**TABLEAU 4.3 : GRU test dataset metrics**

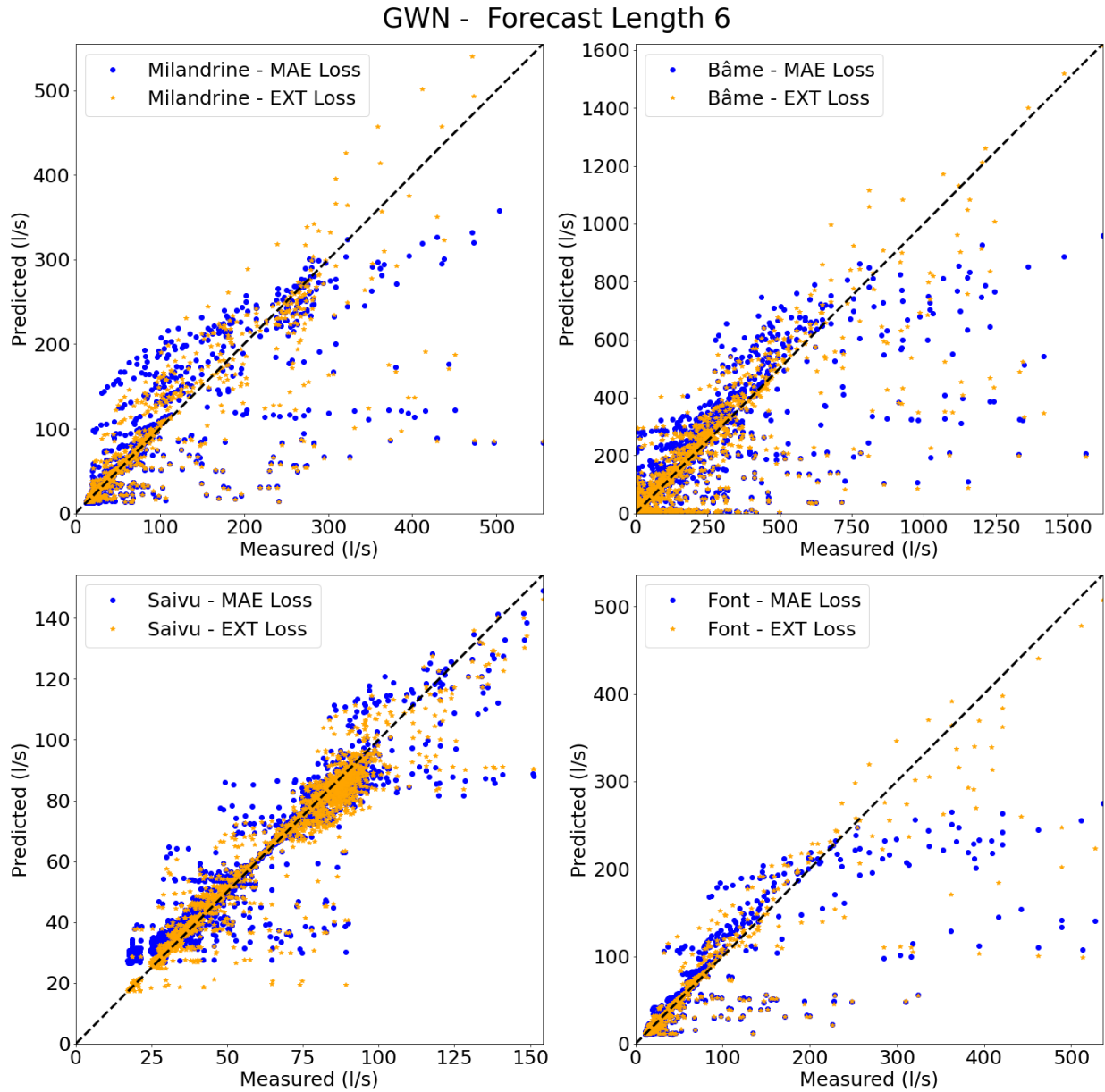
Dataset	Forecast length	GRU - MAE Loss			GRU - EXT Loss		
		$R^2$	RMSE	MAE	$R^2$	RMSE	MAE
Milandre	6 (4-hour)	0.77	50.25 l/s	18.14 l/s	0.78	49.19 l/s	19.23 l/s
	9 (4-hour)	0.68	59.82 l/s	21.33 l/s	0.68	60.00 l/s	22.72 l/s
Yamaska	3 (24-hour)	0.93	0.073 m	0.052 m	0.94	0.070 m	0.050 m
	4 (24-hour)	0.92	0.079 m	0.057 m	0.92	0.077 m	0.055 m



**FIGURE 4.8 : GWN model forecast results (Milandre) - 6 time step forecast lengths.**

### 4.6.3 High Resolution Dataset

The ELF's performance depends on the predictive capability of the model used. Other factors, such as hyperparameter tuning and data resolution, are also crucial for achieving accurate forecasts. To demonstrate the importance of data resolution, we trained the two models using the Milandre data sampled at 1-hour intervals to forecast 6, 9, and 12 time steps ahead (i.e., forecasting 6, 9, and 12



**FIGURE 4.9 : Cross plot of measured and GWN predictions (Milandre) of 6 time step forecast lengths.**

hours ahead, as each time step is 1 hour). The metric results shown in Table 4.4 are significantly better compared to the results from the 4-hour per time step sampled data for both the GWN (Table 4.2) and GRU auto-encoder models (Table 4.3). In this case, the extreme loss improves the metrics for a 6 time step ahead forecast but provides no improvements for longer forecast lengths. This finding highlights the importance of data resolution and model selection in time series forecasting. The results in Table 4.4 suggest that the use of the ELF may not be necessary when high resolution data is available, but would still be beneficial to use as small improvements can still be found.

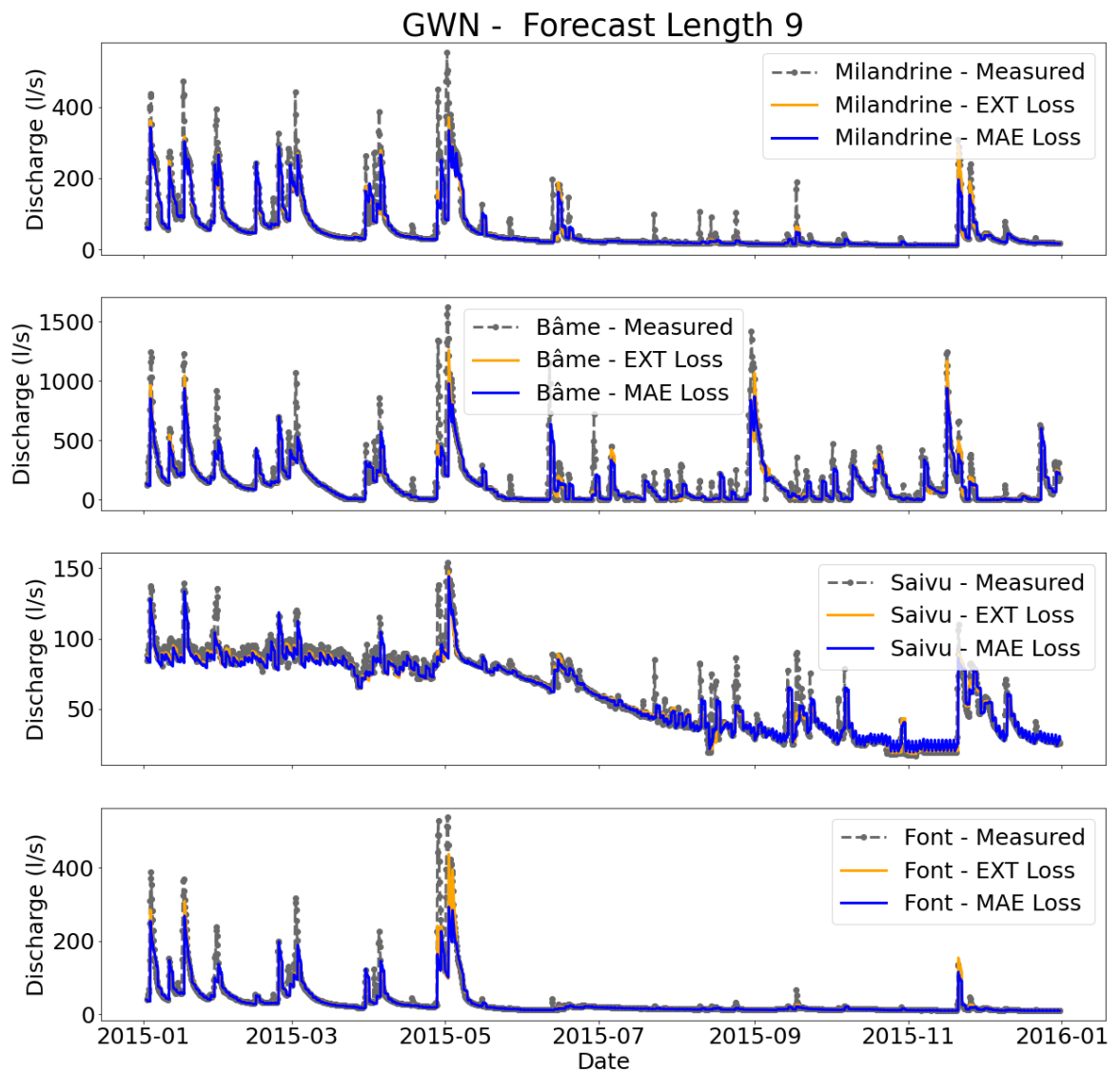
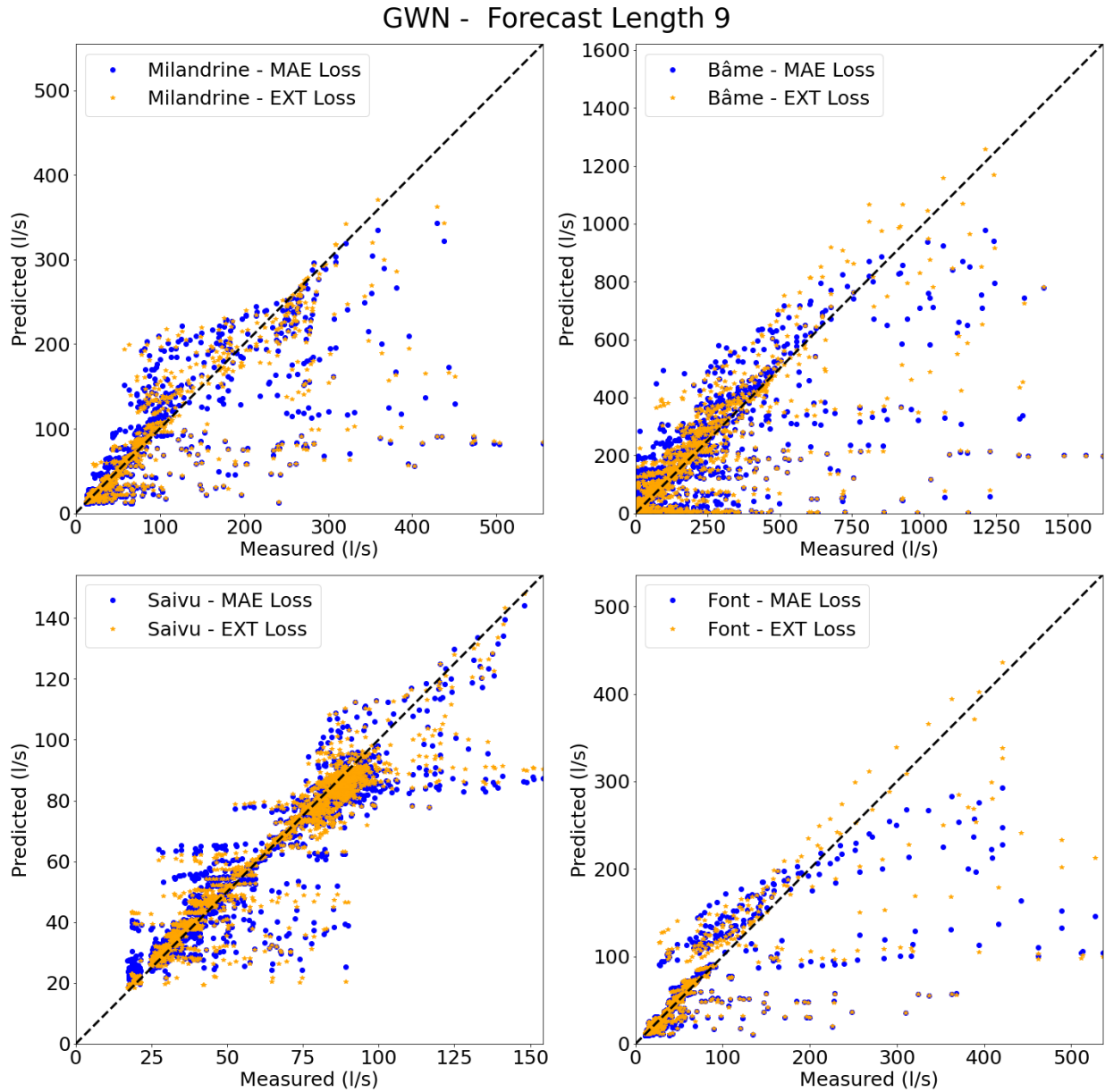


FIGURE 4.10 : GWN model forecast results (Milandre) - 9 time step forecast lengths.

