





Deep Learning Modeling of Groundwater Pollution Sources

Yiannis N. Kontos^{1,2} ^(✉), Theodosios Kassandra² ,
Konstantinos L. Katsifarakis¹ , and Kostas Karatzas² 

¹ School of Civil Engineering, Aristotle University of Thessaloniki, 54124 Thessaloniki, Greece
ykontos@civil.auth.gr

² School of Mechanical Engineering, Aristotle University of Thessaloniki,
54124 Thessaloniki, Greece

Abstract. This research aims at optimizing the monitoring network used to consistently identify pollution's origin in the pollution source identification problem in groundwater hydraulics under real-time/operational applications. For this task, Machine Learning (ML) and Deep Learning (DL) methods are introduced, which can outperform metaheuristics, such as Genetic Algorithms (GAs), in terms of total computational load. To test the approach, a theoretical aquifer with two pumping wells is studied, where one of six possible pollution sources may spread a conservative pollutant. An existing own software simulates a 2D surrogate steady state flow field, using particle tracking to simulate advective mass transport only. A large number of combinations of possible source locations (4 different layout scenarios), hydraulic gradients and pumping wells' flow-rates is used to calculate various features (such as pollutant arrival times, hydraulic drawdowns) in a 29×29 grid. Three ML/DL methods (Random Forests, Multi-Layer Perceptron, Convolutional Neural Networks) are tested for prediction accuracy, while Correlation based Feature Selection (CFS), and targeted tests are used to select subsets/sampling frequencies that can provide similar accuracy with the full datasets. This evaluation process bears promising results and paves the way for monitoring network optimization.

Keywords: Machine Learning · Source Identification · Monitoring network

1 Introduction

Cost of groundwater pollution control and remediation methods, like pump-and-treat or hydraulic control, is high; its minimization is essential yet challenging [1, 2]. Timely pollution detection presupposes existence of a proper network of monitoring wells, also necessary to identify the pollution source and apply “the polluter pays” principle [3], discouraging possible polluters. For a given monitoring scheme, source detection is an optimization problem solved with various metaheuristic [4] or Machine Learning (ML) methods [5]. Lately, water resources scientists have started to utilize Deep Learning (DL) research; Shen [6] provides a great review. Zhang et al. [7] go a step further considering surrogate models' approximation error, and minimizing it. The installation/operation costs of the monitoring network may be quite high, and its minimization is essential,

hence many researchers have tried to deal with optimal network design [8]. This paper aims at implementing ML/DL methods, already proven robust at environmental engineering problems [9], to solve the source detection problem, and investigate minimization of the number (N_r) of monitoring wells and sampling frequency.

1.1 Theoretical Problem Definition

An aquifer of known characteristics is studied (Fig. 1). Two pumping wells (PWs) near the southern boundary provide irrigation/drinking water, defining the flow field together with a North-South natural flow. Six suspected possible sources (S1–S6), capable of instantaneous leakage, may spread a specific pollutant, while four scenarios with different 6-source layouts are considered. The short-term goal is to identify the source, while the long-term is to optimize the monitoring design and solve the real-time/operational problem. The $29 \times 29 = 841$ inner field grid nodes (50 m cell size) serve as possible locations of sources, PWs and monitoring wells (MWs). The over-simplified problem version used for this pilot/evaluation approach is based on the assumptions: i) time $t = 0$ of initial pollution leakage is known, regardless of the source, ii) all sources may leak the same pollutant, only one at a time, instantly (during a time-step), iii) initially, all the nodes, bear MWs, that are used to locally record the following: a) yes/no pollutant detection, b) 1st day of pollution, c) pollution duration, d) hydraulic head drawdown. All these “measured” features entail construction of a MW at the respective nodes. Features (a) relate to a single sampling at a predefined day and in-situ/ex-situ analyses. Features (b) relate to floating or fixed depth (low-cost, smart) sensors sending a single signal when detecting pollution (e.g., measuring electric current variance) or consecutive manual samplings in each timestep, followed by in-situ/ex-situ analyses, until pollution detection. Features (c) imply continuation of the remote or manual measurements after the first pollution detection. Features (d) entail a manual one-off drawdown measurement (steady state flow) or, in case of an existing sensor, can relate to a fixed depth sensor-strip solution (instead of a floating sensor) that can also measure drawdown.

1.2 Research Goals

Research goals are: Find pollution source for Scenarios 1–4 (plus a merged $4 \times 6 = 24$ sources 5th scenario), using various ML/DL methods and evaluate them regarding their accuracy, exploiting the full MW network and features’ values of the over-simplified initial problem version. Conclude on the useful MWs/features and remove the useless ones, a first indirect step towards the optimization of the MW network. Investigate the importance of each feature trying to further decrease their number (feature selection), with various techniques, depending on the ML/DL method used, simultaneously retaining the same prediction accuracy levels. Given the specific spatial layout of the MW network, investigate further indirect MW network optimization (monitoring cost minimization), searching for the lowest temporal discretization (lowest sampling frequencies) that can provide unchanged source prediction accuracy for all scenarios. Finally, evaluate each ML/DL method, concluding on a) pros/cons, b) the significance of the features tested, c) the formulation of the input-output (X–Y) datasets suitable for each ML/DL method. The ultimate goal is to include the best methods in a metaheuristic optimization algorithm

that will eventually automatically optimize the MW network (MW locations, sampling rates/time-instances or strategies).

