SN Computer Science (2022) 3:10
https://doi.org/10.1007/s42979-021-00856-6

ORIGINAL RESEARCH

# Hadoop–Spark Framework for Machine Learning-Based Smart Irrigation Planning

**Asmae El Mezouari**[1] · **Abdelaziz El Fazziki**[1] · **Mohammed Sadgal**[1]

Received: 3 December 2020 / Accepted: 6 September 2021 / Published online: 23 October 2021


## Abstract

Up-to-date, given the expanding increase of the population and the development of human daily lifestyles, the expenditure of freshwater resources increments progressively. It appears that there is a need to optimize at least the consumption of fresh water in agriculture. For this reason, novel various irrigation technologies have been deployed in this context like drip irrigation, flood irrigation, and decision support systems to come up with the constraints of climate changes that decrease the water availability but it is still limited. Therefore, the majority of researchers are working until today on automating the irrigation systems. These smart systems rely mainly on the advances of information technologies like the internet of things, big data, and machine learning for aligning irrigations with climatic changes. Besides, integrating the predictive process helps in anticipating and adapting to the climatic constraints in agriculture, using meticulous soil and environment dependencies analysis based on features' prediction. In this paper, we enriched our proposed flexible online learning (OL) framework designed for promoting irrigation decisions based on soil characteristics analysis and prediction. We shed the light on a comparative study of four predictive methods, in particular, the auto-regressive moving average, the eXtreme Gradient Boosting, the random forest, and the deep artificial neural networks implemented inside the Hadoop/Spark environment to predict the humidity of the soil, relying on soil temperature and time in several depths. In the end, we discussed the precision of these models in various conditions.

**Keywords** Irrigation planning · Time series · Hadoop · Spark · Machine learning · Big data

## Introduction

Referring to [1], it appears that in developing countries the agriculture represents the essential source for many products and food employed by a multitude of organisms to keep and improve their lives, Their traditional methods' freshwater use attains 85% for this purpose, managing limited freshwater appears as a challenging release in agricultural activities specifically when the water needs escalate at a fast pace, like in African countries, the crucial increase of population lowers the water availability, without forgetting that the majority of these countries are in semi-arid zones characterized with unstable rainfall, and long drought periods which reflects negatively on the crop yield. Appropriately, the irrigation task becomes very challenging for the farmer because of the water resource depletion.

Thus, different irrigation techniques have been employed to minimize the use of freshwater resources in agriculture, like flood irrigation that relies on covering the field with water, drip irrigation that decreases water distribution much better through a direct water supply to the root zone or delivering water to the soil surface over valves, pressure lines, and emitters, sprinkler irrigation that applies irrigation identical to natural rainfall through pumping using a system of pipes. Besides, it is perceived that there is a need for powerful resolutions are mandatory to outstrip this restriction. For this reason, researchers have been performing many advancements studies and achievements to invent novel water resources management systems benefiting from the recent advances on the internet of things (IoT), big data, and machine learning (ML). Thus, to support decision-making

This article is part of the topical collection "Advances on Signal Image Technology and Internet based Systems" guest edited by Albert Dipanda, Luigi Gallo and Kokou Yetongnon".

✉ Asmae El Mezouari
asmae.elmezouari@ced.uca.ma

[1] Computer Sciences Department, LISI, University of Marrakech Cadi Ayyad, Marrakech, Morocco

in the drip irrigation system, a smart system relying on soil image processing, field sensing, and mobile technologies to boost the irrigation planning by determining the amount of water required for the plants' cultivation [2]. Likewise, some researchers in [3] proposed an automatic irrigation system that employs a GPRS module and wireless sensor network to optimize up to 90% of water use in comparison to traditional techniques. Furthermore, it appears that the most difficult task of researchers who use machine learning is the selection of the best model, data samples, and the corresponding data sets. Hence to come up with these issues, it is required to test different machine learning algorithms and training models with various data samples to determine the best one for an efficient prediction with the highest accuracy that is assumed by our proposed framework in our previous work in [4], in which we computed three forecasting methods specifically the ARIMA, the XGBoost, and the random forest.

Moreover, this paper provides an advanced online framework to implement more methods in soil features in different depths to deal with an accurate predictive model that anticipates soil parameter changes and better irrigation planning in agriculture standing on supervised learning, especially the deep artificial neural networks.

The remainder of this paper is composed as follows: the next section epitomizes a brief survey about the last smart irrigation systems and soil features forecasting facilities. The third section shows the followed methodology. The fourth section depicts the suggested framework architecture. The fifth section outlines a case study of predicting soil moisture and the forecasting models implemented in our frameworks such as the auto-regressive moving average, the random forest, the eXtreme Gradient Boosting, and the deep artificial neural networks. As considerably, this paper discussed the performance analysis of the tested forecasting methods. The last section, encapsulate the prediction results and synthesizes a comparison of the outcomes of these predictive models. In the end, we conclude by examining the limits, advantages, and potential perspectives of this work.