## 4.7 Conclusions and Discussions

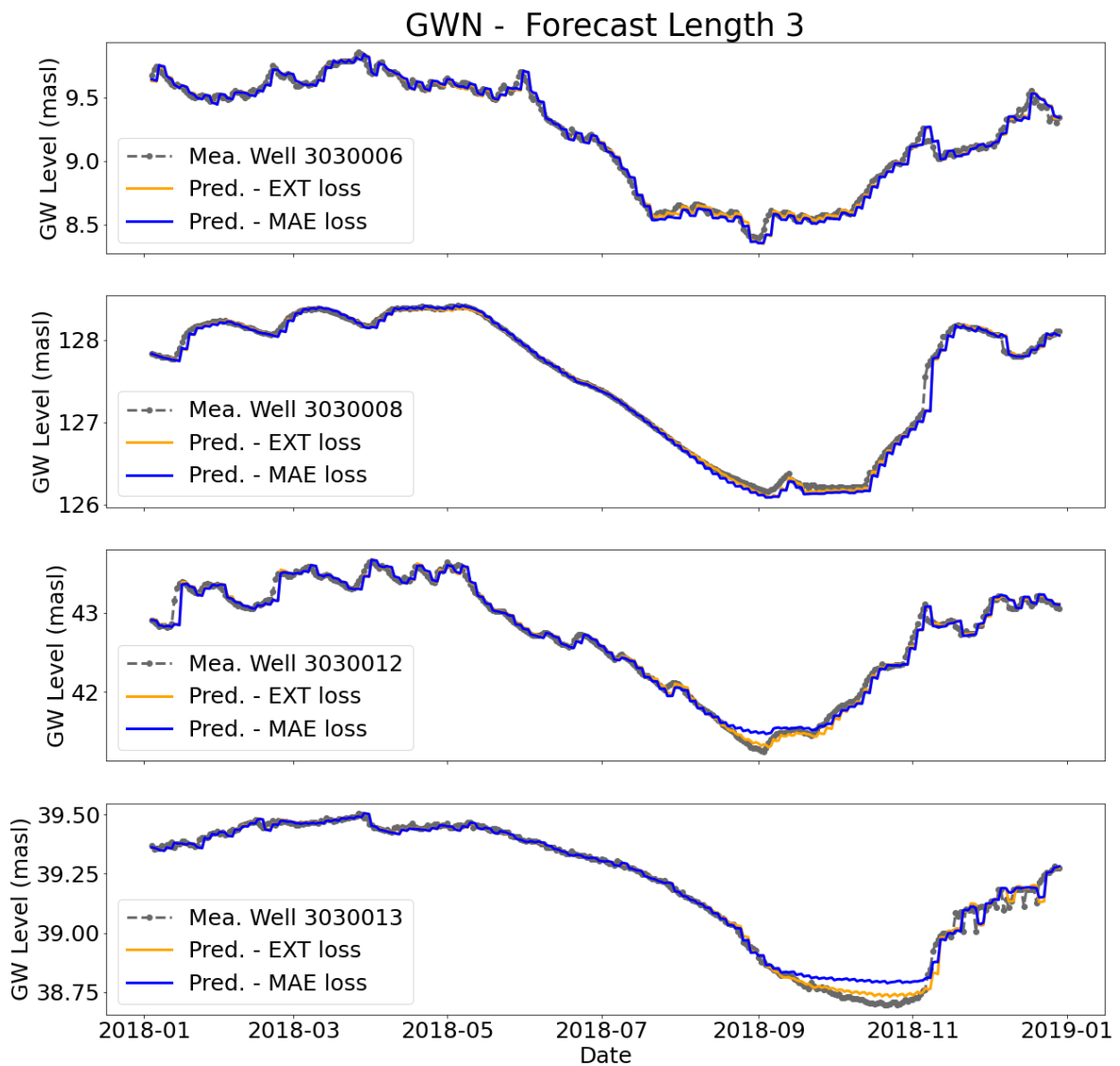
The extreme loss function is performing as intended by improving the forecasts of extreme events. With the Milandre data, the extreme loss helps the GWN model predict high discharge rates more accurately and also aids in improving the low discharge predictions observed at the Saivu station.



**FIGURE 4.11 : Cross plot of measured and GWN predictions (Milandre) of 9 time step forecast lengths.**

For the Yamaska data, the extreme loss function constrains the GWN model predictions, enabling better forecasts of low groundwater levels.

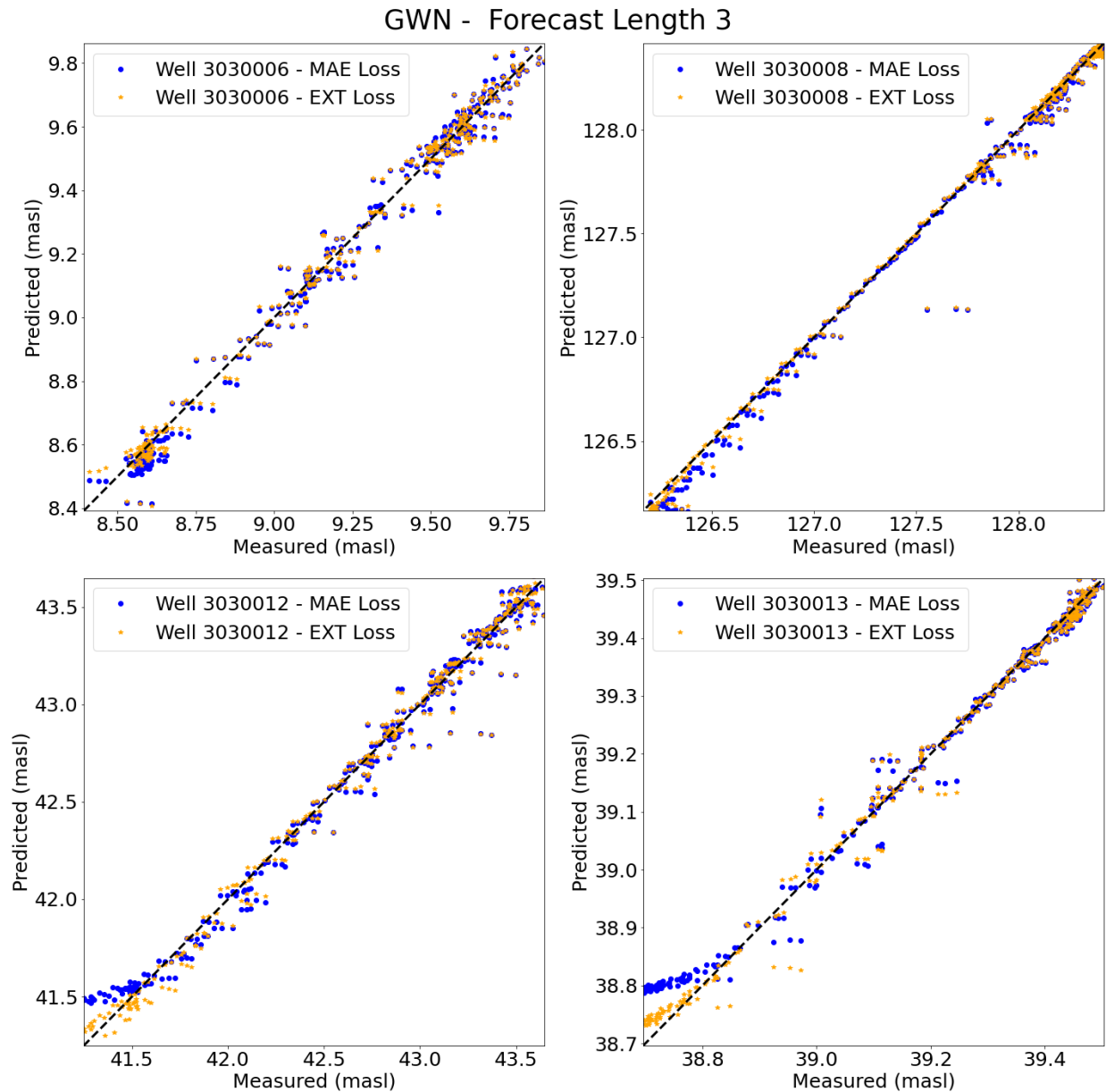
No significant improvements are observed when using the extreme loss function with the GRU auto-encoder model trained on the Yamaska data. This result is unsurprising, as auto-encoder models are inherently well-suited for predicting extreme events (Kieu et al., 2018; Naidoo et al., 2022). However, with the more dynamic Milandre data, slight improvements in forecasting are noted. It is worth mentioning that the GRU auto-encoder model appears to require more data for training compared to the GWN model. For the given size of the Yamaska dataset, the GRU auto-encoder's forecasting capability is limited to 4 days ahead.



**FIGURE 4.12 : GWN model forecast results (Yamaska) - 3 time step forecast lengths.**

The improvements in metrics calculated from the test datasets using the extreme loss function, compared to the MAE loss function, are small. However, these metrics do not fully represent the improvements in forecasting extreme events, as they are calculated over the entire test dataset rather than being isolated for extreme events.

As discussed in the Introduction section, when a DL model is well-suited to forecast extreme events, such as the GRU auto-encoder model used in this study, the extreme loss function is

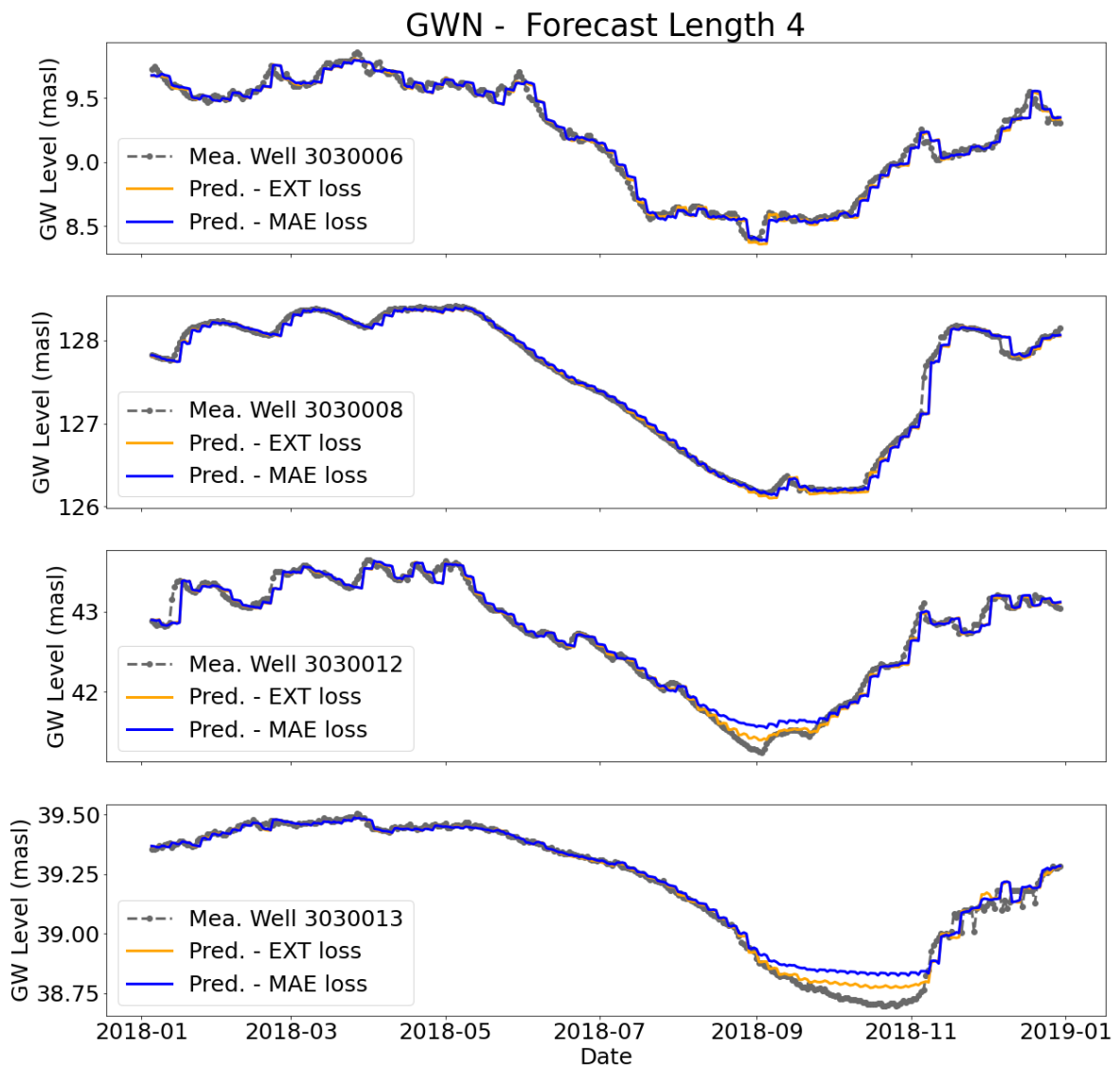


**FIGURE 4.13 : Cross plot of measured and GWN predictions (Yamaska) of 3 time step forecast lengths.**

less effective. This distinction is observed in the differing effectiveness of the extreme loss function on the GWN and GRU auto-encoder predictions. While the effects are clearly visible in the GWN model predictions, they are not as apparent in the GRU auto-encoder predictions.