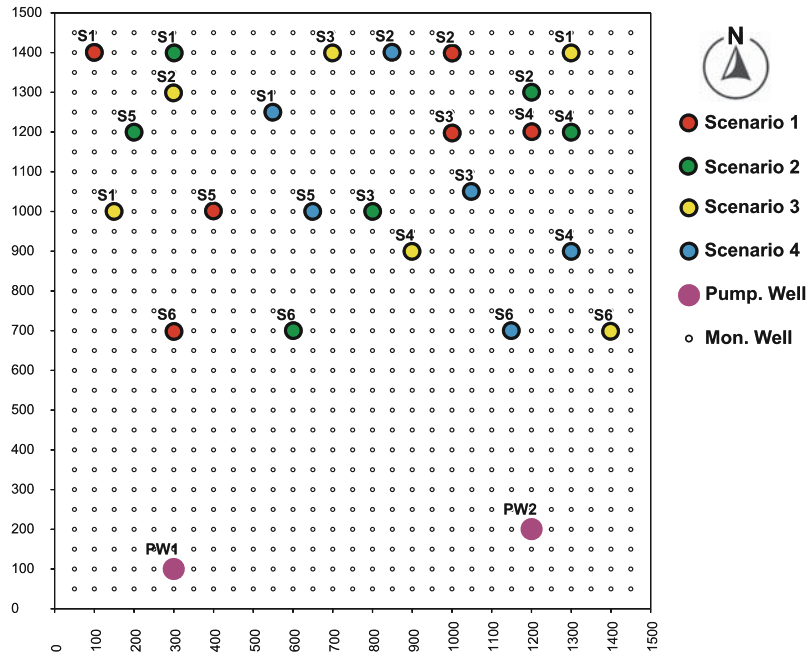


Fig. 1. The 2D theoretical flow field with the possible pollution Sources (S1–S6), the 4 different scenarios/layouts, the PWs and MWs, positioned on a 29×29 50 m-cell-sized grid.

2 Hydraulic Simulation

2.1 Flow Field and Mass Transport Simulation

Balancing the vast computational load needed for the creation of the simulated datasets to feed the data-driven methods, entails simplification of the flow field, hence a surrogate 2D flow field is studied. The $1500 \text{ m} \times 1500 \text{ m}$ theoretical flow field (Fig. 1) includes two PWs of known varying flow-rates during a year. An inter-annually varying North-South natural flow also affects the flow field, while the confined aquifer is assumed to be infinite, homogeneous, isotropic with a plane, horizontal, single-phase steady flow. The combination of the constant/varying values of flow field parameters (Table 1) produces a 15,246 dataset package of cases per scenario.

The pollutant is assumed to spread dominantly via advection, with a Lagrangian Particle Tracking Method (PTM) simulating mass transport [10]. The “MovPo” software used is part of the “OptiManage” optimization suite, created by authors [1]. Their previous research offers guidelines/investigation techniques to define the best parameter values, e.g., the Nr of particles for circular plumes’ simulation, the temporal discretization of the study period, and the suitable PWs’ approximate capture zone [10]. The pumping well pollution criterion is based on a circular approximation of time of travel (during a timestep) capture zones. Practically, a pollutant particle P is assumed to pollute

Table 1. Values, ranges and Nr of values used, concerning flow-field variables/parameters producing the combination of problem instances to be simulated.

Variable	Units	Min	Max	Step	Nr of values
Pollution source (S)	-	1	6	1	6
N-S hydr. grad. (J)	‰	0	2	0.1	21
PW1 flow-rate (Q1)	L/s	215	225	1	11
PW2 flow-rate (Q2)	L/s	225	235	1	11
Thickness (b)	m	50	50	0	1
Hydr. conductivity (K)	m/s	0.0001	0.0001	0	1
Porosity (n)	-	0.2	0.2	0	1
				RUNS =	6·21·11·11 = 15,246

a well W during a certain timestep DT , if and only if the line segment simulating the displacement of P during DT intersects the approximate capture zone of W (i.e., [2]).

The additional simulation burden here is selecting the mathematical criterion of a MW being polluted, hence pollution detected with any sampling method (features a, b, c or d). The point-in-polygon method called “ray casting algorithm” (or “even-odd rule”) [11] is implemented, so that in each timestep (here day), the algorithm checks whether any center of a MW is inside the (moving) polygon defined by the coordinates of the 16 moving particles simulating pollution plume/source (initially circular).

