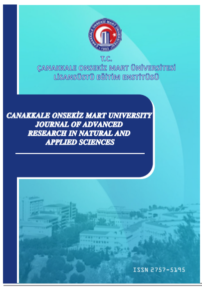
e-ISSN 2757-5195



Çanakkale Onsekiz Mart University
Journal of Advanced Research in Natural and Applied Sciences
Open Access

doi.org/10.28979/jarnas.984312 2022, Vol 8, Issue 2, Pages: 309-321 dergipark.org.tr/tr/pub/jarnas

Sensory Precipitation Forecast Using Artificial Neural Networks and Decision Trees

Hakkı Fırat Altınbilek[1], Hakan Nar[2], Sefa Aksu[2], Ünal Kızıl[2*]

[1]Dept. of Agricultural Structures and Irrigation, School of Graduate Studies, Çanakkale Onsekiz Mart University, Çanakkale, Türkiye
[2]Dept. of Agricultural Structures and Irrigation, Faculty of Agriculture, Çanakkale Onsekiz Mart University, Çanakkale, Türkiye

Article History
Received: 19.08.2021
Accepted: 11.01.2022
Published: 10.06.2022
Research Article

Abstract − Meteorology stations sold in the market have various difficulties in terms of their use, also these systems are costly to obtain. With state of the art sensor technologies, the development of mini weather stations has become easier. In this study it was aimed to develop a prototype low-cost weather station using temperature, relative humidity, ultraviolet (UV), light dependent resistor (LDR), rain and soil moisture sensors to collect major meteorological data. The collected data transmitted to the remote station for logging via a GSM module and the information was sent to the database in the internet environment. In addition, the data from the sensors are organized by correlation. The classification was made according to the data obtained from the rain sensor and the relationship between the other 5 sensors used in the device to the rain classification was examined. Sensor data were scaled between 0-1 with min-max normalization before being subjected to deep learning and machine learning training. In the Decision Tree (DT) a model score of 0.96 was obtained by choosing the maximum depth of 2. The artificial neural network (ANN) deep learning yielded a classification score of 0.92 using 4 hidden layers and 100 epochs in the artificial neural network model.

Keywords − *Agricultural sensors, environmental control, machine learning, precipitation forecast, weather station*

1. Introduction

It is of great importance that environmental data such as relative humidity, precipitation, temperature, UV can be collected and recorded over a time period for agricultural purposes. The sensor networks make it easy and possible to store, evaluate and automate environmental data collection systems (Valada, Kohanbash & Kantor, 2010). With the improvements in technology the sensory systems have become popular because of their advantages such as smaller size, low cost and accessibility.

Using such networks meteorological data can be used to control the production process by instant and simultaneous monitoring. The data transmitted by the smart sensors during a predetermined time interval is collected by a main station located in a server. Then, the decision making systems can interpret the data for agricultural warning, drought, yield, disease, activity management, crop status information, transition to the phenological stage etc. (Zhang, Wang & Wang, 2002).

Sirohi, Tanwar, Himanshu & Jindal (2016) used wireless sensor network to produce weather-related conclusions after processing the information collected from temperature, humidity, and smoke sensors. They reported that it can be promising system in operating the automatic irrigation systems. Ucgun & Kaplan (2017) developed a mini weather station system with an Arduino-based processor that allows predicting data from temperature, humidity, pressure and rain sensors. Kumari, Kasliwal & Valakunde (2018) pre-

[1] hfirataltinbilek@gmail.com
[2] hakannar22@gmail.com
[3] aksusefa@comu.edu.tr
[4] unal@comu.edu.tr

*Corresponding Author

sented an Android and internet-of-things-based (IoT) intelligent environment monitoring system. The developed system is capable of measuring some environmental characteristics such as air, water, and soil. As a result, the system includes a variety of sensors employed with Raspberry Pi card. Durrani, Khurram & Khan (2019) developed a smart weather station. Their design is outfitted with a number of environmental sensors that communicates with the cloud. Furthermore, using machine learning algorithms, they can forecast future weather data. Joshi, Mistry, Khan, Motekar & Chaugule (2021) developed a mini weather station using DHT11 Temperature and Humidity sensor, MQ7 Gas Sensor and MPX10DP Pressure sensor. They reported that all these sensor data can be used in electronic modelling in the cloud environment.