## Related Works

In the view of the past few decades, numerous predictive researches have been carried out to improve the efficiency in water resource supervision relying on the advances of machine learning algorithms and smart systems. Some research among them is focused on predicting actual evapotranspiration from time series analysis such as implementing various machine-learning methods on three types of evapotranspiration models with different input data. Among these methods, the application of M5P regression trees, bagging, random forests, and regression support vectors to data from an experimental site in Central Florida according to [5]. Likewise, a comparison of two types of streamflow modeling was performed [6] using machine learning algorithms. The first one is based only on climatic data (precipitation, temperatures, and potential evapotranspiration), the second one integrates also the previous flows in the data entrees. Many predictive models were tested to predict the river flows such as the multiple linear regression, the TUW hydrological model, the eXtreme Gradient Boosting, the Deep Learning Neural Network, and the Random Forest. The performance analysis was performed using the root mean square error, the $R^2$ statistics, the Kling–Gupta efficiency, and the Nash–Sutcliffe Efficiency) statistics and perceptual bias. Three options have been employed to improve the precision of these flow simulation methods, to see the effect of the selected method on the accuracy of the results, the impact of feature engineering on the accuracy and the efficiency of the created models. Moreover, a smart irrigation decision support system (SIDSS) was done to manage irrigation of crops standing on a weekly estimation of water needs using soil measurements and weather parameters collected by divers autonomous nodes disposed of inside the field using ANFIS, and PLSR machine learning techniques referring to [7]. In the same context, a smart system based on open-source technology performs an algorithm based on *K*-means and SVR methods, has been proposed by [8], that provides the irrigation requirements' forecast for the near future using both of sensing of the ground parameter (soil moisture, soil temperature …etc.), and the weather features (humidity, precipitation, UV, and air temperature) predicted for the near future available on the Internet. In this system, the data input is remotely sensed in the cloud using web services, and the acquisition of information insights is ensured in real-time based on sensors network and weather forecast through a decision support system tool and web visualization. Until today, a new decision support system based on models is invented in [9], that relies on wireless sensors network to collect real-time soil and environmental data, neural network algorithm to predict hourly soil moisture content requirements, and soil evapotranspiration benefitting from Blaney–Criddle method and fuzzy logic to monitor and control irrigation efficiency aligned with the weather and to generate and send adequate mobile notifications about irrigation needs into farmer by GSM modem integration. Consequently, water has been saved and yield has been increased appropriately. Recently, there are several models for analytics in machine learning like support vector machines, decision trees, random forests, artificial neural networks, and Bayesian networks used to support farmers in crop cultivation and intelligent farming. Otherwise, an overview about yield prediction based on agrarian factors and weather features compared supervised and unsupervised machine

learning algorithms using various error patterns such as the root mean square error, the relative root mean square error, the mean absolute error, and the $R^2$ determination coefficient as reported in [10].

In an irrigated area in northwestern Bangladesh dependent on groundwater, a study to evaluate the effects of climate change on the cost of irrigation for different RCP situations was directed in [11] applying a general circulation model (GCMs) for projecting the climate, an experiential hydrological pattern based on support vector machines for simulating the state of the groundwater from climatic variables, and a multiple linear regression to estimate the irrigation charge induced through the groundwater levels' fluctuation. The results reveal that the climate changes provoked declination in groundwater level which inflicted the increase of crop production cost less than other costs. To overcome over-or under-irrigation due to spatial changes in deep percolations, rainfall, runoff, irrigation, crop water use, and irrigation depth, and especially, to support decisions on sprinkler irrigation control, a site-specific integrated irrigation controller was invented which allows real-time monitoring of irrigation tasks through Bluetooth communication using an in-field wireless sensor network (WSN) and remote sensing of soil, canopy, air temperature, and soil moisture retrieved from cultivated fields. This system converts an automated irrigation machine from a traditional mechanical and hydraulic system to a controllable electronic system for individual sprinkler control, then, it monitors their geographic locations by a self-positioning system, and it finally makes a decision, when to irrigate and how much water to apply by each sprinkler head in a specific location. The WISC software was tested for in-field wireless sensor-based closed-loop irrigation control during the 2007 growing season under a linear-move irrigation system on a field planted to malting barley in the Eastern Agricultural Research Center of Montana State University in Sidney and it has succeeded to monitor remotely in real-time field conditions and control feedback for site-specific irrigation with a strong correlation of $R^2 = 0.98$ with water captured by catching cans [12].

With the emphasis on the explosion of massive data analysis technologies, it can be noted that there is a multitude of free tools and libraries in python which made available to public access for machine learning, granting an efficient preparation (Numpy and Pandas, etc.) and deep and accurate data analysis and prediction in a reasonable time (PySpark, Keras, Scikit-learn, etc.), and easy results plotting (Matplotlib, Seaborn, etc.). A meaningful study in [13] has compared these libraries to select the better ones for each kind of data preparation, analysis, or prediction. They recommended the usage of Pandas for data preprocessing and manipulation, politely and seaborn and Matplotlib for data customization and visualization while they suggested for the Deep Learning, the usage of PyTorch or Keras for responsive prototyping, and TensorFlow for active customization. Also, they recommended the usage of Hadoop Streaming and PySpark in the field of big data (Table 1).

## Methodology

We performed in our previous experimentations in [4] three machine learning algorithms, in particular, the extreme gradient boosting, the random forest, and the auto-regressive moving average for training the soil data set using various inputs features selected by the resampling method. In this paper, we trained the data set using also the deep ANN with the same resampling inputs. Then, we predicted the soil moisture in several depths for the test period. XGBoost is a novel technique invented by GBMs to boost the accuracy of predictive models benefiting from the predictive power of multiple learners by engaging the gradient boosting trees. In this algorithm, at each iteration, the final predictive model is the aggregated prediction from several weak learners and a new classifier is added to the previous learning models to reduce its errors. XGBoost is implemented in multiple programming languages in parallel with improving parameters as necessary related to [14]. Another model used in this paper is the random forest, that are a combination of multiple tree predictors that provide autonomous predictions using equivalent input data distribution, and at the end of

**Table 1** An overview of predictive models and relevant features by Asmae El Mezouari and Mehdi Najib [4]

| References | Forecasting methods | Parameters |
|---|---|---|
| [5] | Support vector machines, bagging, random forest, and M5P regression trees | Net solar radiation, heat flux, moisture content, wind-speed, mean moisture, mean temperature |
| [6] | Multiple linear regression, random forest, extreme gradient boosting, and deep learning neural network | Meteorological data (precipitation, temperatures and potential evapotranspiration) |
| [7] | Partial least square regression, and adaptive neuro fuzzy inference systems | Soil moisture, soil temperature, rain fall, wind speed, crop evapotranspiration, radiation, dew point |
| [8] | Support vector regression, and SVR + K-means | Moisture, temperature, UV, weather temperature, humidity, and precipitation |

computing, the highly voted predictions are selected as a final output. Random forest empowers an enormous set of weak classifiers to build a robust classifier [15]. ARIMA is a statistic-based model used for prediction based on linear time series analysis upon which the final predicted results are the product of many past examinations and random error [16]. According to [17], the ARIMA overcomes non-stationary time series issues using the differentiation technique of an order ($d$). ARIMA relies on two models, the pure auto-regressive model represented by lag order ($p$) and the moving average model expressed by order of the moving average ($q$). The last predictive model is the deep artificial neural networks that processes the data using multiple layers in the network, in a way similar to the human brain information processing in biology [18]. Deep learning is distinguished from the basic artificial neural networks in the way that the learning nodes are autonomous and can independently train and process the data itself to improve its learning and intelligence.