2.2 Building of Datasets

In order to build the datasets for the training/validation/evaluation of the ML/DL algorithms, “MovPo” simulates all 15,246 different source-flow field cases (Table 1), running for 46.5 h (Intel Core i7 7700 @3.60 GHz; 16 GB RAM @ 1197 MHz). Figure 2 presents the merged results of such two random Scenario 1 (Fig. 1) layout cases; N-S grad is 1.6‰; $Q1 = 221$ L/s; $Q2 = 231$ L/s. Left case is a classic PTM graphical representation, where (16) separate particle trajectories are calculated as line segments for consecutive (1-day) timesteps, checked for polluting any PW. Right case represents the added simulation of the pollution front, where a hexadecagon’s consecutive displacements are calculated for consecutive timesteps, checked for polluting any MW. The selected polygons drawn represent only: all days that plume is located upon the node of S4, and first day of pollution for any polluted MW up to its PW2 pumping. Zoom 1 is a left case magnified region presenting the 16 symmetrically placed points (+) on the initially circular pollution plume (S1), that serve as starting points for the particles being tracked. The nodes’ enumeration is shown: node 785 is S1 of the Scenario 1. Zoom 2 is a right case magnified region, showing S4 (node 691) polluted for 35 days (35 polygons over node; Table 2). Day 58, first day node 662 is polluted, is also shown.

Table 2 presents the respective results of Fig. 2 right case, as calculated for run 9,632/15,246. The first 2 columns provide dataset/run info: source Nr; respective node enumeration; PW node Nrs; flow/aquifer parameters; Nr of MWs polluted; PW(s) polluted (here only PW2); Nr of days needed (1703).

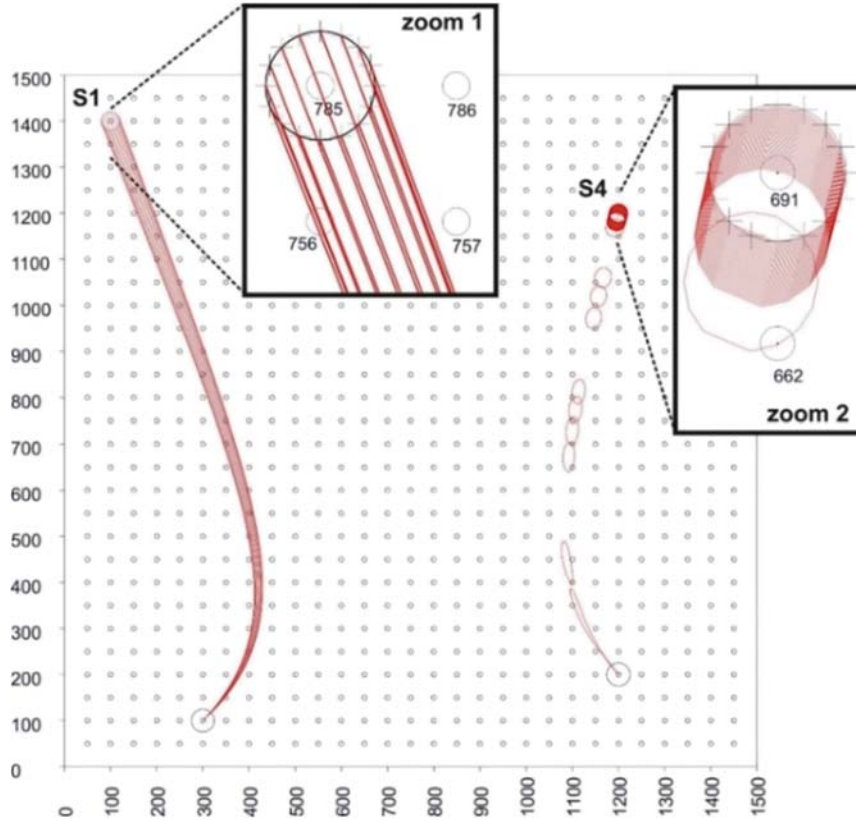


Fig. 2. S1 (left) and S4 (right) merged results (Scenario 1 layout); S1: PTM particle trajectories, for PW pollution criterion; S4: moving pollution front/polygon, for MW pollution criterion.

The ensuing 3 columns present raw non-zero results: “IsMWPol” (short for “Is Monitoring Well Polluted”) includes Nrs of the 10 nodes (representing Source/ MW/PW) polluted in this run/case; “DayPol” (“Day of Pollution”) refers to the first day pollution is detected in the respective node; “DurPol” (“Duration of Pollution”) refers to the duration (days) of each node being polluted.

3 Machine/Deep Learning Implementation - Results

Two basic ML/DL approaches are implemented: Classification (CL) and Computer Vision (CV). In CL, Random Forests (RF [12]) and Multi-Layer Perceptron (MLP [13]) are tested, while in CV, a Convolutional Neural Network (CNN) is tested.

Table 2. Results of Fig. 2 right case as produced by “MovPo” software’s run 9,632/15,246.