### **Declaration of Competing Interest**

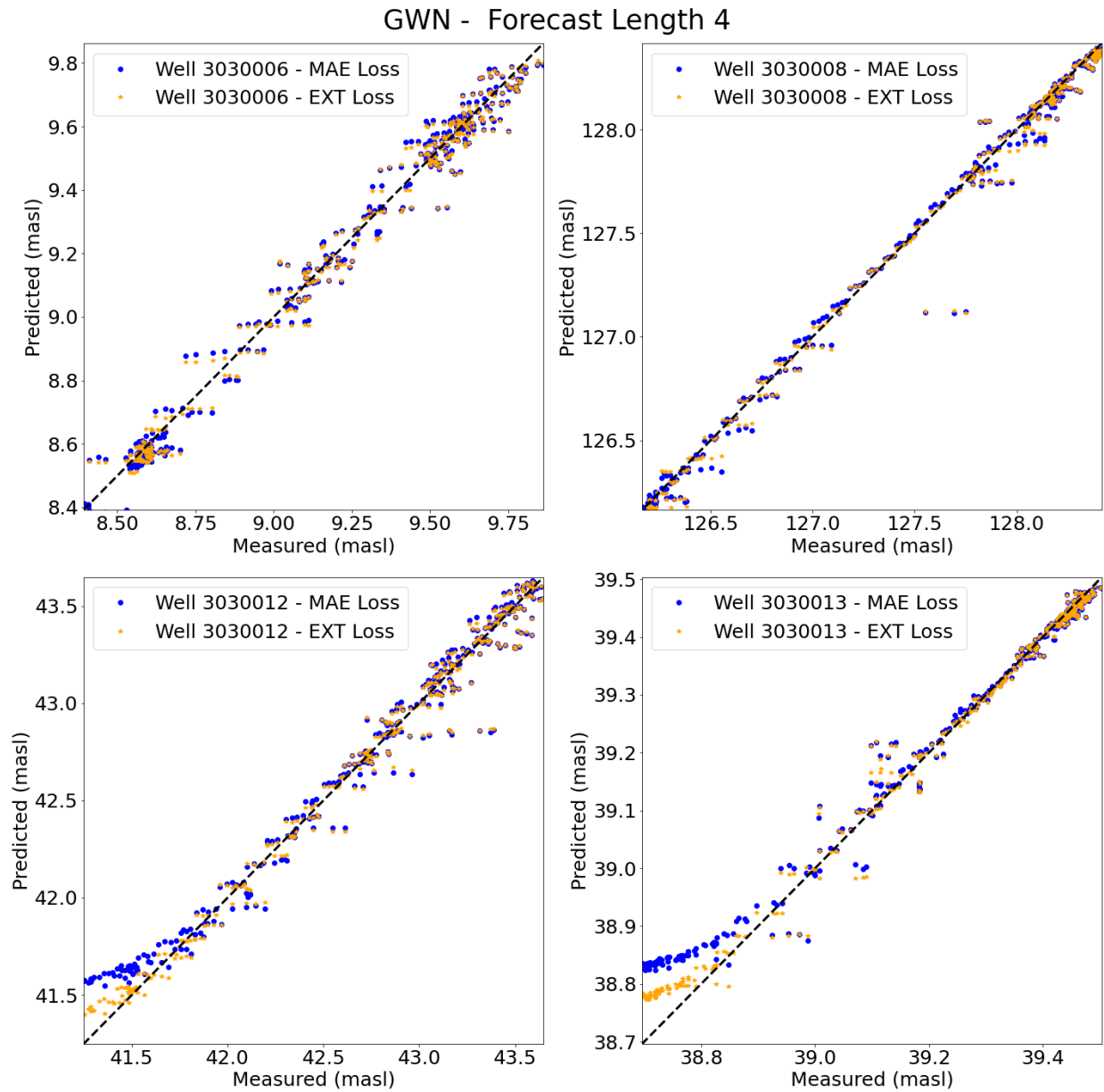
The authors have no known competing financial interests to declare, or have any personal relationships that would have influenced the work presented in this paper.



**FIGURE 4.14 : GWN model forecast results (Yamaska) - 4 time step forecast lengths.**

### Data And Code availability

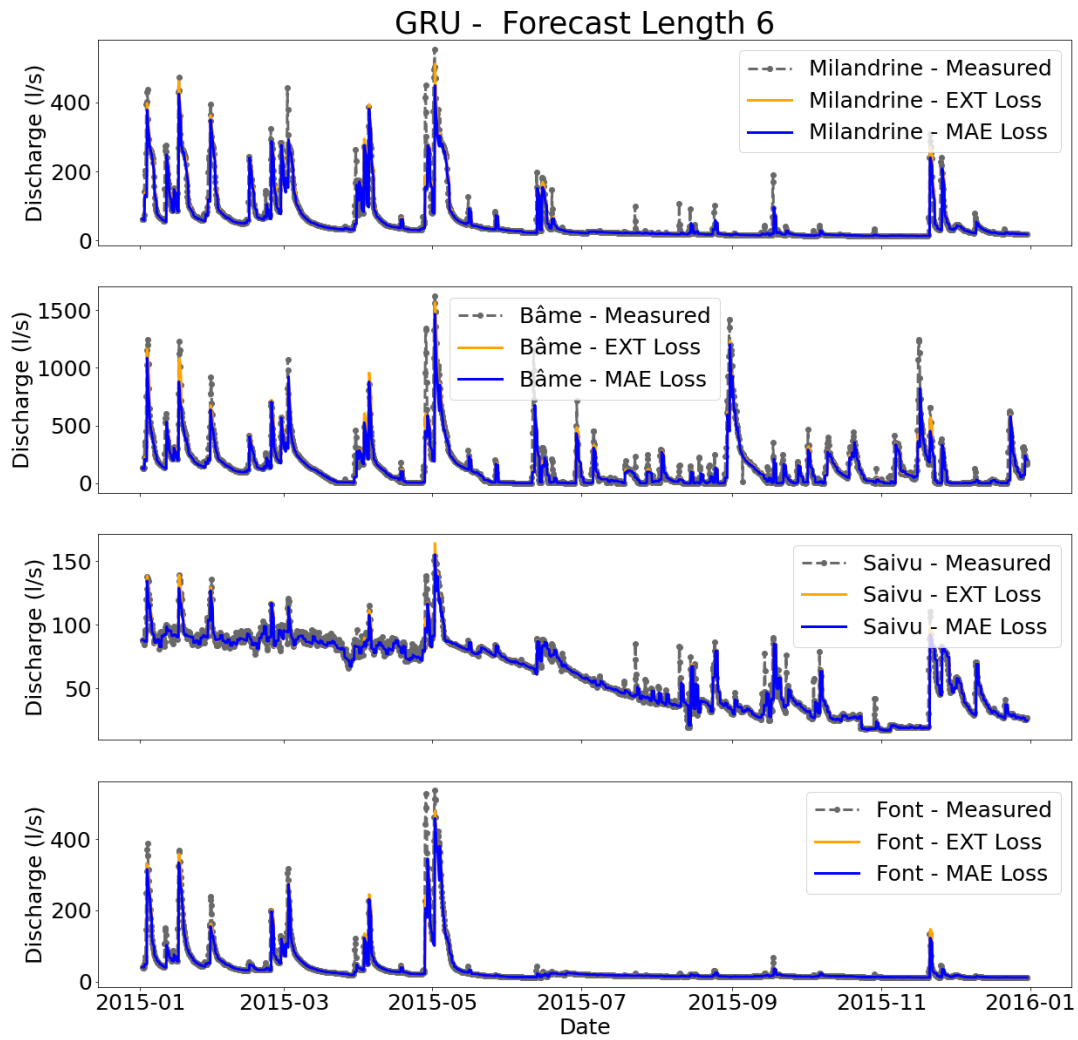
All data and codes for this study are available on Github : [https://github.com/XiaoXia10/ExtremeLoss\\_TimeseriesForecast](https://github.com/XiaoXia10/ExtremeLoss_TimeseriesForecast)



**FIGURE 4.15 : Cross plot of measured and GWN predictions (Yamaska) of 4 time step forecast lengths.**

## Acknowledgments

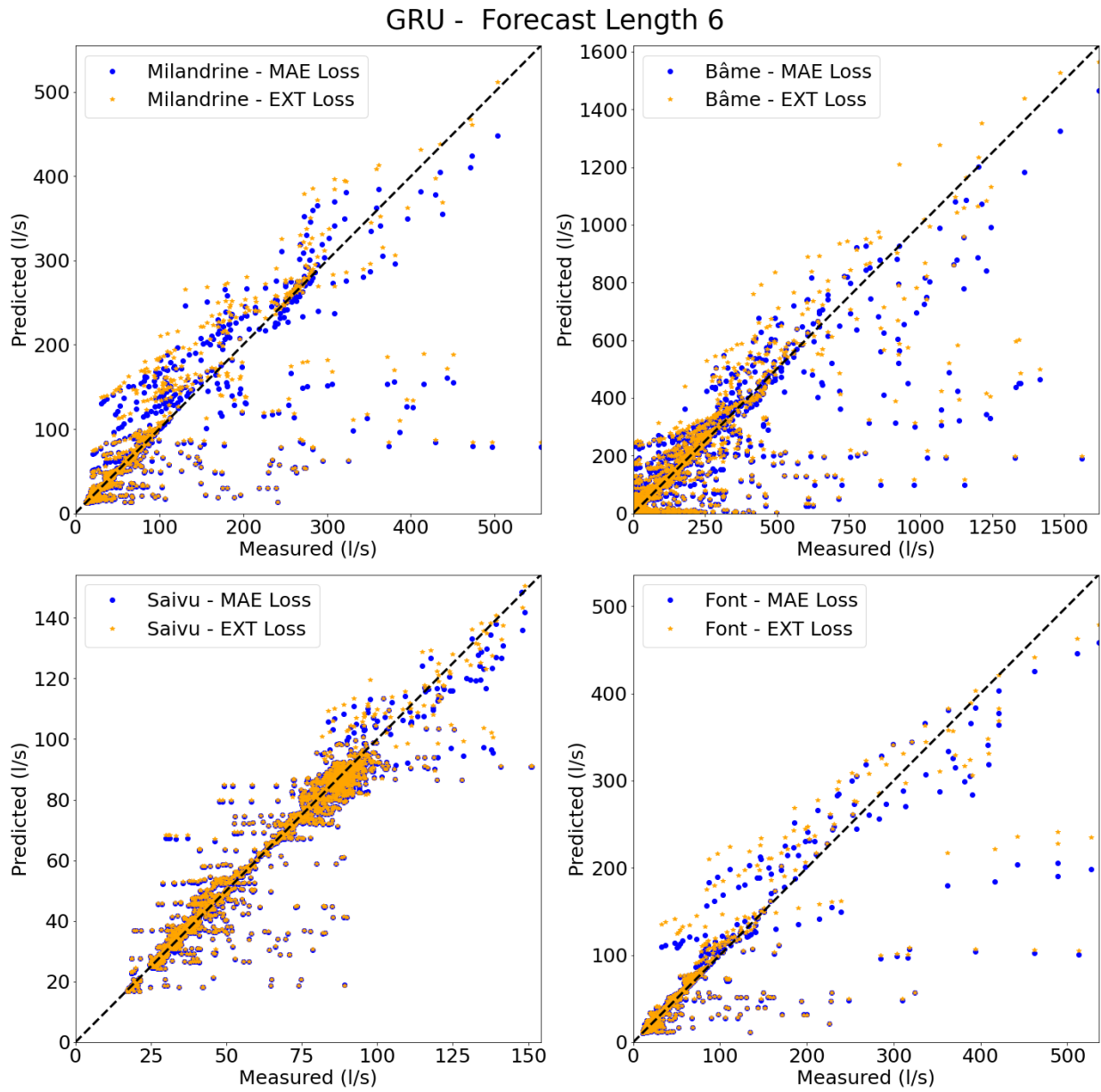
The funding for this study is supported by the PAR grant from Natural Resources Canada. The authors would like to thank Dr. Pierre-Yves Jeannin (Swiss Institute for Speleology and Karst Studies) for providing the Karst spring discharge dataset, and Dr. Daniel Paradis and Lemuel Ramos-Arzola (Institut National de la Recherche Scientifique) for providing data on the Yamaska watershed.