## Framework Structuring

The proposed framework is invented for soil diagnosis and prediction to boost irrigation scheduling and to ease decision-making in agriculture. Furthermore, it aims not only to allow upload and integrate dataset, to provide predictions but to interpret and export the best results through a web interface as well. The Fig. 1 represents an illustration of the proposed framework composition.

This framework could be described as follow, the first data integration module aims to import the soil and environment data extracted in a MongoDB database, from the environment and soil sensing into the data storage unit of the Hadoop ecosystem (HDFS); while the second data processing module intends to perform parallel predictions through processing different machine learning algorithms, in particular, the ARIMA, XGBoost, random forest, and deep artificial neural networks to select the best predictive model after analyzing their performances, and to export the results through a web interface. The resulting predictions would be used for predicting the soil and environment features that would be employed for water need estimation and irrigation planning. It allows adding other predictive models, features, and performance measures to perform optimum predictions in a fast way.

## Materials and Methods

### Data Processing Tools

Dealing with big data storage and analysis in the agricultural field is a challenging subject, especially when we are talking about aligning with climate changes; we are automatically oriented to lift complications related to retrieving knowledge from climatic and hydrologic historical data. For this reason, we have chosen to integrate data inside the Hadoop using MongoDB Connector, HDFS to read from and write data to the disk, and Apache Spark to speed up machine learning processing via in-memory computation (RAM) as is shown in Fig. 2.

In this paper, we focus on testing different machine learning techniques such as ARIMA, XGBoost, random forest, deep ANN to make predictions in parallel and on a distributed scale and to analyze performance to select and save the best predictive model.

For machine learning processing, spark deployed an open-source and powerful library called MLlib that makes it scalable and wieldy [19]. As with each basic predictive model implementation, and after integrating the data in HDFS using MongoDB Connector; we performed our predictions in Hadoop Spark using python (PySpark) by following the next steps, using diverse libraries (Sparktk, XGBoost4J-Spark, Tree, Keras, …) for each algorithm as needed:

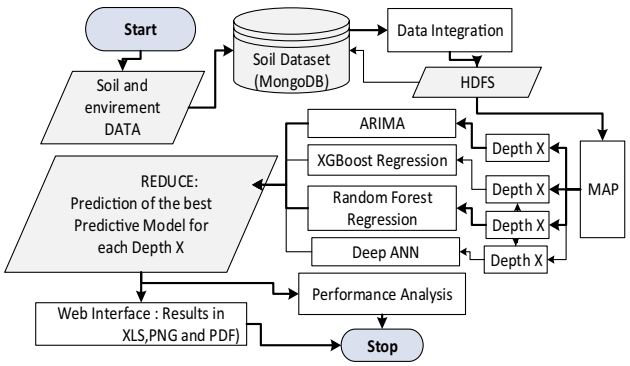


**Fig. 1** An overview of the proposed framework

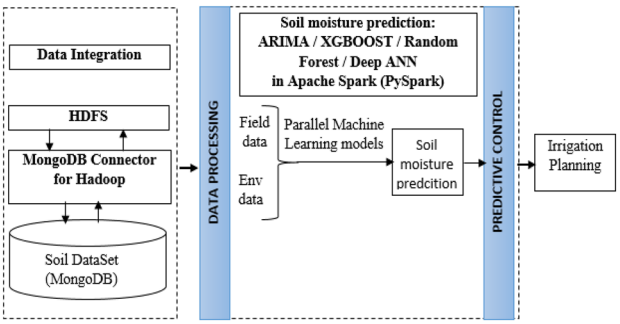


**Fig. 2** An overview of the proposed techniques [4]

- **Step 1**: Reading data file from h in spark, constructing the data frame, and using time method to invoke time.
- **Step 2**: Splitting the dataset into train data and test data.
- **Step 3**: Converting the data into vectors using Vector Assembler.
- **Step 4**: Transforming the vectors into necessary data frames.
- **Step 5**: Building and fitting the model with the training and test data to train it.
- **Step 6**: Making predictions on the test data.
- **Step 7**: Calculating performance measures (MAE, MSE, RMSE, $R^2$-accuracy) to evaluate the accuracy of the model.
- **Step 8**: Exporting and conceiving results.

### Performance Measures

We measured the performance of these predictive models using the mean absolute error, the r-squared accuracy, and the root mean square error.

#### RMSE

The RMSE is the measure of the residuals within divined and perceived values. In general, the decrement of this measure describes that the precision is high. In our case, all predictive models have the same output feature (irrigation amount). Consequently, the root mean square error is sufficient to assess their performance. It is calculated using the formula:

$$\text{RMSE} = \sqrt{\sum_{i=1}^{n} \frac{\left(yi - \overline{y_i}\right)^2}{n}}, \tag{1}$$

where $n$ is the count of the data, $yi$ is the ongoing output of instance $i$, and $\overline{y_i}$ is the corresponding ending estimation.