RUN info (scenario 1)		IsMWPo	DayPol	DurPol
RUN count	9632/15246	111	1073	1
Source Nr	4	225	995	60
Source node	691	370	778	37
PW1 node	35	399	712	65
PW2 node	111	428	654	68
N-S grad (%)	1.6	457	607	43
Q1 (L/s)	221	545	381	61
Q2 (L/s)	231	574	305	69
Nr MWs polluted	10	603	242	44
DayPol PW1	-	662	58	54
DayPol PW2	1073	691	1	35

3.1 Machine Learning (Classification) - Random Forests, Multi-layer Perceptron

In CL approach, time dimension and spatial correlation of MWs are not considered, hence the simple formulation of the datasets (Type A). Each simulated variable combination (see Table 1) produces a single dataset and is printed in a single line in the results’ file (see Table 3). The dataset presented in Table 2 is actually the 9,632nd dataset of Table 3, for Scenario 1. The target variable (output-Y), as far as the prediction is concerned, is a discrete class, with nominal values from 1 to 6 (source Nr). The 4 types of input-X variables are: a) whether node i is polluted at all (IsMWPo = 1 or 0), b) Nr of days that node i is polluted since the leakage start (DayPol = 0–2500; integer), c) duration of node i pollution (DurPol = 0–2500; integer), and d) hydraulic head drawdown at node i (Drawd ≥ 0 ; real). The initial Type A data package is a $15,246 \times (4 \times (29 \times 29) + 1) = 15,246 \times 3,365$ matrix per scenario (including Scenario 5).

Table 3. Demonstration of the structure of Type A datasets for the Classification approach.

Dataset	input-X												output-Y
	IsMWPo(i)			DayPol(i)			DurPol(i)			Drawd(i)			Source
	$X_i = 0/1$			$X_i = 0-2500$			$X_i = 0-2500$			$X_i = 0-\infty$			$\Upsilon = 1-6$
	1	...	841	1	...	841	1	...	841	1	...	841	
1	X_i	...	X_i	X_i	...	X_i	X_i	...	X_i	X_i	...	X_i	Y_i
2	X_i	...	X_i	X_i	...	X_i	X_i	...	X_i	X_i	...	X_i	Y_i
...
15246	X_i	...	X_i	X_i	...	X_i	X_i	...	X_i	X_i	...	X_i	Y_i

The process to source prediction for each scenario (using WEKA [14]) is:

Step 1-Masking Datasets: For added realism reasons, all data from the following nodes are removed/masked: i) the 9 adjacent to the sources nodes ($9 \times 6 = 54$), ii) all the nodes in the 4 southern grid series where PWs are located (nodes 1 to $29 \times 4 = 116$), decreasing features to 3195.

Step 2-Remove Useless: A simple “RemoveUseless” filter is implemented to the masked subset, for the many blank variables, as the respective nodes are never polluted, regardless of the source inducing useless perplexity to the ML method, resulting to a filtered subset (FL) of approximately 1000 features/scenario.

Step 3-Feature Selection: CL features the advantage of facilitating Correlation-based Feature subset Selection (CFS) [15], indirectly leading to a pseudo-optimization of the monitoring network, minimizing the Nr of MWs (not the sampling frequency), based on the efficiency of identifying the source criterion. Two different search methods are used, Best First and Greedy Stepwise, leading to the respective subsets.

Step 4-ML implementation: ML algorithms, RF (100 trees) and MLP (3 hidden layers; learning rate = 0.3; 500 epochs; Adam optimizer; MSE loss function) are implemented for the 3 subsets (RF-FL, RF-BF, RF-GS, MLP-FL, MLP-BF, MLP-GS; see Table 4), to predict the 1–6 class. Models are evaluated using 10-fold cross validation.

Table 5 and Fig. 3 present the accuracy metrics for all ML methods/scenarios, including Scenario 5 (merged scenarios 1–4). In this enhanced scenario, the dataset package is now sized $60,984 \times 3,365$, while the classes to predict are $4 \times 6 = 24$.

Table 4. Features’ selected subsets (FL, BF, GS) per scenario: Nr of MWs selected; features’ Nr (Type A datasets columns; see Table 3), and feature types (IsMWPOL, DayPol, DurPol, Drawd).

Scenario	Subset	Nr MWs	Nr Feat	IsMWPOL	DayPol	DurPol	Drawd
1	FL	657	1086	143	143	143	657
	BF	6	6	4	2	0	0
	GS	23	55	17	18	20	0
2	FL	658	958	100	100	100	658
	BF	6	6	6	0	0	0
	GS	21	52	21	20	10	0
3	FL	657	1137	160	160	160	657
	BF	6	6	6	0	0	0
	GS	23	53	23	18	12	0
4	FL	657	1122	155	155	155	657
	BF	6	6	5	0	1	0
	GS	21	43	21	10	12	0
5	FL	507	2028	507	507	507	507
	BF	23	26	10	10	6	0
	GS	45	71	30	20	21	0