**FIGURE 4.16 : GRU model forecast results (Milandre) - 6 time step forecast lengths.**

## Appendix A

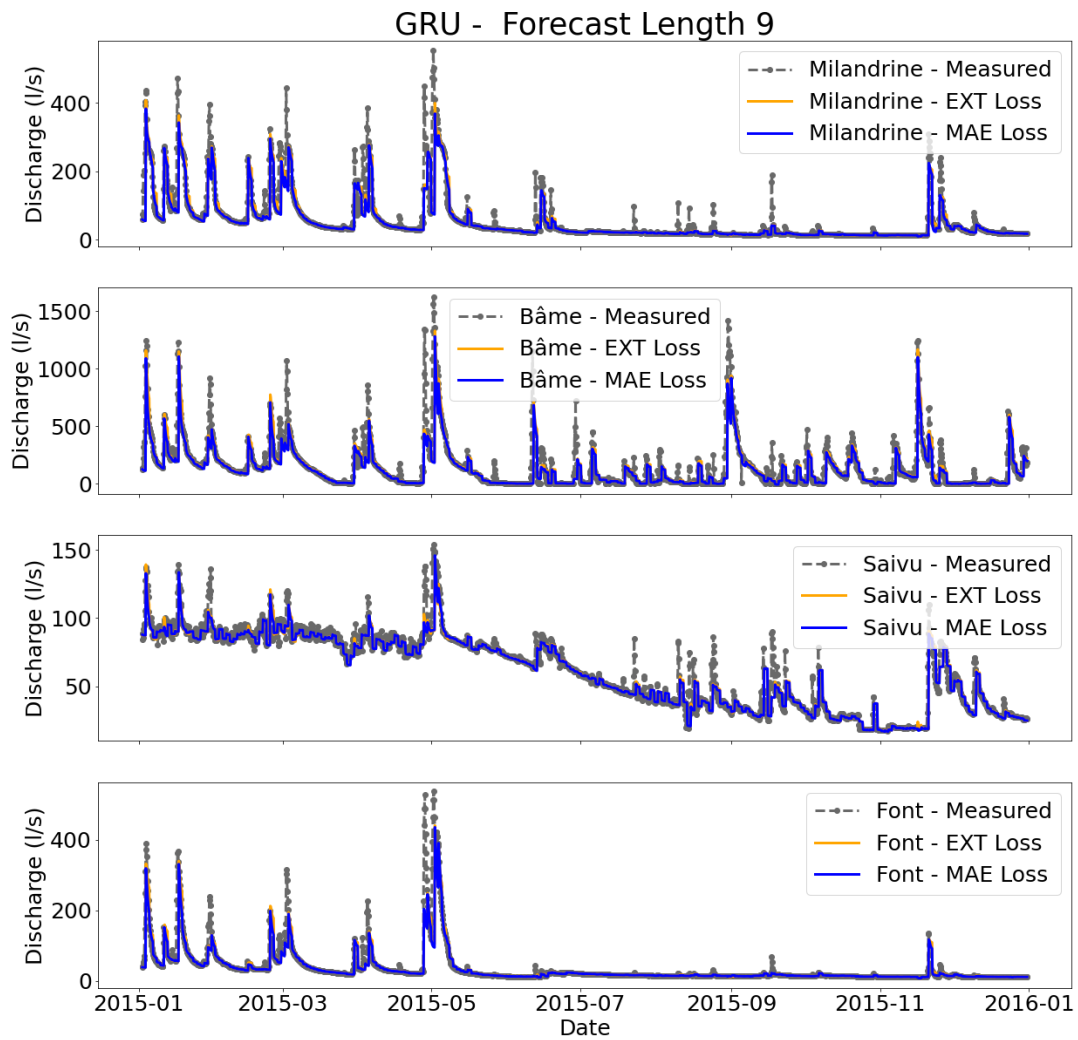
All codes and supplementary data for MPS simulations are available on Github : [https://github.com/XiaoXia10/ExtremeLoss\\_TimeseriesForecast](https://github.com/XiaoXia10/ExtremeLoss_TimeseriesForecast)



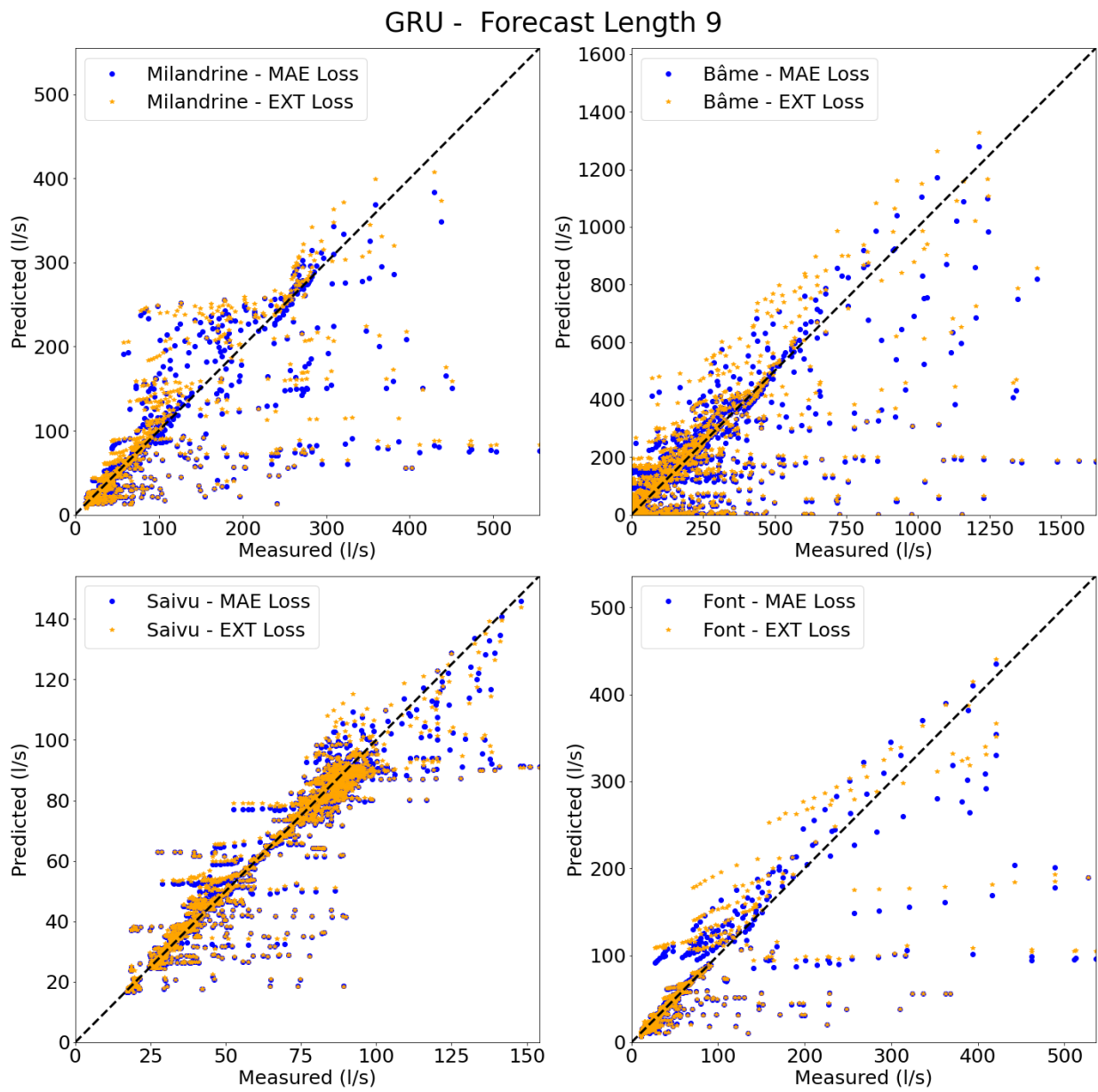
**FIGURE 4.17 : Cross plot of measured and GRU predictions (Milandre) of 6 time step forecast lengths.**

**TABLEAU 4.4 : Test dataset metrics from GWN model with hourly measured data (Milandre)**

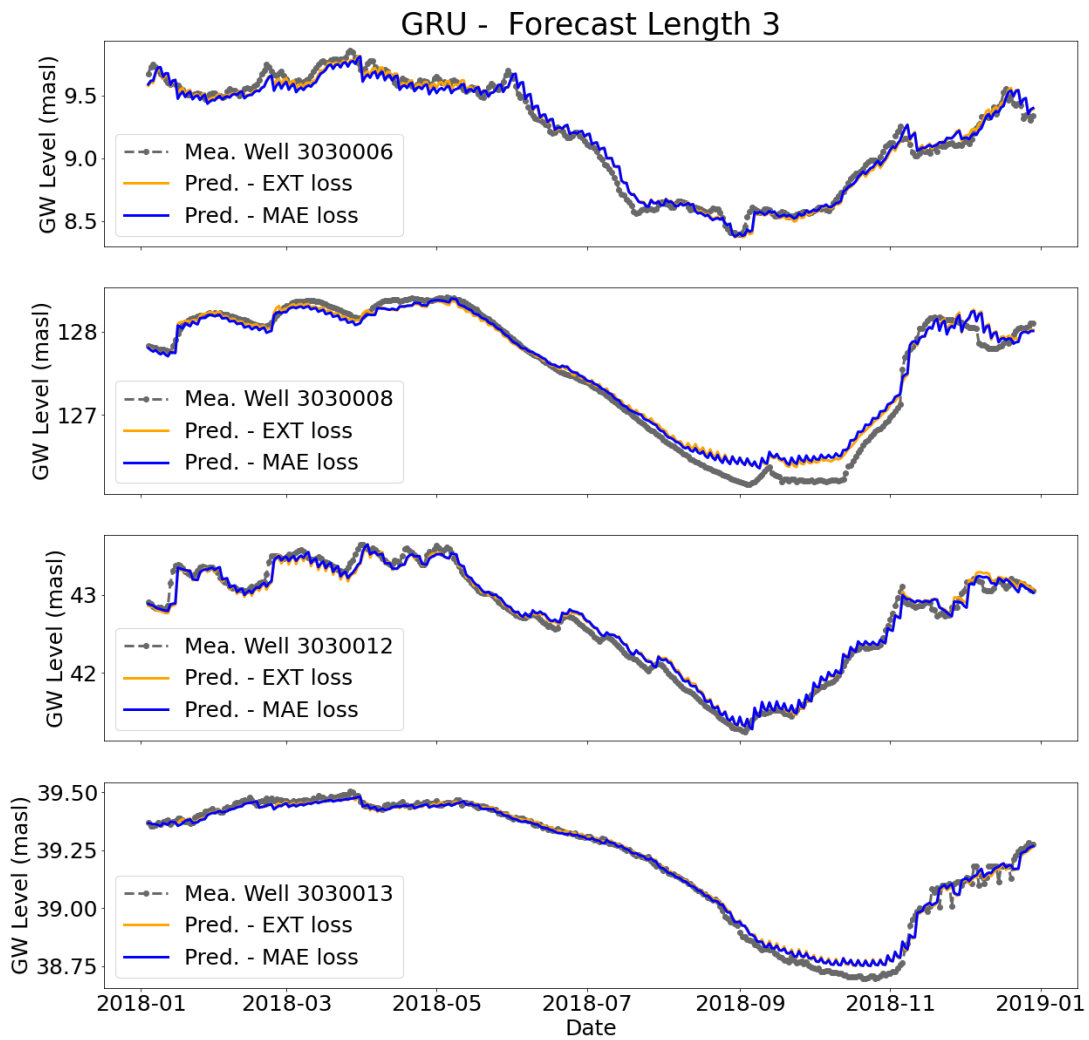
Model	Forecast length	MAE Loss			EXT Loss		
		$R^2$	RMSE (l/s)	MAE (l/s)	$R^2$	RMSE (l/s)	MAE (l/s)
GWN	6	0.95	21.12	7.04	0.96	19.58	6.15
	9	0.94	24.43	7.78	0.94	24.44	8.02
	12	0.91	29.76	9.75	0.91	29.83	10.32
GRU	6	0.959	19.60	6.75	0.962	18.82	6.32
	9	0.936	24.92	8.66	0.941	23.92	8.66
	12	0.914	29.06	9.71	0.910	29.61	10.33



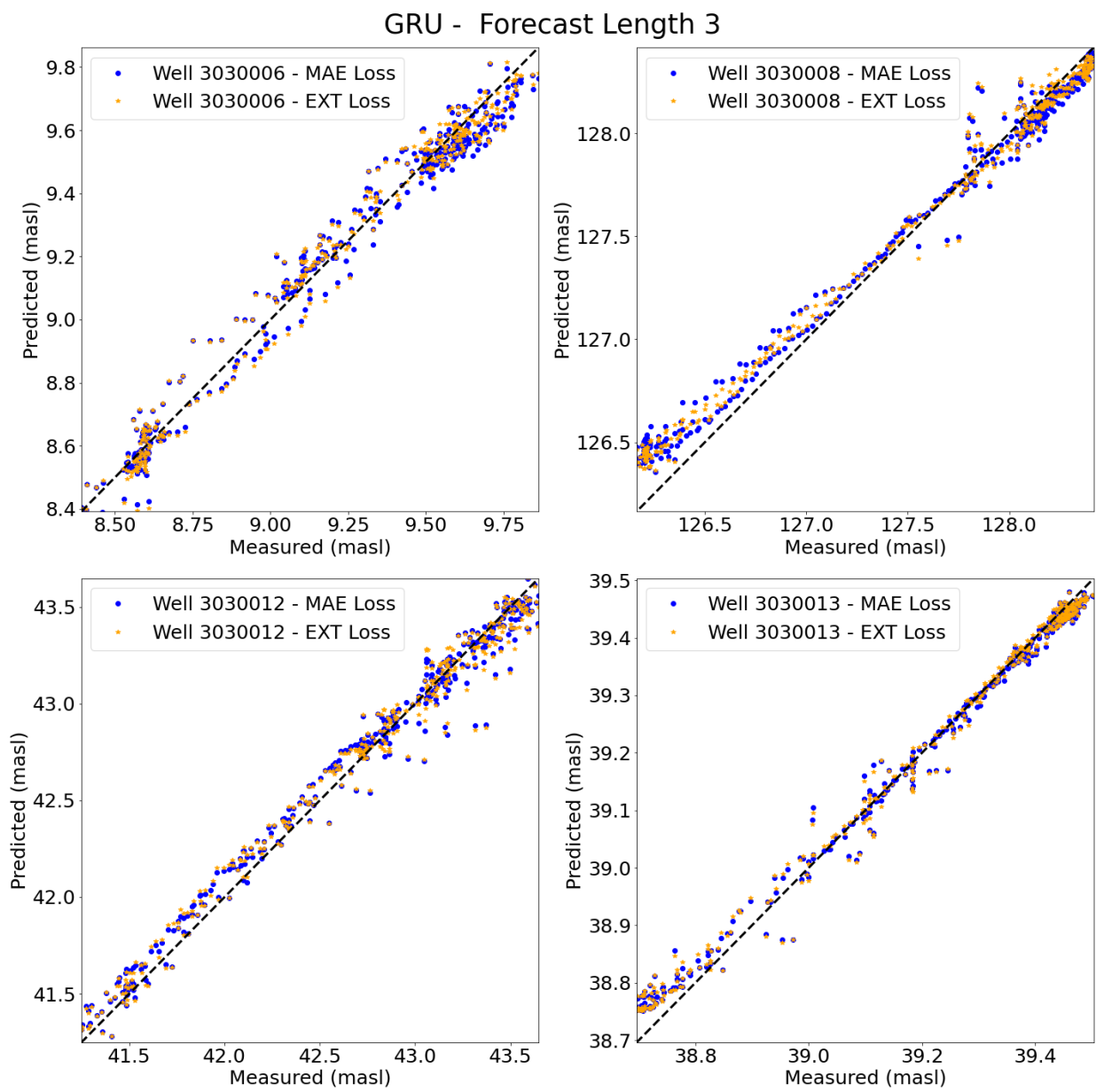
**FIGURE 4.18 : GRU model forecast results (Milandre) - 9 time step forecast lengths.**