Machine learning (ML) has been becoming a very useful tool in engineering since it significantly reduces the computational time (Bhagat, Tung & Yaseen, 2020). With the recent advancements in ML techniques, it started to be used as a statistical prediction and classification method in meteorology. Thus, a very convenient method has emerged for estimating data at different scales (Bochtis, Liakos, Busato, Moshou & Pearson, 2018). Supervised machine learning ensures the predictability of the output based on the classes created. This depends on accuracy of training and testing of the model for better classification or predictions. Abyaneh, Varkeshi, Golmohammadi & Mohammadi (2016) has reported the performance of artificial neural networks (ANNs) in soil temperature predictions. Gagne, McGovern & Xue (2014) investigated how several machine learning algorithms performed in creating probabilistic, deterministic, and quantile precipitation forecasts for individual grid locations over the central United States. They reported by constantly leveraging the potential of the available information, machine learning techniques can improve precipitation forecasts. Huang, Lin, Huang & Xing (2017) used the K-nearest neighbour (KNN) method to execute rainfall forecasting in Beijing, and the empirical findings showed that this method produces good predicting results. Young & Liu (2015) suggested a physically based and artificial neural network hybrid model to improve rainfall-runoff modelling. The two components of the hybrid model based on separate philosophies, in particular, complement each other in terms of intrinsic strengths and limits. Manandhar, Dev, Lee, Meng & Winkler (2019) proposed to use the 4-year (2012–2015) database weather features for rainfall prediction by using machine-learning algorithm. In comparison to previous research, this strategy considerably lowers the rate of false alarms Ortiz-Garcia, Salcedo-Sanz & Casanova-Mateo (2014) presented a study on the performance of the Support Vector Machine (SVM) in a problem of daily precipitation prediction. They showed that comparing alternative approach to predict precipitation show that the SVM is the best of the classification models considered. Mehr (2021) predicted the standardized precipitation index and standardized precipitation evaporation index in cities of central Ankara and Antalya, respectively. He reported that gradient boosting decision tree (GBT) is a promising technique in drought classification. Mohapatra, Rakesh, Purwar & Dimri (2021) classified 117 years of precipitation data from 36 meteorological stations (1901-2017) in India as dry, wet and transitional periods using cluster analysis approach. Hwang, Orenstein, Cohen, Pfeiffer & Mackey (2019) developed a database that can predict seasonal temperature and precipitation data using local linear regression and k-nearest neighbour models. They reported that both models give better temperature and precipitation forecasts than the US Climate Forecasting System. Yeditha, Kasi, Rathinasamy & Agarwal (2020) developed an artificial neural network model to compare precipitation data obtained from 2 different satellite drivers with classical weather station data. They reported that the model is more promising than other benchmark models and has the potential to foresee extraordinary surge occasions.

In this study, a low-cost weather station is developed employing cloud-based database management and machine learning techniques to predict precipitation. In the cloud-based web service, readings of 5 different environmental sensors values incorporated into the Arduino microprocessor. The decision trees (DT) and artificial neural networks (ANN) deep learning model are used to develop models that can predict precipitation using sensory data.

2. Materials and Methods

2.1 Database

Daily weather data is collected with the prototype weather station placed nearby a research greenhouse located on the Çanakkale Onsekiz Mart University, Faculty of Agriculture campus between 14.01.2021 and 23.02.2021. The data coming from the sensors were recorded in the MySQL web-based database with 1-minute intervals via the GSM module. Due to its web-based nature, the outputs of the sensor values can be monitored without the need for installation on any device with internet access. The flow chart of the system is shown in Figure 1.