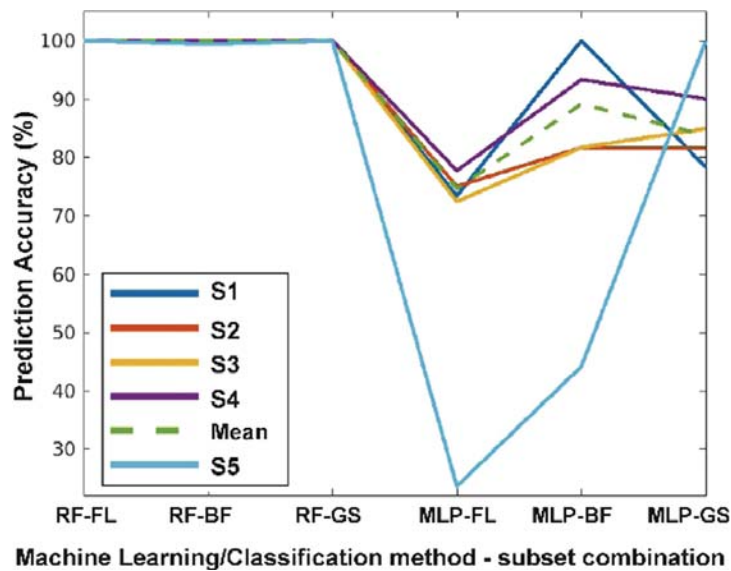


Fig. 3. Prediction accuracy (%) metrics of Classification methods.

Table 5. Pollution source prediction accuracy (%) metrics of machine learning/classification methods (Random Forests and Multiple Perceptron combined with feature selection methods)

Scenario	Accuracy (%) of ML method					
	RF-FL	RF-BF	RF-GS	MLP-FL	MLP-BF	MLP-GS
1	100.00	99.97	100.00	73.38	99.97	78.33
2	100.00	100.00	100.00	75.00	81.65	81.63
3	100.00	100.00	100.00	72.41	81.66	84.90
4	100.00	100.00	100.00	77.69	93.32	90.00
5	99.98	99.41	99.94	23.55	44.06	99.94
Mean 1–4	100.00	99.99	100.00	74.62	89.15	83.72
StDev 1–4	0.000	0.013	0.000	1.999	7.855	4.308

MLP performs poorly in the FL subset, improving with fewer features. RF perform better than MLP, as the problem's structure is "if-else": "if this MW is polluted then the source is that". This is expected as in 3/4 basic scenarios, each MW is polluted by a specific source, while even in Scenario 1, with 2 MWs polluted by 2 different sources, but on different days, RF is nearly 100% accurate. Formulating the basic scenarios 1–4 to be solved by ML methods resembles brute force; a simple monitoring could match observations-sources: 6 MWs manage to monitor 6 sources. This is not the case for the 24-source Scenario 5; increased complexity does not lead to a clear univocal MW-source match: 23 MWs achieve $\approx 100\%$ accuracy. Both CFS methods are capable of selecting the absolutely necessary information, decreasing the features needed to lead to a successful prediction each time. Finally, drawdown measurements are unsurprisingly never selected as important features; the simplified steady state flow field does not vary depending on the source location. At this point, the real-time/operational problem

is solved, with accurate predictions. Indirect optimization of the monitoring network occurs, but only in the spatial dimension. CFS minimizes the Nr of MWs, but there is no control in the temporal dimension.

3.2 Deep Learning (Computer Vision) - Convolutional Neural Networks

In CV approach, both spatial and temporal dimensions are considered, hence the more complex data formulation (Type B). Each simulated variable combination (see Table 1) produces up to 2500 files (days/runs) per simulation (out of 15,246) per scenario. Each file is a 29×29 -sized matrix/frame containing “1” or “0” elements, depending on yes/no pollution of the specific node that specific day. Day 1 frame contains a “1” in the source-node, constituting the target variable (output-Y). All ensuing frames can be used as training data for the DL algorithm (input-X variables). Figure 4 graphically presents the single dataset/run presented in Type A form in Table 2, but in Type B form. The full Type B dataset package is a batch of $15,246 \text{ folders} \times 2500 \text{ files} = 38,115,000$ files/frames for each of the 5 scenarios.