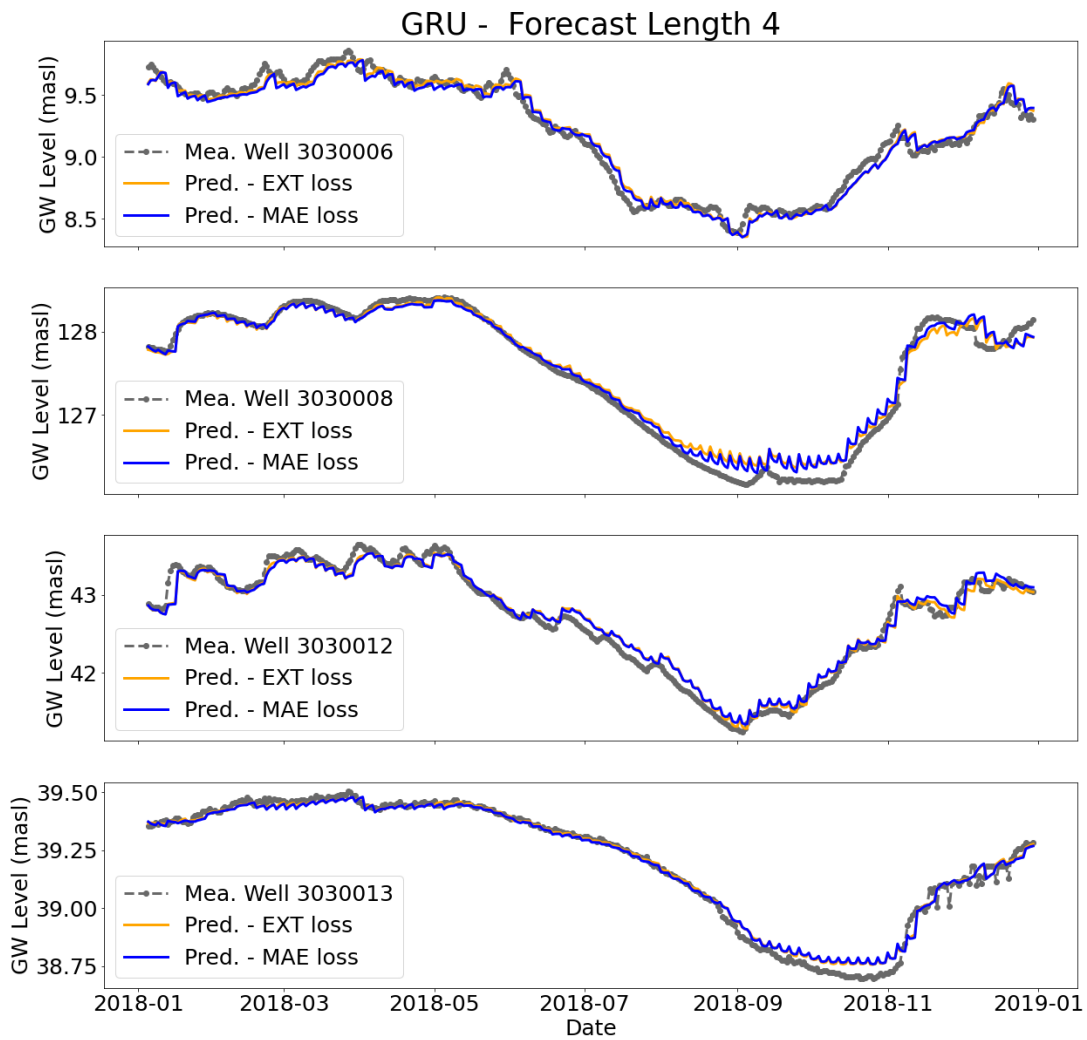
**FIGURE 4.19 : Cross plot of measured and GRU predictions (Milandre) of 9 time step forecast lengths.**



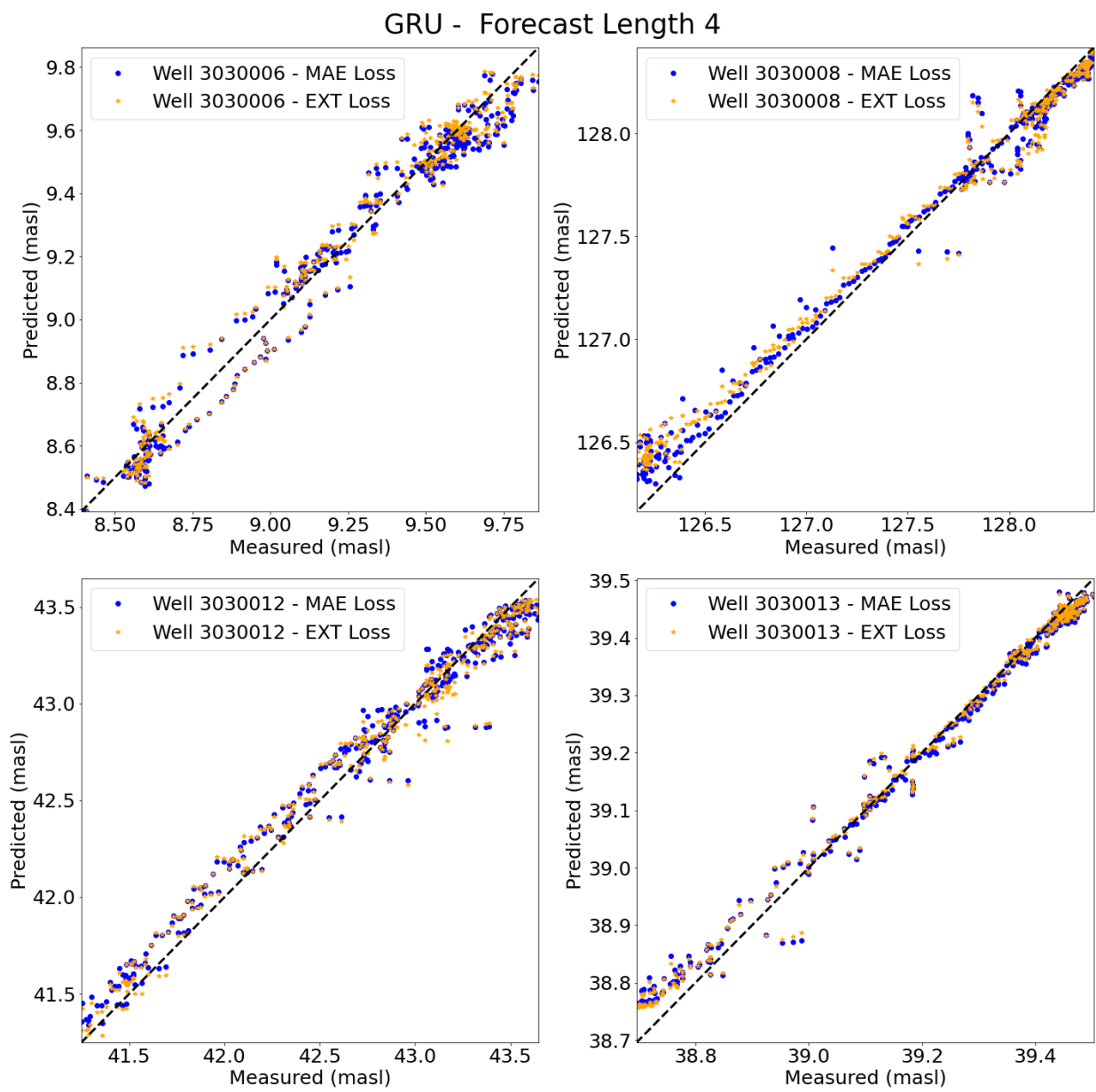
**FIGURE 4.20 : GRU model forecast results (Yamaska) - 3 time step forecast lengths.**



**FIGURE 4.21 : Cross plot of measured and GRU predictions (Yamaska) of 3 time step forecast lengths.**



**FIGURE 4.22 : GRU model forecast results (Yamaska) - 4 time step forecast lengths.**



**FIGURE 4.23 : Cross plot of measured and GRU predictions (Yamaska) of 4 time step forecast lengths.**

## 5 DISCUSSION

---

Durant ma thèse, je me suis consacrée à la création d'algorithmes permettant la prévision spatio-temporelle des niveaux d'eau à l'aide de modèles d'apprentissage automatique, plus particulièrement les modèles de réseaux de neurones par graphes. J'ai développé ces algorithmes pour deux types d'écoulements différents, soit l'écoulement de l'eau souterraine dans les aquifères granulaires et de roc fracturé du sud du Québec et l'écoulement karstique dans le nord-ouest de la Suisse. Pour ce faire, j'ai atteint trois objectifs techniques, chacun correspondant à un article soumis ou en préparation dans des revues scientifiques revisées par les pairs.

Pour le premier objectif, un cadre intégrant un modèle RNG a été développé comme substitut d'un modèle numérique hydrogéologique afin de prévoir les niveaux piézométriques spatio-temporels dans les puits de surveillance et de pompage dans un aquifère granulaire en contact avec un aquifère rocheux typique de la vallée du Saint-Laurent au Québec. Le cadre développé dans ce contexte est déployé avec succès et est capable de prévoir les niveaux de nappe phréatique dans des puits avec des débits de pompage variables. Deux scénarios de pompage sont appliqués au modèle physique pour réaliser des simulations transitoires. Le modèle GCN-LSTM, entraîné avec les données simulées, atteint un coefficient de corrélation  $R^2$  supérieur à 0,92 pour la prévision des niveaux d'eau pour les données de l'ensemble des tests. Les résultats démontrent que cet algorithme peut servir de substitut aux modèles physiques de l'écoulement de l'eau souterraine.

Une des contributions a été de montrer qu'il est possible d'utiliser les données générées par modélisation numérique afin d'entraîner un algorithme d'apprentissage profond. En effet, les séries temporelles de niveaux d'eau au Québec (et ailleurs dans le monde) sont généralement rares et de faible résolution. Certaines juridictions européennes ont commencé à installer des sondes automatiques qui envoient les données en temps réel pour le suivi des nappes et la prédiction des étiages. En France, par exemple, le Bureau de Recherche Géologique et des Mines (BRGM) a déployé un réseau de 1650 puits avec des sondes fournissant des données en temps réel (<https://www.brgm.fr/fr/etat-nappes-eau-souterraine-suivi-assure-brgm>). Dans quelques années, ces séries temporelles réelles pourront être utilisées seules ou de concert avec les données modélisées, pour entraîner des réseaux de neurones permettant de prédire le comportement des nappes en temps quasi-réel.

L'algorithme que j'ai développé n'a été testé que pour un cas d'étude et pour l'écoulement de l'eau souterraine seulement. Son extension pour prédire le transport advectif des contaminants serait précieuse. De plus, l'adaptation du modèle RNG pour le rendre inductif permettrait une plus grande flexibilité dans la mise à jour des puits d'observation d'un réseau de surveillance. Par exemple, en adoptant un modèle RNG inductif, il serait possible d'estimer les niveaux de nappe phréatique dans des puits non échantillonnés à partir des puits échantillonnés, grâce à la propagation de l'information le long des arêtes du graphe. En effet, un modèle RNG inductif

---

peut traiter des graphes dynamiques, où des nœuds (puits) peuvent être ajoutés ou supprimés, permettant ainsi aux nouveaux puits d'exploiter l'information des puits existants via la structure du graphe. En outre, ce modèle RNG inductif pourrait également être entraîné avec des séries temporelles de niveaux de contaminants mesurés dans les puits, facilitant ainsi l'estimation des concentrations de contaminants dans des régions où les mesures directes sont absentes.