#### MAE

The mean absolute error measures the absolute deviation between the true and the predicted values. This means that the results that have a negative sign are ignored. MAE is calculated as

$$\text{MAE} = \sum_{i=1}^{n} \frac{|yi - xi|}{n}. \tag{2}$$

**R-Squared Accuracy** R-squared is the fraction by which the variance of the dependent variable is more than the variance of the errors. It describes the square of the correlation between the observed and estimated variables,

$$R^2 \text{ accuracy} = \text{Explained variation/total variation}, \tag{3}$$

The R-squared accuracy is a percentage between 0 and 100:

- 0 tells that the model did not interpret the variability in predicted data around its mean.
- 100 proves that the predictive model reveals completely the variability in the independent variable around its mean.

### Case Study

In the present case study, we tried to test and compare the accuracy of the ARIMA, Random Forest, XGBoost, and Deep ANN methods representing the fundamental shaft of scheduling irrigation. For this purpose, we used a 5 years' real time-series of hourly soil moisture and temperature data, sensed in five depths (5, 20, 35, 50 and 75 cm) from the rain-snow transition zone, the Johnston Draw catchment, Reynolds Creek Experimental Watershed, and Critical Zone Observatory, USA [20]. This Dataset contains over 35,064 records from 10/1/2010 to 09/30/2014. We have tested these methods in the data of the 5 cm depth. Figure 5 illustrates the trends of the studied time-series (Fig. 3).

## Results and Discussion

### ARIMA Model Selection

ARIMA model selection consists of specifying the three parameters $p$, $d$, and $q$. In the first step, we started by the determination of the parameter "$d$" that represents the differencing order needed to make the time series stationary. This parameter is determined by the autocorrelation diagram analysis of the time series. Figure 4 shows a regular decrease in the autocorrelation values which indicates that the initial time-series is not stationary. In our case, the first differencing order was adequate to make it stationary, and the null autocorrelation value in Fig. 5 proves this hypothesis.

The auto-correlation (AC) and the partial-autocorrelation (PAC) diagrams are used to determine the "$q$" parameters of the MA model, and the "$p$" parameter of the AR model. Referring to Fig. 5, any order of the AC diagram exceeds the confidence level. Thus, parameter $q$ is equal to 0. Based on the analysis of the PAC in Fig. 4, we can observe that the first two autocorrelation values exceed the confidence level. As a result, the "$p$" parameter can be equal to 0, 1 or 2. To select the best configuration, we tested the accuracy of all combinations of the three parameters like the following configurations ARIMA (1, 0, 0), ARIMA (1, 1, 1), and ARIMA (2, 0, 2) using out-of-time cross-validation.