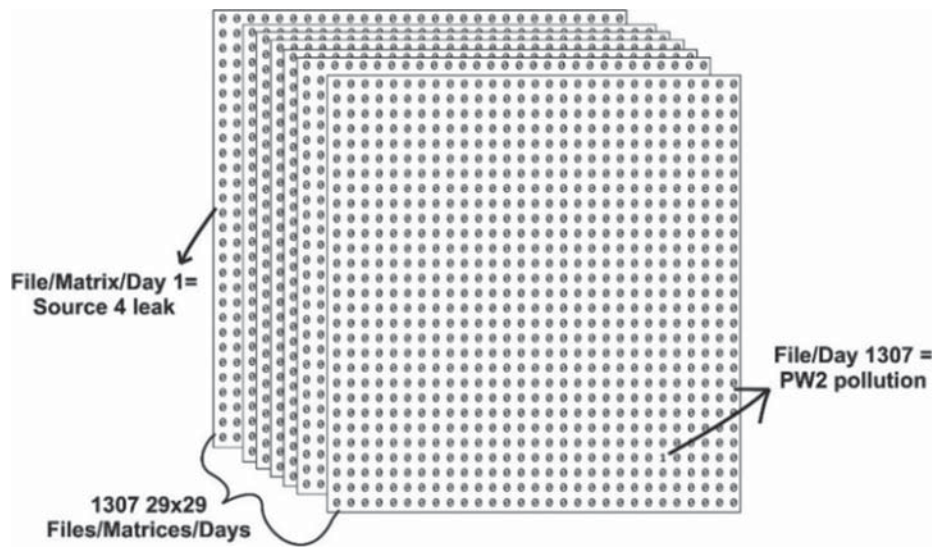


Fig. 4. Results of Fig. 2 right case (see Table 2), in Type B data formulation used in CV approach (batch of up to 2500, here 1307, 29×29 frames), instead of single-line Type A form (Table 3).

The process to source prediction for each scenario (using Google Colab [16]) is:

Step 1-Datasets’ Temporal Masking: Not all produced 2500 frames per simulation are fed into the CNN. The leading frames, of days when source-nodes are polluted, are concealed (t1 to t51, t49, t48, t47 for Scenarios 1, 2, 3, 4, respectively), as well the latter frames, when PWs are polluted (t469, t782, t327, t310 to t2500 for Scenarios 1,2,3,4, respectively). The strictest common time window t52–t309 (258 frames) is selected for all simulations, for realism/uniformity/comparability reasons. As a result, input-X variables actually comprise of the sum of all available frames; all available matrices are added together creating a super-frame/image (see Fig. 5). 29×29 frames are cropped into 28×28 ones, for proper CNN operation (29^{th} column/series deleted).

Step 2-Time-variant datasets: Type B data formulation accommodates data manipulation in the temporal dimension: different versions of datasets are created by summing the frames every 1, 10, 20, ..., 80 days (9 time-variant data packages/scenario). Each image version corresponds to a different sampling (manual or sensors-telemetry based) frequency, enter indirect optimization of the time-dimension of the monitoring network.

Step 3-Import CFS subsets from CL: Feature subsets (MWs to be constructed) selected by CFS methods in CL, are used to indirectly introduce spatial optimization of monitoring network in the temporal optimization of CV (Step 2). The use of subsets entails the respective masking of the finalized super-images (unselected elements in the matrices are given zero values). Finally, 3 subsets (full dataset CNN+ subsets CNN-BF and CNN-GS) combined with 9 temporal variations produce 27 datasets/scenario.

Step 4-Implementation: For each one of the 27 datasets a CNN is implemented, using the U-Net segmentation architecture (Adam optimizer; learning rate = 0.0001; binary cross entropy loss function; 50 epochs; batch size = 32; see Fig. 6) [17]. The models are evaluated with the train test split method (60% train - 40% test), while accuracy is used as a performance metric. Results are presented in Table 6 and Fig. 7.

CNN in full data mode, meaning a large costly monitoring network, exhibits exquisite performance, even with high frequencies of sampling (CNN per 50-day = 98% accuracy). Combining BF subsets from CL with CNN leads to disappointing results, while CNN-GS combination is promising. CNN-GS exhibits >70% min accuracy most of the time for frequencies up to 50 days, in the basic Scenarios 1–4; not the case in Scenario 5, where the CFS subsets do not lead to accepted accuracies even for daily monitoring.

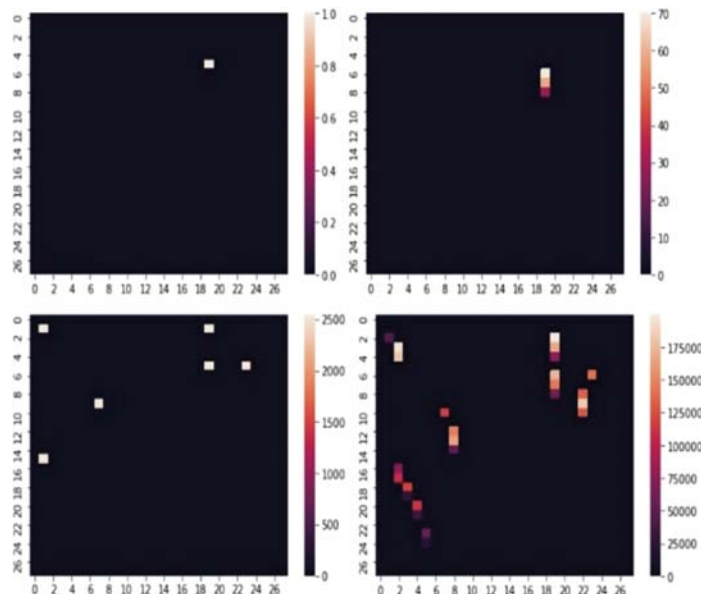


Fig. 5. Graphical representation of output-Y (target) input-X variables: a) Y for S3, b) X for S3, c) Y for all 6 sources, d) X for all 6 sources (Scenario 1, step = 1 day, t52-t309).