Le deuxième objectif a été d'étudier si les modèles d'AA, tels que les modèles RNG, peuvent prévoir avec précision les niveaux d'eau spatio-temporels dans un réseau karstique en n'utilisant que les séries temporelles de terrain mesurées à quelques stations. Dans cette étude, le jeu de données utilisé est un ensemble de mesures dans un réseau karstique suisse rendu disponible grâce au partenariat avec l'université de Neuchâtel. Une analyse des critères nécessaires pour obtenir des prévisions précises a été effectuée. Il a été montré que, pour cet ensemble de données, le critère principal pour obtenir des prévisions précises des niveaux d'eau est la résolution temporelle des données. Bien que la taille du jeu de données soit importante, la résolution des données s'avère être un facteur encore plus critique. Cette conclusion a un impact majeur sur la surveillance des réseaux karstiques et donc, sur les communautés qui dépendent de cette source d'eau potable.

En plus de l'analyse de sensibilité sur les données, trois modèles RNP différents (un modèle auto-encodeur LSTM, un modèle auto-encodeur GRU et le modèle GWN) ont été testés. Le modèle GWN a été identifié comme étant le plus performant pour différentes résolutions de données et différentes étapes de prévision. Cet objectif s'inscrit dans la continuité du premier, qui consistait à développer un cadre de substitution aux modèles physiques afin de surmonter les défis liés à la modélisation physique. Dans le cas des systèmes karstiques, l'objectif est de déterminer si les modèles d'AA peuvent remplacer les modèles physiques. En effet, pour ce type d'écoulement, la plus grande incertitude lors de la modélisation est de caractériser le réseau de karsts en effectuant des campagnes spéléologiques. Ces campagnes restent donc limitées à l'accès que les humains peuvent avoir dans le réseau. Des tests de conservation de traceurs sont aussi effectués, mais ils ne renseignent pas sur la structure physique du réseau en tant que tel.

La portée de cette étude pourrait être élargie si des données plus volumineuses étaient disponibles, car les données utilisées ne s'étendaient que sur une période de trois ans. Ce jeu de données a été choisi en raison de sa pertinence pour d'autres études en hydrogéologie et hydrologie, étant donné que l'acquisition de jeux de données de très grande taille demeure un défi. Bien que cette étude permette de déterminer la quantité minimale de données requise pour obtenir des prévisions précises, il serait utile d'incorporer davantage de données d'entraînement afin d'évaluer comment et si l'augmentation des données améliorera la précision des niveaux de débit des sources. Cependant, je n'ai pas à ma disposition de jeu de données temporel de grande taille provenant de plusieurs puits qui pourrait être utilisé à des fins de test. Il serait intéressant de pouvoir mesurer les temps de réaction des débits des sources à des facteurs de stress tels que la pluie et la fonte des neiges printanières afin de comparer ces temps de réaction à la résolution d'échan-

---

tillonnage des données testée dans cette étude. Ce type d'essai sur le terrain serait spécifique au site, car chaque système karstique est unique et ne peut donc pas être généralisé.

Le troisième objectif a été de développer une nouvelle fonction de perte (extreme loss function; ELF) afin d'améliorer la capacité des modèles d'apprentissage automatique à prédire les événements extrêmes dans les séries temporelles. Étant donné que la prévision précise des niveaux d'eau peut être réalisée à l'aide de modèles d'apprentissage automatique, comme démontré dans l'Objectif 2, il est pertinent d'étendre encore davantage leurs capacités.

Pour cet objectif, deux modèles différents sont testés avec deux jeux de données distincts. L'un provient d'un système karstique en Suisse, tandis que l'autre est issu d'un jeu de données de puits de surveillance au Québec, Canada. L'ELF est appliquée aux modèles auto-encodeur GRU et GWN. Les résultats indiquent que l'utilisation de l'ELF améliore la performance des prévisions par rapport aux modèles entraînés avec la fonction de perte basée sur l'erreur moyenne absolue (EMA) pour les deux jeux de données. Le modèle GWN profite davantage de l'amélioration de la prédiction des niveaux d'eau avec l'ELF que le modèle auto-encodeur GRU. Bien que le modèle auto-encodeur GRU présente déjà une grande précision dans la prévision des événements extrêmes et ne nécessite pas nécessairement l'ELF, l'intégration de la nouvelle fonction de perte conduit néanmoins à de légères améliorations des performances de prévision pour les deux jeux de données.

La fonction ELF pourrait être améliorée davantage par l'ajout d'une fonction dynamique pour ajuster la variance des données. De plus, une optimisation de la fonction ELF en termes d'efficacité computationnelle serait également bénéfique.



## 6 CONCLUSION

---

Comme démontré dans cette thèse, les modèles de RNG sont des outils puissants pour la gestion des ressources en eau souterraine, notamment dans le contexte des pressions anthropiques et climatiques. Ces modèles ont le potentiel d'améliorer la prise de décision en fournissant des prévisions précises et basées sur les données spatio-temporelles des niveaux d'eau. Ces modèles peuvent être couplés ou non avec les modèles numériques de l'écoulement de l'eau souterraine. Les résultats ouvrent aussi la question sur l'acquisition systématique de données temporelles pour le suivi des nappes et leur gestion à long terme. Ces données permettent d'entraîner des réseaux de neurones profonds mais aussi permettraient de mieux calibrer les modèles numériques classiques. Compte tenu de leurs capacités prometteuses et de l'accès à des séries temporelles de niveaux d'eau, la poursuite des recherches et du développement des approches RNG serait hautement bénéfique pour la communauté scientifique ainsi que pour les gestionnaires des ressources en eau.

Un aspect clé de ce travail est l'engagement en faveur des logiciels ouverts et librement accessibles. Les outils en accès libre favorisent non seulement la transparence et la reproductibilité, mais permettent également à une communauté plus large de chercheurs, de décideurs et de praticiens d'améliorer, d'adapter et d'appliquer ces modèles à divers défis hydrogéologiques et hydrologiques. En encourageant la collaboration et le partage des connaissances, les logiciels en accès libre joueront un rôle crucial dans l'avancement de la modélisation des eaux souterraines et assureront une gestion durable des ressources en eau.

Les modèles RNG sont indéniablement utiles, mais ils peuvent être difficiles à entraîner, et leur architecture en graphe est particulièrement complexe. Contrairement aux modèles d'AA classiques qui fonctionnent sur des données structurées, les GNN nécessitent un graphe défini explicitement, ce qui peut être difficile à concevoir et à optimiser. Il n'existe pas de normes universelles pour la construction des graphes, car leur structure optimale dépend fortement du système étudié et des données disponibles. Des graphes mal construits peuvent entraîner un apprentissage sous-optimal, des coûts de calculs élevés et une réduction de l'interprétabilité du modèle. Ces défis font de la construction des graphes une étape cruciale, mais difficile, pour l'application des GNN à des problèmes concrets tels que la prévision des niveaux de nappe phréatique dans des systèmes aquifères complexes. En dépit de ces défis, il reste avantageux de développer et d'utiliser ces modèles en hydrogéologie. Une fois entraînés par un expert en IA, ils peuvent être utilisés par toute personne ayant des connaissances de base en hydrogéologie et en programmation. Actuellement, la plupart des modèles d'AA sont en libre accès et ne nécessitent pas de licence, ce qui réduit également la pression financière sur les utilisateurs.

Un domaine d'intérêt est l'intégration des réseaux de neurones de type bayésien (de Freitas, 2003; Mullachery et al., 2018) avec les modèles de RNG afin de quantifier l'incertitude dans les prévi-

---

sions. Cette approche améliorerait l'interprétabilité des prédictions et offrirait aux chercheurs une compréhension plus approfondie du niveau de confiance du modèle. En intégrant l'inférence bayésienne, le modèle peut générer des sorties probabilistes qui prennent en compte l'incertitude. Une telle intégration est particulièrement précieuse dans les applications où la prise de décision repose sur le niveau de confiance des prédictions. Les travaux futurs pourraient explorer l'intégration des réseaux de type bayésien et évaluer plus en détail l'incertitude des prédictions des modèles RNG.

Il ne fait aucun doute que l'intérêt pour les méthodes d'AA a connu une croissance exponentielle ces dernières années. Cet engouement est compréhensible, car les modèles d'AA sont des outils puissants, applicables et utiles dans de nombreux domaines dans lesquels les données portent en elles l'information nécessaire à la prédiction. Il est aussi à noter qu'il existe des hypothèses sur les données, comme la stationnarité. Dans le cas où les conditions du système aquifère changent de façon dramatique, rien ne garantit la justesse des prévisions par une AA entraînée dans des conditions différentes. Dans ce cas, le couplage et l'entraînement avec des modèles numériques de l'écoulement seraient à envisager. L'autre aspect non investigué est le transfert de domaine (domain gap), à savoir si un RNG entraîné sur un aquifère peut être appliqué ou ré-entraîné en quelques itérations (few shot learning) sur un aquifère similaire.

En conclusion, les modèles d'AA sont actuellement à la mode, mais ils ne sont pas triviaux à mettre en place et à déployer de manière opérationnelle. Leur utilisation sans une étude approfondie et rigoureuse n'est pas recommandée.

## BIBLIOGRAPHIE

- Aabø TM, Dramsch JS, Würtzen CL, Seyum S, Welch M (2020) An integrated workflow for fracture characterization in chalk reservoirs, applied to the kraka field. *Marine and Petroleum Geology*, 112:104065.
- Adamowski J, Chan HF (2011) A wavelet neural network conjunction model for groundwater level forecasting. *Journal of Hydrology*, 407(1-4):28–40.
- Afzaal H, Farooque AA, Abbas F, Acharya B, Esau T (2019) Groundwater estimation from major physical hydrology components using artificial neural networks and deep learning. *Water*, 12(1): 5.
- Alzubaidi L, Zhang J, Humaidi AJ, Al-Dujaili A, Duan Y, Al-Shamma O, Santamaría J, Fadhel MA, Al-Amidie M, Farhan L (2021) Review of deep learning : concepts, cnn architectures, challenges, applications, future directions. *Journal of big Data*, 8:1–74.
- An L, Hao Y, Yeh TCJ, Liu Y, Liu W, Zhang B (2020) Simulation of karst spring discharge using a combination of time–frequency analysis methods and long short-term memory neural networks. *Journal of Hydrology*, 589:125320.
- An Y, Lu W, Yan X (2018) A surrogate-based simulation–optimization approach application to parameters' identification for the hydrogeosphere model. *Environmental Earth Sciences*, 77(17): 621.
- Anderson MP, Woessner WW, Hunt RJ (2015) *Applied groundwater modeling : simulation of flow and advective transport*. Academic Press.
- Assem H, Ghariba S, Makrai G, Johnston P, Gill L, Pilla F (2017) Urban water flow and water level prediction based on deep learning. *Machine Learning and Knowledge Discovery in Databases : European Conference, ECML PKDD 2017, Skopje, Macedonia, September 18–22, 2017, Proceedings, Part III 10*, Springer, pages 317–329.
- Atwood J, Towsley D (2016) Diffusion-convolutional neural networks. *Advances in neural information processing systems*, 29.
- Bai T, Tahmasebi P (2023) Graph neural network for groundwater level forecasting. *Journal of Hydrology*, 616:128792.
- Bakker M, Schaars F (2019) Solving groundwater flow problems with time series analysis : you may not even need another model. *Groundwater*, 57(6):826–833.
- Bastian M, Heymann S, Jacomy M (2009a) Gephi : an open source software for exploring and manipulating networks. *Proceedings of the International AAAI Conference on Web and Social Media*, pages 361–362.
- Bastian M, Heymann S, Jacomy M (2009b) *Gephi : An Open Source Software for Exploring and Manipulating Networks*. <http://www.aaai.org/ocs/index.php/ICWSM/09/paper/view/154>.
- Bates B, Kundzewicz Z, Wu S (2008) *Climate change and water*. Intergovernmental Panel on Climate Change Secretariat.

- 
- Beaudry C, Lefebvre R, Rivard C, Cloutier V (2018) Conceptual model of regional groundwater flow based on hydrogeochemistry (montérégie est, québec, canada). *Canadian Water Resources Journal/Revue canadienne des ressources hydriques*, 43(2):152–172.
- Béraud T, Claprood M, Gloaguen E (2024) A sequential ensemble smoother for multiple data assimilation in hydrogeological modelling. *Frontiers*.
- Bowes BD, Sadler JM, Morsy MM, Behl M, Goodall JL (2019) Forecasting groundwater table in a flood prone coastal city with long short-term memory and recurrent neural networks. *Water*, 11(5):1098.
- Brakenhoff DA, Vonk MA, Collenteur RA, Van Baar M, Bakker M (2022) Application of time series analysis to estimate drawdown from multiple well fields. *Frontiers in earth science*, 10:907609.
- Bruna J, Zaremba W, Szlam A, LeCun Y (2013) Spectral networks and locally connected networks on graphs. *arXiv preprint arXiv :1312.6203*.
- Bui KHN, Cho JW, Yi HK (2022) Spatial-temporal graph neural network for traffic forecasting : An overview and open research issues. *Applied Intelligence*, 52(3):2763–2774.
- Cafilisch A (2006) Network and graph analyses of folding free energy surfaces. *Current Opinion in Structural Biology*, 16(1):71–78.
- Chang Y, Wu J, Jiang G (2015) Modeling the hydrological behavior of a karst spring using a nonlinear reservoir-pipe model. *Hydrogeology Journal*, 23(5):901.
- Che Z, Purushotham S, Cho K, Sontag D, Liu Y (2018) Recurrent neural networks for multivariate time series with missing values. *Scientific reports*, 8(1):6085.
- Chen X, Zhang Z, Soulsby C, Cheng Q, Binley A, Jiang R, Tao M (2018) Characterizing the heterogeneity of karst critical zone and its hydrological function : An integrated approach. *Hydrological Processes*, 32(19):2932–2946.
- Cheng S, Qiao X, Shi Y, Wang D (2020) Comparison of machine learning methods for predicting karst spring discharge in north china. *arXiv preprint arXiv :2007.12951*.
- Cho K, Van Merriënboer B, Gulcehre C, Bahdanau D, Bougares F, Schwenk H, Bengio Y (2014) Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv :1406.1078*.
- Claprood M, Béraud T, Boutin LC, Gloaguen E, Martel R (2023) Rapport final du projet : Utes-ii ville mercier. INRS–ETE. 126 pages.
- Claprood M, Gloaguen E, Béraud T et al. (2022) A case study using seismic reflection and well logs to reduce and quantify uncertainty during a hydrogeological assessment. *Frontiers in Water*, 3:779149.
- Clark SR, Pagendam D, Ryan L (2022) Forecasting multiple groundwater time series with local and global deep learning networks. *International Journal of Environmental Research and Public Health*, 19(9):5091.
- Cook BI, Mankin JS, Anchukaitis KJ (2018) Climate change and drought : From past to future. *Current Climate Change Reports*, 4:164–179.