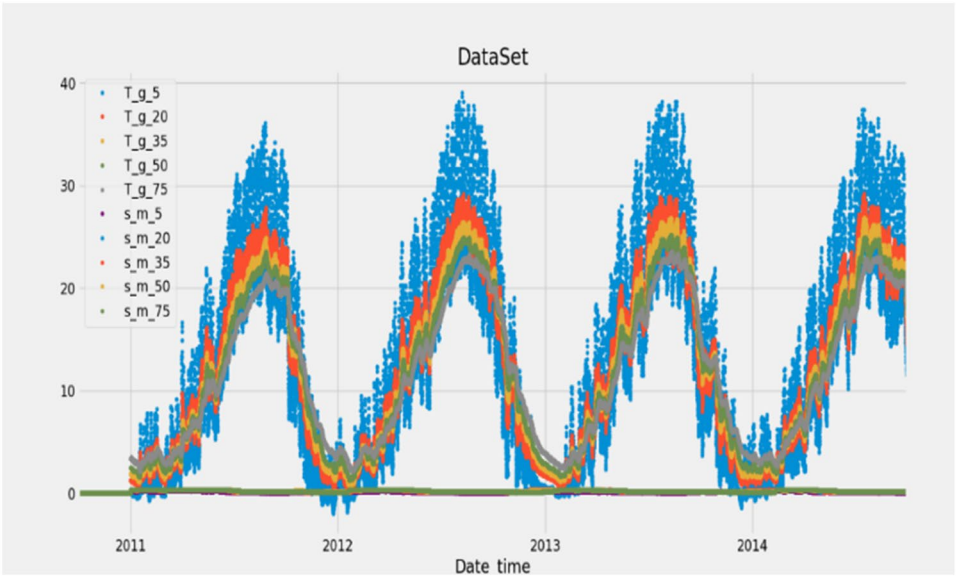


**Fig. 3** Times-series of hourly soil moisture and temperature in 5 depths (USA) [4]

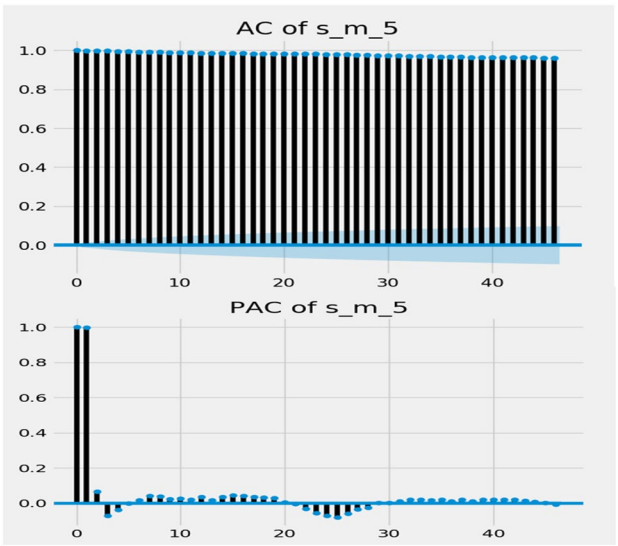


**Fig. 4** AC and PAC of non-stationary soil moisture in 5 cm depth [4]

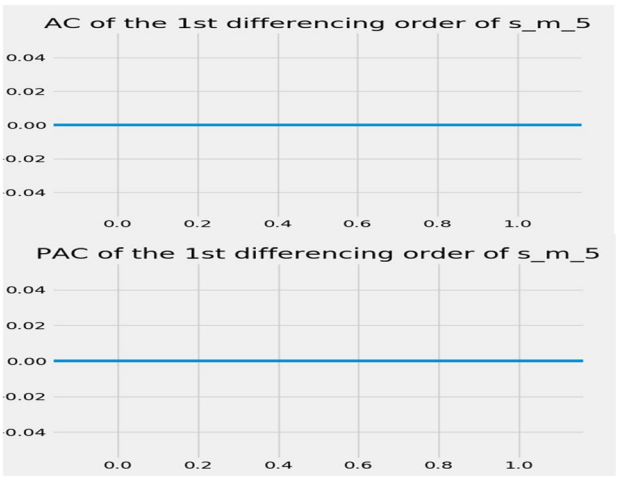


**Fig. 5** AC and PAC—first-order differencing soil moisture in 5 cm depth [4]

In out-of-time cross-validation, we can go back a few steps in time and predict the future for as many steps as we have taken. Then we perform the comparison between the forecast and the perceived data. To do out-of-time cross-validation, you need to build a training and test dataset by dividing the time series into two adjoining parts near the 75:25 ratio or a reasonable distribution based on the time frequency of the series. The obtained results show that the last model outperforms the other models in terms of accuracy. Consequently, ARIMA (2, 0, 2) is chosen for soil moisture prediction. Figure 6 portrays the results of the soil moisture prediction for the tested last year using this model.

## XGBoost Model Selection

In this step, we trained XGBoost Regression Model using soil moisture and temperature of four years in the 5 cm depth and then we used only the soil moisture parameter in the same depth. After that, we performed predictions for the last year. The trend in Fig. 7 illustrates the results of prediction using soil moisture and temperature parameters and the trend in Fig. 8 shows the result of the prediction using only soil moisture parameter.

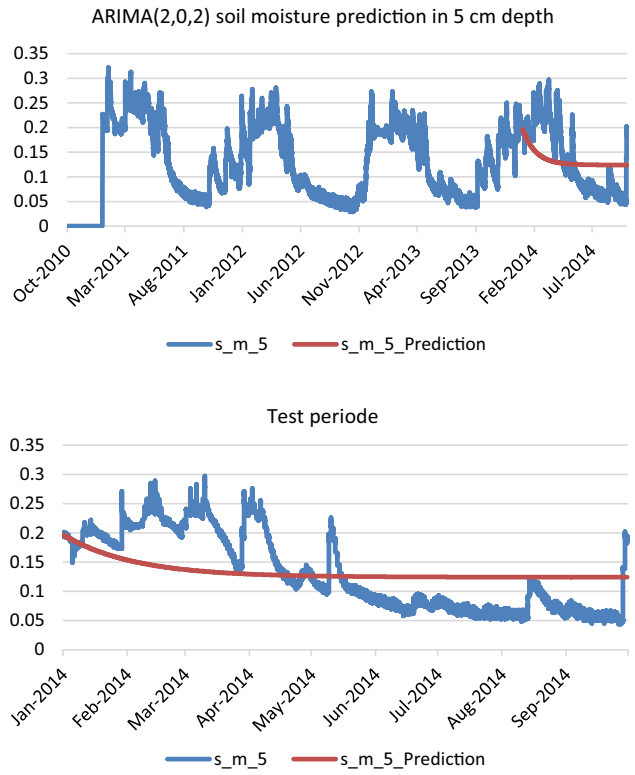


**Fig. 6** The 5cm depth' soil moisture prediction using exclusively the soil moisture parameter [4]

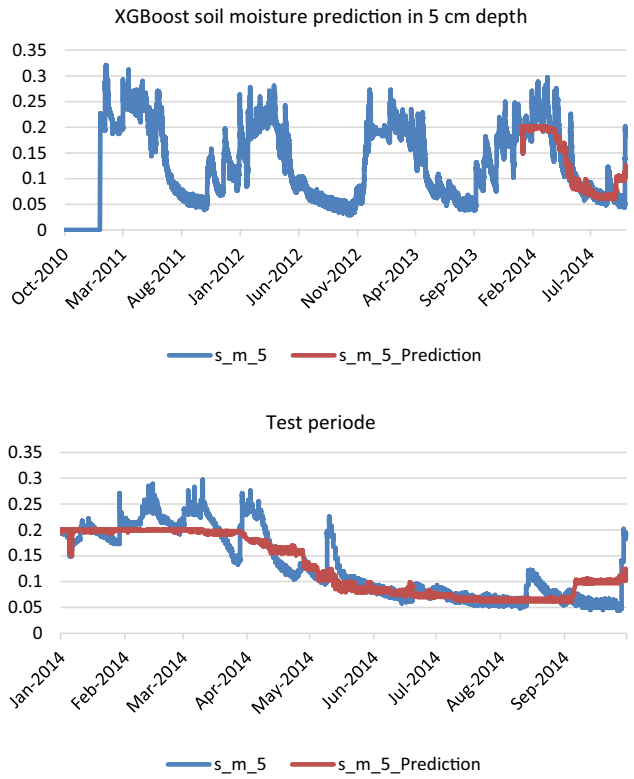


**Fig. 7** XGBoost soil moisture prediction in 5 cm depth using soil moisture and temperature parameters [4]

## Random Forest Model Selection

In this step, we trained the random forest regression model using soil moisture and temperature of 4 years in the 5 cm depth, and then we used only the soil moisture parameter in the same depth. After that, we performed predictions for the last year. The trend below in Fig. 9 shows the result of prediction using soil moisture and temperature parameters and the trend in Fig. 10 shows the result of prediction using only soil moisture parameter.

## Deep Artificial Neural Network Model Selection

In this step, we trained deep artificial neural network model using the first four years' soil moisture and temperature in the 5 cm depth, and then we used only the soil moisture parameter in the same depth. After that, we made predictions for the last year.