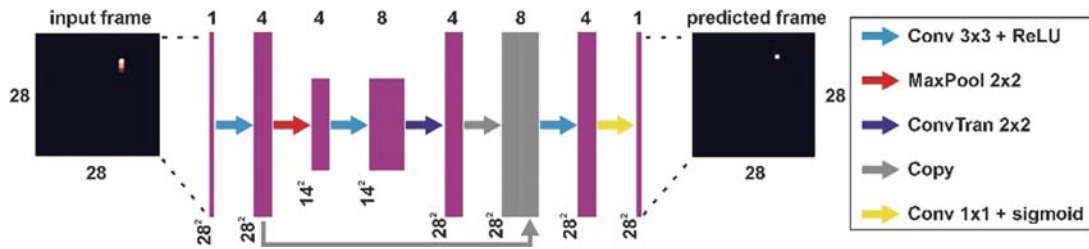


Fig. 6. The U-Net segmentation architecture used in the Computer Vision approach.

Table 6. CNN accuracy (%) for various (9) temporal discretizations and (3) spatial MW distributions (full monitoring network: CNN, BF/GS-optimized network: CNN-BF/CNN-GS).

Scenario	Freq (days) =	1	10	20	30	40	50	60	70	80
	Subset	Accuracy (%) in predicting pollution source								
1	CNN	100	100	100	100	100	98	89	60	76
	CNN-BF	17	17	16	17	17	15	16	0	17
	CNN-GS	91	82	79	74	86	92	86	54	60
2	CNN	100	100	100	100	100	100	94	85	80
	CNN-BF	17	17	17	17	17	0	0	13	17
	CNN-GS	88	86	83	83	82	71	62	64	80
3	CNN	83	84	83	100	100	100	100	95	82
	CNN-BF	0	0	0	0	0	0	0	0	0
	CNN-GS	41	70	70	82	82	58	64	68	91
4	CNN	100	99	99	89	87	87	85	79	69
	CNN-BF	41	40	40	40	40	40	34	0.1	20
	CNN-GS	74	73	74	73	72	70	57	19	31
5	CNN	96	98	97	96	92	94	80	67	62
	CNN-BF	11	21	23	19	19	17	16	15	18
	CNN-GS	25	30	34	23	32	21	28	17	24

4 Conclusions

The complex inverse groundwater hydraulics problem of pollution source identification is successfully addressed via ML/DL implementations. In a given groundwater flow field, for a given monitoring network, namely known MWs’ locations/sampling rates (manual or sensor-based), RF and CNN perform excellently. This leads to the first future research step: Replace hydraulic (flow field and mass transport) simulation with RF/CNN to minimize computational load, so that a real-time operational application or a Decision Support System controlling/supporting a monitoring network scheme can be faster, yet equally accurate at predicting pollution origin against pre-registered suspicious sources. Current research also proposes the use of the CFS feature selection method, combined

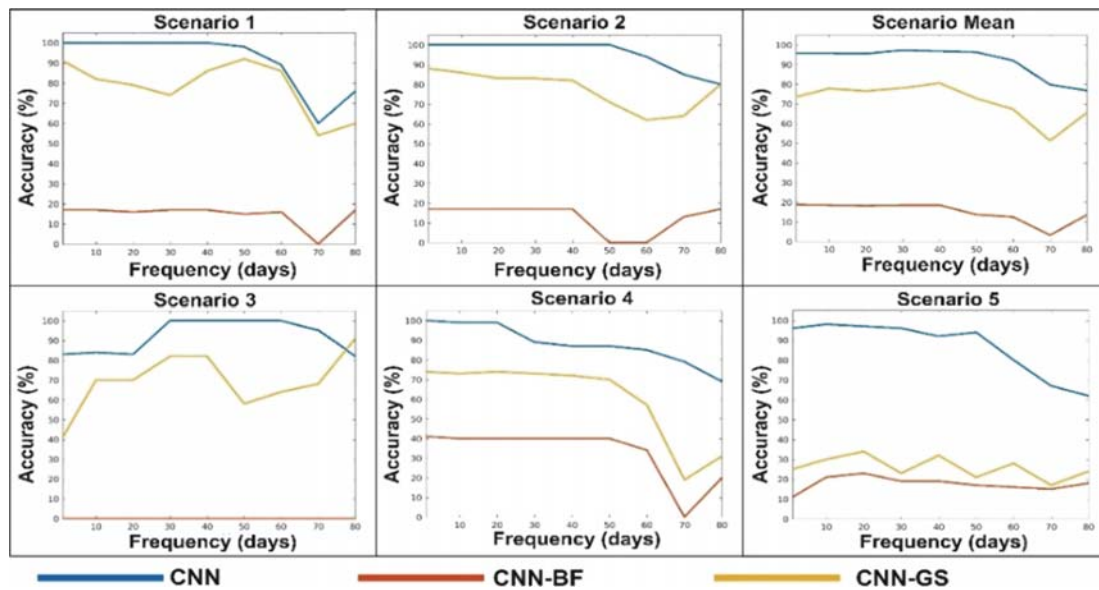


Fig. 7. Diagrams of CNN accuracy (%) for various temporal discretizations and spatial MW distributions (full monitoring network: CNN, BF/GS-optimized network: CNN-BF/CNN-GS).