- 
- Côrte G, Dramsch J, Amini H, MacBeth C (2020) Deep neural network application for 4d seismic inversion to changes in pressure and saturation : Optimizing the use of synthetic training datasets. *Geophysical Prospecting*, 68(7):2164–2185.
- Cortes C, Vapnik V (1995) Support-vector networks. *Machine learning*, 20:273–297.
- Dai A (2013) Increasing drought under global warming in observations and models. *Nature climate change*, 3(1):52–58.
- Daliakopoulos IN, Coulibaly P, Tsanis IK (2005) Groundwater level forecasting using artificial neural networks. *Journal of hydrology*, 309(1-4):229–240.
- Dauphin YN, Fan A, Auli M, Grangier D (2017) Language modeling with gated convolutional networks. *Proceedings of the 34th International Conference on Machine Learning*. Precup D, Teh YW (éditeurs), PMLR, volume 70 de *Proceedings of Machine Learning Research*, pages 933–941.
- de Freitas JFG (2003). *Bayesian methods for neural networks*. Thèse de doctorat, University of Cambridge.
- Defferrard M, Bresson X, Vandergheynst P (2016). Convolutional neural networks on graphs with fast localized spectral filtering. *Advances in Neural Information Processing Systems*, volume 29.
- Delottier H, Therrien R, Young N, Paradis D (2022). A hybrid approach for integrated surface and subsurface hydrologic simulation of baseflow with iterative ensemble smoother. *Journal of Hydrology*, 606:127406.
- Devlin J (2018). Bert : Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv :1810.04805*.
- Diehl F, Brunner T, Le MT, Knoll A (2019). Graph neural networks for modelling traffic participant interaction. *2019 IEEE Intelligent Vehicles Symposium (IV)*, IEEE, pages 695–701.
- Do TD, Duong MT, Dang QV, Le MH (2018). Real-time self-driving car navigation using deep neural network. *2018 4th International Conference on Green Technology and Sustainable Development (GTSD)*, IEEE, pages 7–12.
- Doumar J, Kassem AH, Gurdak JJ (2018). Impact of historic and future climate on spring recharge and discharge based on an integrated numerical modelling approach : Application on a snow-governed semi-arid karst catchment area. *Journal of Hydrology*, 565:636–649.
- Doumar J, Sauter M, Geyer T (2012). Simulation of flow processes in a large scale karst system with an integrated catchment model (mike she)–identification of relevant parameters influencing spring discharge. *Journal of Hydrology*, 426:112–123.
- Dramsch JS (2020). 70 years of machine learning in geoscience in review. *Advances in geophysics*, 61:1–55.
- Dramsch JS, Christensen AN, MacBeth C, Lüthje M (2021). Deep unsupervised 4-d seismic 3-d time-shift estimation with convolutional neural networks. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–16.
- Evensen G (2003). The ensemble kalman filter : Theoretical formulation and practical implementation. *Ocean Dynamics*, 53:343–367.

- 
- Evensen G (2009). The ensemble kalman filter for combined state and parameter estimation. *IEEE Control Systems Magazine*, 29(3):83–104.
- Fang L, Shao D (2022). Application of long short-term memory (lstm) on the prediction of rainfall-runoff in karst area. *Frontiers in Physics*, 9:790687.
- Favre I (2001). *Base de données des essais de traçage du plateau karstique de Bure (JU), SIG, interprétations statistiques*. Centre d'hydrogéologie-Université de Neuchâtel.
- Fine TL (2006). *Feedforward neural network methodology*. Springer Science & Business Media.
- Fukushima K (1988). Neocognitron : A hierarchical neural network capable of visual pattern recognition. *Neural networks*, 1(2):119–130.
- Gai Y, Wang M, Wu Y, Wang E, Deng X, Liu Y, Yeh TCJ, Hao Y (2023). Simulation of spring discharge using graph neural networks at niangziguan springs, china. *Journal of Hydrology*, 625:130079.
- Gal Y, Ghahramani Z (2016). Dropout as a bayesian approximation : Representing model uncertainty in deep learning. *international conference on machine learning*, PMLR, pages 1050–1059.
- Gardner MA, Sunkavalli K, Yumer E, Shen X, Gambaretto E, Gagné C, Lalonde JF (2017). Learning to predict indoor illumination from a single image. *arXiv preprint arXiv :1704.00090*.
- Ghanbarian-Alavijeh B, Liaghat A, Sohrabi S (2010). Estimating saturated hydraulic conductivity from soil physical properties using neural networks model. *World Acad. Sci. Eng. Technol*, 4: 108–113.
- Globensky Y (1987). *Géologie des basses-terres du Saint-Laurent*. Gouvernement du Québec, Ministère de l'énergie et des ressources, Direction . . .
- Gopalakrishnan K, Khaitan SK, Choudhary A, Agrawal A (2017). Deep convolutional neural networks with transfer learning for computer vision-based data-driven pavement distress detection. *Construction and building materials*, 157:322–330.
- Grasso D, Jeannin P (1994). Estimation des pertes dans la partie aval du reseau karstique de la milandrine : bilan hydrique au sein d'un aquifère karstique. *Bulletin d'Hydrogéologie*, 13:115–128.
- Hamilton W, Ying Z, Leskovec J (2017a). Inductive representation learning on large graphs. *Advances in neural information processing systems*, 30.
- Hamilton WL, Ying R, Leskovec J (2017b). Representation learning on graphs : Methods and applications. *arXiv preprint arXiv :1709.05584*.
- He Q, Barajas-Solano D, Tartakovsky G, Tartakovsky AM (2020). Physics-informed neural networks for multiphysics data assimilation with application to subsurface transport. *Advances in Water Resources*, 141:103610.
- He Q, Tartakovsky AM (2021a). Physics-informed neural network method for forward and backward advection-dispersion equations. *Water Resources Research*, 57(7):e2020WR029479.
- He X, He Q, Chen JS (2021b). Deep autoencoders for physics-constrained data-driven nonlinear materials modeling. *Computer Methods in Applied Mechanics and Engineering*, 385:114034.

- 
- Hillier M, Wellmann F, Brodaric B et al. (2021). Three-dimensional structural geological modeling using graph neural networks. *Mathematical Geosciences*, 53(8):1725–1749.
- Ho TK (1995). Random decision forests. *Proceedings of 3rd international conference on document analysis and recognition*, IEEE, volume 1, pages 278–282.
- Hochreiter S, Schmidhuber J (1997). Long short-term memory. *Neural Computation*, 9(8):1735–1780.
- Hopfield JJ (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the national academy of sciences*, 79(8):2554–2558.
- Hunt RJ, Welter DE (2010). Taking account of “unknown unknowns”. *Ground Water*, 48(4).
- Irshad O, Khan MUG, Iqbal R, Basheer S, Bashir AK (2020). Performance optimization of iot based biological systems using deep learning. *Computer Communications*, 155:24–31.
- Jadon A, Patil A, Jadon S (2024). A comprehensive survey of regression-based loss functions for time series forecasting. *International Conference on Data Management, Analytics & Innovation*, Springer, pages 117–147.
- Jasechko S, Seybold H, Perrone D, Fan Y, Shamsudduha M, Taylor RG, Fallatah O, Kirchner JW (2024). Rapid groundwater decline and some cases of recovery in aquifers globally. *Nature*, 625(7996):715–721.
- Jeannin PY, Artigue G, Butscher C, Chang Y, Charlier JB, Duran L, Gill L, Hartmann A, Johannet A, Jourde H et al. (2021). Karst modelling challenge 1 : Results of hydrological modelling. *Journal of Hydrology*, 600:126508.
- Jin S, Zeng X, Xia F, Huang W, Liu X (2021). Application of deep learning methods in biological networks. *Briefings in bioinformatics*, 22(2):1902–1917.
- Kanagaraj N, Hicks D, Goyal A, Tiwari S, Singh G (2021). Deep learning using computer vision in self driving cars for lane and traffic sign detection. *International Journal of System Assurance Engineering and Management*, 12(6):1011–1025.
- Kang J, Zhang WQ, Liu J (2016). Gated recurrent units based hybrid acoustic models for robust speech recognition. *2016 10th international symposium on Chinese spoken language processing (ISCSLP)*, IEEE, pages 1–5.
- Kawahara M, Inoue T, Nishio Y (2010). Image processing application using cnn with dynamic template. *2010 12th International Workshop on Cellular Nanoscale Networks and their Applications (CNNA 2010)*, IEEE, pages 1–6.
- Khan S, Rahmani H, Shah SAA, Bennamoun M, Medioni G, Dickinson S (2018). A guide to convolutional neural networks for computer vision.
- Kieu T, Yang B, Jensen CS (2018). Outlier detection for multidimensional time series using deep neural networks. *2018 19th IEEE international conference on mobile data management (MDM)*, IEEE, pages 125–134.
- Kingma DP, Ba J (2014). Adam : A method for stochastic optimization. *arXiv preprint arXiv :1412.6980*.

- 
- Kingma DP, Ba J (2017). Adam : A method for stochastic optimization. *arXiv preprint arXiv :1412.6980*.
- Kipf TN, Welling M (2016). Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv :1609.02907*.
- Kovács A, Jeannin PY (2003). Hydrogeological overview of the bure plateau, ajoie, switzerland. *Ecolgae Geologicae Helvetiae*, 96(3):367–380.
- Krige D (1951). A statistical approach to some basic mine valuation problems on the witwatersrand. *J Chem Metal Min Soc S Afr*, 52:119–139.
- Krumbein WC, Dacey MF (1969). Markov chains and embedded markov chains in geology. *Journal of the International Association for Mathematical Geology*, 1(1):79–96.
- Latif J, Xiao C, Imran A, Tu S (2019). Medical imaging using machine learning and deep learning algorithms : a review. *2019 2nd International conference on computing, mathematics and engineering technologies (iCoMET)*, IEEE, pages 1–5.
- Lauzon D, Marcotte D (2023). Joint hydrofacies-hydraulic conductivity modeling based on a constructive spectral algorithm constrained by transient head data. *Hydrogeology Journal*, 31(6):1647–1664.
- LeCun Y, Bottou L, Bengio Y, Haffner P (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.
- LeCun Y, Jackel LD, Bottou L, Cortes C, Denker JS, Drucker H, Guyon I, Muller UA, Sackinger E, Simard P et al. (1995). Learning algorithms for classification : A comparison on handwritten digit recognition. *Neural networks : the statistical mechanics perspective*, 261(276):2.
- Lee SJ, Yoon HK (2021). Discontinuity predictions of porosity and hydraulic conductivity based on electrical resistivity in slopes through deep learning algorithms. *Sensors*, 21(4):1412.
- Li J, Lu W, Luo J (2021). Groundwater contamination sources identification based on the long-short term memory network. *Journal of Hydrology*, 601:126670.
- Li S, Chen Y, Wei W, Fang G, Duan W (2024). The increase in extreme precipitation and its proportion over global land. *Journal of Hydrology*, 628:130456.
- Li Y, Xu J, Anastasiu DC (2023). An extreme-adaptive time series prediction model based on probability-enhanced lstm neural networks. *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 8684–8691.
- Li Y, Yu R, Shahabi C, Liu Y (2017). Diffusion convolutional recurrent neural network : Data-driven traffic forecasting. *arXiv preprint arXiv :1707.01926*.
- Lou G, Shi H (2020). Face image recognition based on convolutional neural network. *China communications*, 17(2):117–124.
- Lundervold AS, Lundervold A (2018). An overview of deep learning in medical imaging focusing on mri. *arXiv preprint arXiv :1811.10052*.
- Ma C, Jiao H, Yeh TCJ, Zhu J, Hao H, Lu J, Dong J, Hao Y (2024a). Simulation of spring discharge using deep learning, considering the spatiotemporal variability of precipitation. *Simulation*, 20:1.