We created our Deep ANN (multi-layer perceptron) using the Keras sequential model combined with the rmsprop optimizer, which is a very popular optimization algorithm. We also employed an input layer of the ten relevant features for the first sampling input (soil moisture, soil temperature, hour, day of the week, quarter, month, year, day of the year, day of the month, and the week of the year) and an input

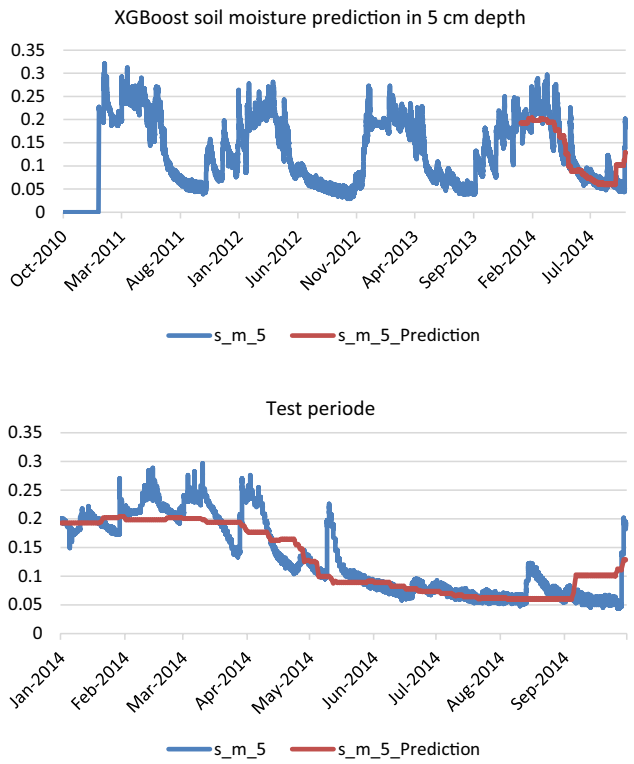


**Fig. 8** XGBoost soil moisture prediction in 5 cm depth using only soil moisture parameter [4]

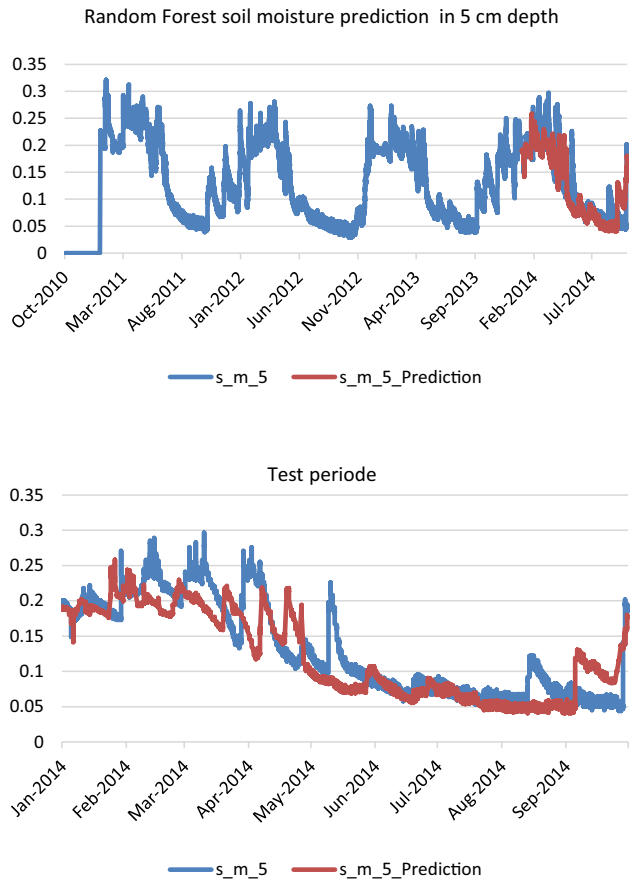


**Fig. 9** Random forests soil moisture prediction in 5 cm depth using soil moisture and temperature parameters [4]

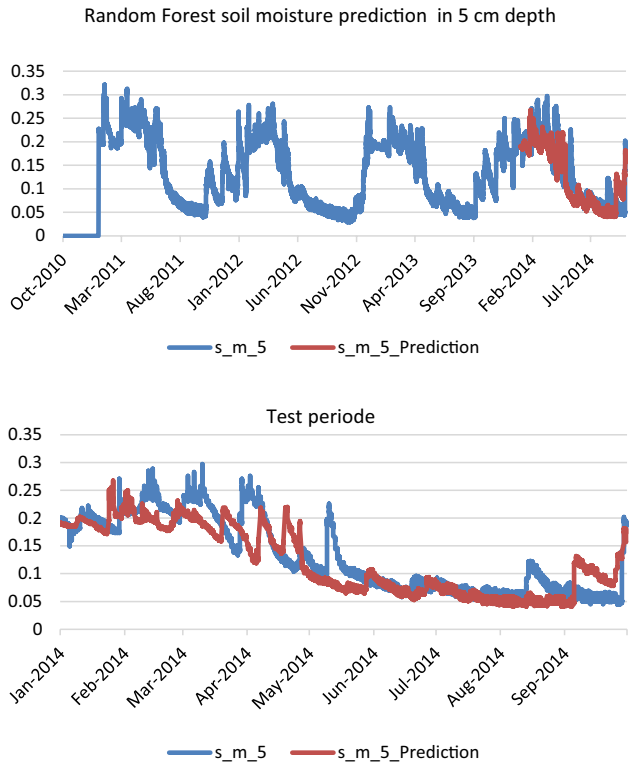


**Fig. 10** Random forest soil moisture prediction in 5 cm depth using only soil moisture parameter [4]

layer of the 9 relevant features for the second sampling input (soil moisture, hour, day of the week, quarter, month, year, day of the year, day of the month, and the week of the year), with the activation function relu and using 100 hidden units for all experiments. Moreover, we added a hidden layer with 60 hidden units, with the activation function relu and an output layer for predicting the target feature (soil moisture). Likewise, we used the MSE as a loss function, the MAE, and the accuracy as evaluation metrics. Figure 11 shows the ANN model used to perform prediction with the described typical configuration for linear regression. After trying many configurations, like different random hidden units, and various activation functions, we found that this model is the appropriate option in terms of accuracy.