with the proposed data formulation, to introduce indirect spatial optimization of the monitoring network. The fact that CNN structure facilitates search of time-variant feature subsets, conducted here by trial-and-error tests, ultimately leads to an indirect spatial and temporal optimization of the monitoring network/schedule. For example, the managing authority of the pseudo-realistic Scenario 1 case could create at least two monitoring network strategies: a) only with the 6 MWs and a monitoring sampling frequency of 1 day (RF-BF) achieving 99.97% prediction accuracy, or b) with 23 MWs and monitoring sampling frequency of 50 days (CNN-GS) achieving 92% accuracy, within seconds (if sensors and in-situ analysis is used) or hours (if manual sampling and ex-situ analysis is used) after 309 days since the leak incident. The real breakthrough is that these promising pilot implementations pave the way to the next step of controlling the masking of time/space dimensions (testing various subsets) with metaheuristic methods (e.g., GAs) for CNN training/validation to directly optimize the monitoring network/schedule, minimizing the respective cost function.

Acknowledgments. This research is co-financed by Greece and the European Union (European Social Fund- ESF) through the Operational Programme «Human Re-sources Development, Education and Lifelong Learning 2014-20» in the context of the project “Evolution of Computational Intelligence in Environmental Engineering-Generalization, Improvement, Optimal Combination of Methodologies in Air Quality & Water Resources Problems” (MIS 5052163).

References

1. Kontos, Y., Katsifarakis, K.: Optimization of management of polluted fractured aquifers using genetic algorithms. *Eur. Water* **40**, 31–42 (2012)
2. Kontos, Y., Katsifarakis, K.: Optimal management of a theoretical coastal aquifer with combined pollution and salinization problems, using Genetic Algorithms. *Energy* **136**, 32–44 (2017)

3. European Parliament and Council: Directive 2004/35/CE of the E.P. and E.C. of 21-4-2004 on environmental liability with regard to the prevention and remedying of environmental damage, OJ L 143, 30.4.2004, pp. 56–75 (current version 26/06/2019) (2004)
4. Han, K., et al.: Application of a genetic algorithm to groundwater pollution source identification. *J. Hydrol.* **589**, 125343 (2020)
5. Mo, S., Zabarar, N., Shi, X., Wu, J.: Deep autoregressive neural networks for high-dimensional inverse problems in groundwater contaminant source identification. *Water Resour. Res.* **55**(5), 3856–3881 (2019)
6. Shen, C.: A transdisciplinary review of deep learning research and its relevance for water resources scientists. *Water Resour. Res.* **54**(11), 8558–8593 (2018)
7. Zhang, J., Zheng, Q., Chen, D., Wu, L., Zeng, L.: Surrogate-based Bayesian inverse modeling of the hydrological system: an adaptive approach considering surrogate approximation error. *Water Resour. Res.* **56**(1), 1–25 (2020). <https://doi.org/10.1029/2019WR025721>
8. Janardhanan, S., et al.: Optimal design and prediction-independent verification of groundwater monitoring network. *Water* **12**(1), 123 (2020)
9. Bagkis, E., Kassandros, T., Karteris, M., Karteris, A., Karatzas, K.: Analyzing and improving the performance of a particulate matter low cost air quality monitoring device. *Atmosphere* **12**(2), 251 (2021)
10. Anderson, M., Woessner, W., Hunt, R.: Particle tracking, Chap. 8. In: Anderson, M., Woessner, W., Hunt, R. (eds.) *Applied Groundwater Modeling*, 2nd edn., pp. 331–373. Academic Press, San Diego (2015)
11. Shimrat, M.: Algorithm 112: position of point relative to polygon. *Commun. ACM* **5**(8), 434 (1962)
12. Breiman, L.: Random Forests. *Mach. Learn.* **45**, 5–32 (2001)
13. McCulloch, W., Pitts, W.: A logical calculus of the ideas immanent in nervous activity. *Bull. Math. Biol.* **5**, 115–133 (1943)
14. Frank, E., Hall, M., Witten, I.: The WEKA workbench. Online Appendix for “Data Mining: Practical Machine Learning Tools and Techniques”, 4th edn. Morgan Kaufmann (2016)
15. Hall, M.: Correlation-based feature selection for machine learning. Ph.D. Thesis, Department of Computer Science, University of Waikato Hamilton (2000)
16. Bisong, E.: Google Colaboratory, Building Machine and Deep Learning Models on Google Cloud Platform: A Comprehensive Guide for Beginners, pp. 59–64. Apress, Berkeley (2019)
17. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) *MICCAI 2015*. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24574-4_28