- 
- Ma Y, Leonarduzzi E, Defnet A, Melchior P, Condon LE, Maxwell RM (2024b). Water table depth estimates over the contiguous united states using a random forest model. *Groundwater*, 62(1): 34–43.
- Mariethoz G, Renard P, Straubhaar J (2010). The direct sampling method to perform multiple-point geostatistical simulations. *Water Resources Research*, 46(11).
- Mariethoz G, Straubhaar J, Renard P, Chugunova T, Biver P (2015). Constraining distance-based multipoint simulations to proportions and trends. *Environmental Modelling & Software*, 72:184–197.
- Miville F (2020). Modélisation stochastique du bassin versant karstique de la source du betteraz (ju). Mémoire de maîtrise, CHYN [Centre d'hydrogéologie et de géothermie]-Université de Neuchâtel.
- Mo S, Zabarar N, Shi X, Wu J (2019). Deep autoregressive neural networks for high-dimensional inverse problems in groundwater contaminant source identification. *Water Resources Research*, 55(5):3856–3881.
- Montáns FJ, Chinesta F, Gómez-Bombarelli R, Kutz JN (2019). Data-driven modeling and learning in science and engineering. *Comptes Rendus Mécanique*, 347(11):845–855.
- Mullachery V, Khera A, Husain A (2018). Bayesian neural networks. *arXiv preprint arXiv :1801.07710*.
- Naidoo V, Du S (2022). A deep learning method for the detection and compensation of outlier events in stock data. *Electronics*, 11(21):3465.
- Neven A, Schorpp L, Renard P (2022). Stochastic multi-fidelity joint hydrogeophysical inversion of consistent geological models. *Frontiers in Water*, 4:989440.
- Newendorp PD (1976). Decision analysis for petroleum exploration.
- Opoku PA, Shu L, Ansah-Narh T, Banahene P, Yao KBMO, Kwaw AK, Niu S (2024). Prediction of karst spring discharge using lstm with bayesian optimisation hyperparameter tuning : a laboratory physical model approach. *Modeling Earth Systems and Environment*, 10(1):1457–1482.
- Oriani F, Straubhaar J, Mariethoz G, Renard P (2014). Spatial rainfall simulation : Trading time for space with multiple point statistics. *AGU Fall Meeting Abstracts*, volume 2014, pages H21H–0808.
- Paradis D, Ballard JM, Lefebvre R, Savard MM (2017). Multi-scale nitrate transport in a sandstone aquifer system under intensive agriculture. *Hydrogeology Journal*, 26(2):511–531.
- Paradis D, Vigneault H, Lefebvre R et al. (2016). Groundwater nitrate concentration evolution under climate change and agricultural adaptation scenarios : Prince edward island, canada. *Earth System Dynamics*, 7(1):183–202.
- Patra SR, Chu HJ, Tatas (2023). Regional groundwater sequential forecasting using global and local lstm models. *Journal of Hydrology : Regional Studies*, 47:101442.
- Perrin J (2003). *A conceptual model of flow and transport in a karst aquifer based on spatial and temporal variations of natural tracers*. Thèse de doctorat, Université de Neuchâtel.

- 
- Pltonykova H, Koeppel S, Bernardini F, Tiefenauer-Linardon S, de Strasser L, Connor R (2020). The united nations world water development report 2020 : Water and climate change.
- Pözl A, Blaschke AP, Komma J, Farnleitner AH, Derx J (2024). Transformer versus lstm : A comparison of deep learning models for karst spring discharge forecasting. *Water Resources Research*, 60(4):e2022WR032602.
- Popel M, Bojar O (2018). Training tips for the transformer model. *arXiv preprint arXiv :1804.00247*.
- Preston FW, Henderson J (1964). *Fourier series characterization of cyclic sediments for stratigraphic correlation*. Kansas Geological Survey.
- Previati A, Crosta G (2024). On groundwater flow and shallow geothermal potential : A surrogate model for regional scale analyses. *Science of the Total Environment*, 912:169046.
- Pyrcz MJ, Deutsch CV (2014). *Geostatistical reservoir modeling*. Oxford University Press, USA.
- Radford A (2018). Improving language understanding by generative pre-training.
- Raissi M, Perdikaris P, Karniadakis GE (2017a). Physics informed deep learning (part i) : Data-driven solutions of nonlinear partial differential equations. *arXiv preprint arXiv :1711.10561*.
- Raissi M, Perdikaris P, Karniadakis GE (2017b). *Physics Informed Deep Learning (Part II) : Data-driven Discovery of Nonlinear Partial Differential Equations*. <https://arxiv.org/abs/1711.10566>.
- Remmal Y, Hudon C, Hamilton P, Rondeau M, Gagnon P (2017). Forecasting the magnitude and composition of phytoplankton blooms in a eutrophic lowland river (rivière yamaska, que., canada). *Canadian Journal of Fisheries and Aquatic Sciences*, 74(8):1298–1311.
- Sadoff C, Muller M et al. (2007). Better water resources management : Greater resilience today, more effective adaptation tomorrow.
- Sahoo BB, Jha R, Singh A, Kumar D (2019). Long short-term memory (lstm) recurrent neural network for low-flow hydrological time series forecasting. *Acta Geophysica*, 67(5):1471–1481.
- Sak H, Senior AW, Beaufays F et al. (2014). Long short-term memory recurrent neural network architectures for large scale acoustic modeling. *Interspeech*, volume 2014, pages 338–342.
- Sakizadeh M, Mohamed MM, Klammler H (2019). Trend analysis and spatial prediction of groundwater levels using time series forecasting and a novel spatio-temporal method. *Water Resources Management*, 33:1425–1437.
- Scarselli F, Gori M, Tsoi AC, Hagenbuchner M, Monfardini G (2008). The graph neural network model. *IEEE Transactions on Neural Networks*, 20(1):61–80.
- Scarselli F, Tsoi AC (1998). Universal approximation using feedforward neural networks : A survey of some existing methods, and some new results. *Neural networks*, 11(1):15–37.
- Seo Y, Defferrard M, Vandergheynst P, Bresson X (2018). Structured sequence modeling with graph convolutional recurrent networks. *Neural Information Processing : 25th International Conference, ICONIP 2018, Siem Reap, Cambodia, December 13-16, 2018, Proceedings, Part I*, Springer, pages 362–373.
- Sezen C, Bezak N, Bai Y, Šraj M (2019). Hydrological modelling of karst catchment using lumped conceptual and data mining models. *Journal of Hydrology*, 576:98–110.

- 
- Shapiro AM, Day-Lewis FD (2022). Reframing groundwater hydrology as a data-driven science. *Groundwater*, 60(4):455–456.
- Shleifer S, McCreery C, Chitters V (2019). Incrementally improving graph wavenet performance on traffic prediction. *arXiv preprint arXiv :1912.07390*.
- Siegel DI, Hinchey EJ (2019). Big data and the curse of scale. *Ground Water*, 57(4).
- Solgi R, Loaiciga HA, Kram M (2021). Long short-term memory neural network (lstm-nn) for aquifer level time series forecasting using in-situ piezometric observations. *Journal of Hydrology*, 601: 126800.
- Solomatine DP, Ostfeld A (2008). Data-driven modelling : some past experiences and new approaches. *Journal of hydroinformatics*, 10(1):3–22.
- Stevanović Z (2018). Global distribution and use of water from karst aquifers. *Geological Society, London, Special Publications*, 466(1):217–236.
- Stevanović Z (2019). Karst waters in potable water supply : a global scale overview. *Environmental Earth Sciences*, 78(23):662.
- Straubhaar J, Renard P (2021). Conditioning multiple-point statistics simulation to inequality data. *Earth and Space Science*, 8(5):e2020EA001515.
- Strokach A, Becerra D, Corbi-Verge C et al. (2020). Fast and flexible protein design using deep graph neural networks. *Cell Systems*, 11(4):402–411.e4.
- Sun AY, Jiang P, Mudunuru MK, Chen X (2021). Explore spatio-temporal learning of large sample hydrology using graph neural networks. *Water Resources Research*, 57(12):e2021WR030394.
- Taccari ML, Wang H, Nuttall J, Chen X, Jimack PK (2024). Spatial-temporal graph neural networks for groundwater data. *Scientific Reports*, 14(1):24564. DOI :10.1038/s41598-024-75385-20.
- Takafuji EHdM, Rocha MMd, Manzione RL (2019). Groundwater level prediction/forecasting and assessment of uncertainty using sgs and arima models : a case study in the bauru aquifer system (brazil). *Natural Resources Research*, 28(2):487–503.
- Tartakovsky AM, Marrero CO, Perdikaris P et al. (2020). Physics-informed deep neural networks for learning parameters and constitutive relationships in subsurface flow problems. *Water Resources Research*, 56(5):e2019WR026731.
- Trunz C (2017). Modélisation du bassin versant du betteraz (ju) dans le cadre du projet chlorokarst. Mémoire de maîtrise, CHYN [Centre d'hydrogéologie et de géothermie]-Université de Neuchâtel.
- Van Den Oord A, Dieleman S, Zen H, Simonyan K, Vinyals O, Graves A, Kalchbrenner N, Senior A, Kavukcuoglu K et al. (2016). Wavenet : A generative model for raw audio. *arXiv preprint arXiv :1609.03499*, 12.
- Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I (2017). Attention is all you need. *Advances in neural information processing systems*, 30.
- Vonk MA, Collenteur RA, Panday S, Schaars F, Bakker M (2024). Time series analysis of nonlinear head dynamics using synthetic data generated with a variably saturated model. *Groundwater*.

- 
- Vuilleumier C, Jeannin PY, Hessenauer M, Perrochet P (2021). Hydraulics and turbidity generation in the milandre cave (switzerland). *Water Resources Research*, 57(8):e2020WR029550.
- Vuilleumier C, Jeannin PY, Perrochet P (2019). Physics-based fine-scale numerical model of a karst system (milandre cave, switzerland). *Hydrogeology Journal*, 27(7):2347–2363.
- Wang N, Zhang D, Chang H, Li H (2020a). Deep learning of subsurface flow via theory-guided neural network. *Journal of Hydrology*, 584:124700.
- Wang Q, Ma Y, Zhao K, Tian Y (2020b). A comprehensive survey of loss functions in machine learning. *Annals of Data Science*, pages 1–26.
- Watkins CJCH et al. (1989). Learning from delayed rewards.
- Wu H, Qiao R (2021). Physics-constrained deep learning for data assimilation of subsurface transport. *Energy and AI*, 3:100044.
- Wu Z, Pan S, Long G, Jiang J, Zhang C (2019). Graph wavenet for deep spatial-temporal graph modeling. *arXiv preprint arXiv :1906.00121*.
- Wunsch A, Liesch T, Broda S (2018). Forecasting groundwater levels using nonlinear autoregressive networks with exogenous input (narx). *Journal of Hydrology*, 567:743–758.
- Wunsch A, Liesch T, Broda S (2021). Groundwater level forecasting with artificial neural networks : A comparison of long short-term memory (lstm), convolutional neural networks (cnns), and non-linear autoregressive networks with exogenous input (narx). *Hydrology and Earth System Sciences*, 25(3):1671–1687.
- Wunsch A, Liesch T, Cinkus G, Ravbar N, Chen Z, Mazzilli N, Jourde H, Goldscheider N (2022). Karst spring discharge modeling based on deep learning using spatially distributed input data. *Hydrology and Earth System Sciences*, 26(9):2405–2430.
- Xu C, Misra S, Srinivasan P, Ma S (2019). When petrophysics meets big data : What can machine do? *SPE Middle East oil and gas show and conference*, SPE, D041S038R002 pages.
- Xu K, Hu W, Leskovec J, Jegelka S (2018). How powerful are graph neural networks? *arXiv preprint arXiv :1810.00826*.
- Xue L (2020). mt5 : A massively multilingual pre-trained text-to-text transformer. *arXiv preprint arXiv :2010.11934*.
- Yan B, Harp DR, Chen B, Pawar R (2022). A physics-constrained deep learning model for simulating multiphase flow in 3d heterogeneous porous media. *Fuel*, 313:122693.
- Yin W, Fan Z, Tangdamrongsub N, Hu L, Zhang M (2021). Comparison of physical and data-driven models to forecast groundwater level changes with the inclusion of grace—a case study over the state of victoria, australia. *Journal of Hydrology*, 602:126735.
- Ying R, He R, Chen K et al. (2018). Graph convolutional neural networks for web-scale recommender systems. *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 974–983.
- Yongkai A, Xueman Y, Wenxi L et al. (2022). An improved bayesian approach linked to a surrogate model for identifying groundwater pollution sources. *Hydrogeology Journal*, 30(2):601–616.

- 
- Yoon H, Hyun Y, Ha K, Lee KK, Kim GB (2016). A method to improve the stability and accuracy of ann-and svm-based time series models for long-term groundwater level predictions. *Computers & geosciences*, 90:144–155.
- Yu F, Koltun V (2016). *Multi-Scale Context Aggregation by Dilated Convolutions*. <https://arxiv.org/abs/1511.07122>.
- Zhang J, Zhu Y, Zhang X, Ye M, Yang J (2018). Developing a long short-term memory (lstm) based model for predicting water table depth in agricultural areas. *Journal of hydrology*, 561:918–929.
- Zhang W, Duan L, Liu T, Shi Z, Shi X, Chang Y, Qu S, Wang G (2024). A hybrid framework based on lstm for predicting karst spring discharge using historical data. *Journal of Hydrology*, 633:130946.
- Zhao L, Song Y, Zhang C et al. (2019). T-gcn : A temporal graph convolutional network for traffic prediction. *IEEE Transactions on Intelligent Transportation Systems*, 21(9):3848–3858.
- Zhou J, Cui G, Hu S, Zhang Z, Yang C, Liu Z, Wang L, Li C, Sun M (2020). Graph neural networks : A review of methods and applications. *AI open*, 1:57–81.
- Zuercher L (2018). Modélisation numérique des écoulements et du transport de polluants dans le système karstique de la source du betteraz (ju). Mémoire de maîtrise.