The trend below in Fig. 12 shows the result of prediction using soil moisture and temperature parameters and the trend in Fig. 13 shows the result of prediction using only soil moisture parameter. It seems that the curve representing the forecast of soil moisture based solely on soil moisture follows a quite precise trend close to the real values.

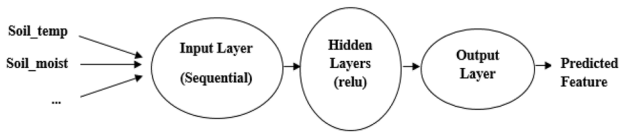


**Fig. 11** The ANN model's configuration

## Synthesis

The tests carried out by the ARIMA (2, 0, 2), the XGBoost, the random forests, and the deep ANN algorithms show that both methods are accurate for forecasting soil moisture. The calculation of the root mean square error, the mean absolute error, and the R-squared accuracy prove the efficiency of these models, Table 2 shows the evaluation matrix.

Finally, we find that the deep ANN outperforms both models in terms of accuracy in predicting using only the soil moisture parameter and this is maybe related to the homogeneity in the data that reinforces the learning of the deep ANN model.

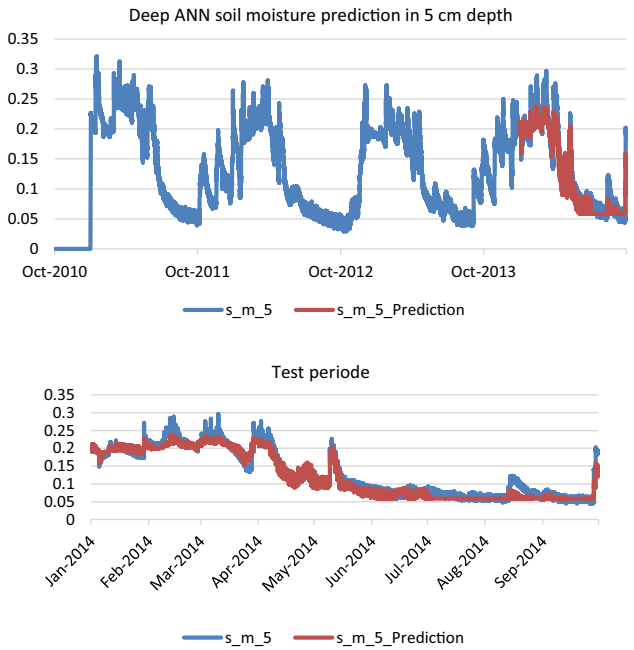


**Fig. 12** Deep ANN soil moisture prediction in 5 cm depth using soil moisture and temperature parameters

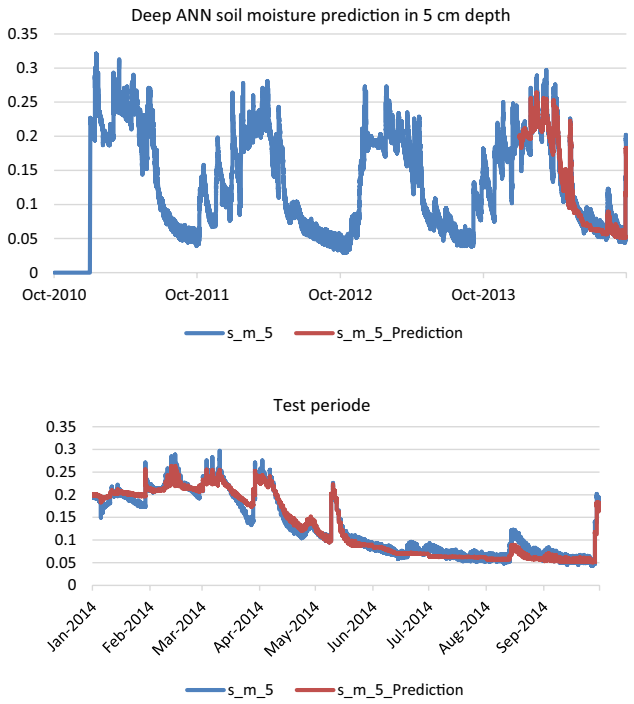


**Fig. 13** Deep ANN soil moisture prediction in 5 cm depth using only soil moisture parameter

In addition, the appropriate configuration of the model like increasing the number of epochs minimizes the errors. Likewise, the activation function injects nonlinear properties into the network to learn any complex relationship between input and output; and that represents one of the principal interesting concepts in universal approximation implemented in the ANN model which improves the efficiency of the model. Besides, we observed clearly that XGBoost and Random Forests give the same accuracy whether if we used the soil temperature parameter or not.

## Conclusion and Perspectives

In the present paper, we engaged in improving our proposed framework that allows us to compute various predictive algorithms over the soil variables in different depths, by adding the Artificial Neural Network model in the processing phase. The main goal of this experimentation is to select the most accurate predictive model that will anticipate the soil state changes and help farmers in aligning irrigation scheduling with climatic changes; and basically, in improving the yield in agriculture, benefitting from various supervised learning machines by comparing their efficiencie. Hence, to select the best predictive model that would improve the irrigation planning, we have compared four forecasting models especially ARIMA, XGBoost, Random Forests, and Deep ANN in terms of several precision measures such as the MAE, the RMSE, and the R-squared accuracy..

**Table 2** Confusion matrix of the evaluated models

| Models/parameters | Evaluation | Moisture and temperature | Only soil moisture |
|---|---|---|---|
| ARIMA | MAE | – | 0.052 |
| | RMSE | – | 0.060 |
| | R-squared accuracy | – | 0.211 |
| XGBoost | MAE | 0.021 | 0.022 |
| | RMSE | 0.148 | 0.148 |
| | R-squared accuracy | −0.386 | −0.386 |
| Random forests | MAE | 0.027 | 0.027 |
| | RMSE | 0.148 | 0.148 |
| | R-squared accuracy | −0.386 | −0.386 |
| ANN | MAE | 0.013 | **0.008** |
| | RMSE | 0.017 | **0.011** |
| | R-squared accuracy | 0.92 | **0.97** |

Moreover, to approve this solution, we tested these forecasting methods upon a real-time series of soil moisture and temperature in the USA and we have examined their efficiency using different performance measures. Based on the results, we found that both ARIMA, XGBoost, Random

Forests, and Deep ANN models provided accurate predictions. However, the Deep ANN outperforms all models in terms of precision in all cases.

As a perspective, we suppose that employing such efficient and powerful processing and predictive tools in forecasting the soil state could support irrigation planning in the short and the long terms. Also, testing other predictive models in future work could improve the results, particularly while integrating different parameters and validation processes.

**Author contributions** Not applicable.

**Funding** Not applicable.

## Declarations

**Conflict of interest** On behalf of all authors, the corresponding author states that there is no conflict of interest.

**Availability of data and material** The data is open-source.

**Code availability** Not applicable.

**Ethics approval** Not applicable.

**Consent to participate** Not applicable.

**Consent for publication** Not applicable.

## References

1. Jury WA, Vaux HJ. The emerging global water crisis: managing scarcity and conflict between water users. Adv Agron. 2007;95:1–76.
2. Barkunan SR, Bhanumathi V, Sethuram J. Smart sensor for automatic drip irrigation system for paddy cultivation. Comput Electr Eng. 2019;73:180–93. https://doi.org/10.1016/j.compeleceng.2018.11.013.
3. Gutierrez J, Villa-Medina JF, Nieto-Garibay A, Porta-Gandara MA. Automated irrigation system using a wireless sensor network and GPRS module. IEEE Trans Instrum Meas. 2014. https://doi.org/10.1109/TIM.2013.2276487.
4. Mezouari AEL, Najib M (2019) A Hadoop based framework for soil parameters prediction. IEEE, pp 681–687
5. Granata F. Evapotranspiration evaluation models based on machine learning algorithms—a comparative study. Agric Water Manag. 2019;217:303–15.
6. Cisty M., Soldanova V. (2018) Flow Prediction Versus Flow Simulation Using Machine Learning Algorithms. In: Perner P. (eds) Machine Learning and Data Mining in Pattern Recognition. MLDM 2018. Lecture Notes in Computer Science, vol 10935. Springer, Cham. https://doi.org/10.1007/978-3-319-96133-0_28.
7. Navarro-Hellín H, Martínez-del-Rincon J, Domingo-Miguel R, et al. A decision support system for managing irrigation in agriculture. Comput Electron Agric. 2016;124:121–31.
8. Goap A, Sharma D, Shukla AK, Krishna CR. An IoT based smart irrigation management system using machine learning and open source technologies. Comput Electron Agric. 2018;155:41–9.
9. Mohapatra AG, Lenka SK, Keswani B. Neural network and fuzzy logic based smart DSS model for irrigation notification and control in precision agriculture. Proc Natl Acad Sci India Sect A Phys Sci. 2019;89:67–76. https://doi.org/10.1007/s40010-017-0401-6.
10. Elavarasan D, Vincent DR, Sharma V, et al. Forecasting yield by integrating agrarian factors and machine learning models: a survey. Comput Electron Agric. 2018;155:257–82. https://doi.org/10.1016/j.compag.2018.10.024.
11. Salem GSA, Kazama S, Shahid S, Dey NC. Impacts of climate change on groundwater level and irrigation cost in a groundwater dependent irrigated region. Agric Water Manag. 2018;208:33–42. https://doi.org/10.1016/j.agwat.2018.06.011.
12. Kim Y, Evans RG. Software design for wireless sensor-based site-specific irrigation. Comput Electron Agric. 2009;66:159–65.
13. Stancin I, Jovic A (2019) An overview and comparison of free Python libraries for data mining and big data analysis. In: 42th International conference on Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2019, pp. 977–982. https://doi.org/10.23919/MIPRO.2019.8757088.
14. Gupta A, Gusain K, Popli B (2016) Verifying the value and veracity of extreme gradient boosted decision trees on a variety of datasets. In: 11th International Conference on Industrial and Information Systems (ICIIS), 2016, pp. 457–462. https://doi.org/10.1109/ICIINFS.2016.8262984.
15. Breiman L. Random forests. Mach Learn. 2001. https://doi.org/10.1023/A:1010933404324.
16. Narayanan P, Basistha A, Sarkar S, Kamna S. Trend analysis and ARIMA modelling of pre-monsoon rainfall data for western India. Comptes Rendus Geosci. 2013;345:22–7.
17. Ramos P, Santos N, Rebelo R. Performance of state space and ARIMA models for consumer retail sales forecasting. Robot Comput Integr Manuf. 2015;34:151–63.
18. Schmidhuber J. Deep learning in neural networks: an overview. Neural Netw. 2015;61:85–117.
19. Meng X, Bradley J, Yavuz B, et al. MLlib: machine learning in Apache Spark. J Mach Learn Res. 2016;17:1235–41.
20. Godsey SE, Marks D, Kormos PR, et al. Eleven years of mountain weather, snow, soil moisture and streamflow data from the rain–snow transition zone–the Johnston Draw catchment, Reynolds Creek Experimental Watershed and Critical Zone Observatory, USA. Earth Syst Sci Data. 2018;10:1207–16.