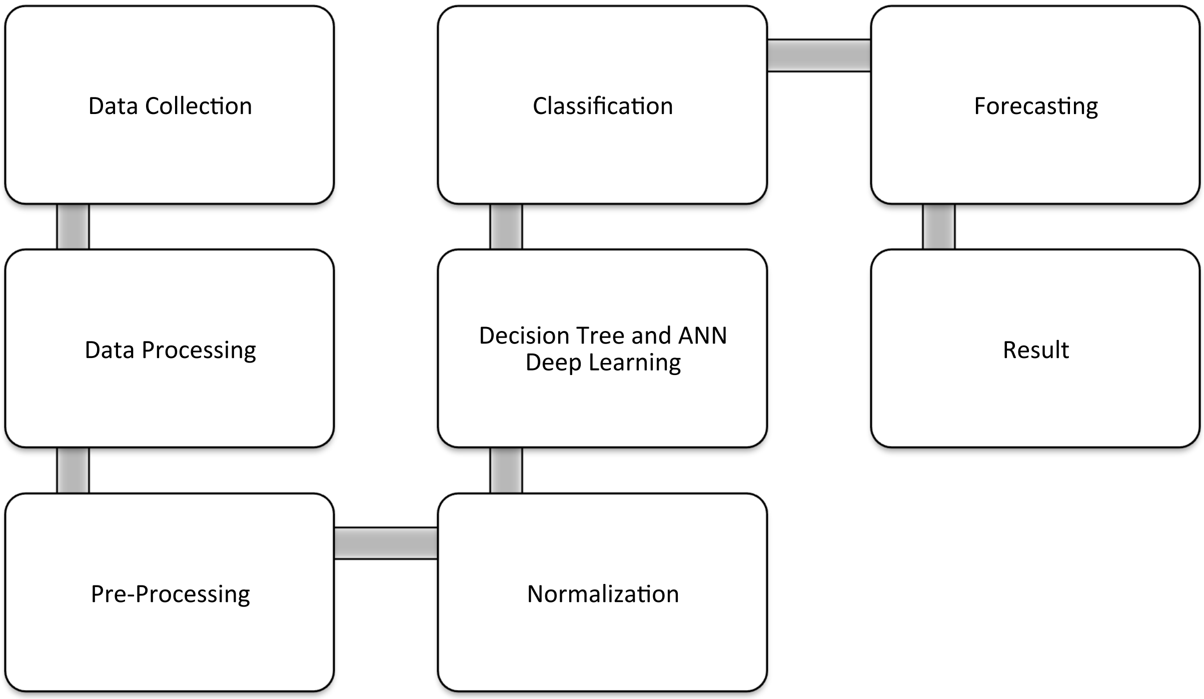


Figure 1. The flowchart of the system

Data from the Arduino Rain sensor is classified as dry or wet conditions. Depending on the sensor response, the differences in precipitation values can be graded and give an idea about the intensity of the rain. Threshold values of rain sensor data are given in Table 1. All sensor data were taken at 1-minute intervals and stored in an internet-based database. Binning and min-max scaler methods were applied on the dataset and noisy data were eliminated.

Table 1

Rain sensor thresholds

Sensitive Area		**Capacitance**	**Ratio Capacitance (%)**	**Thresholds**	**Precipitation Class**
Range-Dry (%)	**Range-Water (%)**	**(pF)**			
81-100	0-20	100	0	Between 0 and 76	0
61-80	21-40	176	76	Between 77 and 232	1
41-60	41-60	232	232	Between 233 and 359	2
21-40	61-80	300	300	Between 233 and 300	3
0-20	81-100	359	359	>= 359	4

Due to the high stability of the support and the inalterability of the "sensitive" surface, the Arduino rain sensor ensures exceptional reliability, even after cleaning with solvents and/or working hard situations. Furthermore, in the presence of water, the capacitance rises to high levels compared to dry conditions, with a 359% change in the ratio (Ardiansyah et al., 2018). When the system detects rain, it classifies precipitation according to the threshold values given in Table 1. The rain sensor's output value has a sensitivity range of 0 to 100 %. The resulting response is in the sensitivity of dry environments if the value is near to 100 %. The ensuing response will be in the damp and humid environments if the value is near to the minimal value.

2.1.1 Binning Method

Data binning, also known as bucketing, is a pre-processing technique for reducing the impact of modest observation errors. The original data values are separated into small intervals known as bins, and then a general value produced for that bin is used to replace them. This smooths the input data and, in the case of limited datasets, may lower the likelihood of overfitting (Teanby, 2006).

The continuous data received from the Arduino Rain sensor should be spread over the range. For this, the binning method operation to clean out the outliers by finding the Min and Max values were used in this study. The normalization step of the Binning limits in the Dataset and the corresponding precipitation classes are shown in Table 2, and then the Min Max scaler method is used.

Table 2

Few precipitation forecast records after binning normalization

Binning Limits	Arduino Rain Sensor Output	Precipitation Forecast
0	55	0
77	139	1
233	185	1
300	255	2
359	304	3
973	968	4

As a result of normalization of rain sensor data with Binning Limits, each 10712 records were grouped. The mean value of each group was calculated and thus the min – max scaler method was used based on the group averages.

2.1.2 Min Max Scaler

MinMaxScaler subtracts the feature's minimum value from each value in the feature, then divides by the range. The range is the difference between the maximum and least values at the start. The shape of the original distribution is preserved by Min- Max Scaler. It has no effect on the information included in the original data. The default range for the feature returned by MinMaxScaler is 0 to 1 (Tian & Chen, 2021).

All sensor values were inserted into the MinMaxScaler method and noisy data were eliminated based on the means of features. After the elimination of 569 outlier data in Jupyter Notebook, remainder 53.563 readings were used in modelling purposes. MinMaxScaler equivalent of some records is given Table 3.

Table 3

Min-max scaler equivalent of some records

Id	Rain Ratio	TNS	NTC	LDR	UV	Temp	RH	Precipitation Class
1	0.041169	0.0	0.609342	0.990536	0.092227	0.605263	0.152778	0
640	0.050465	0.247688	0.609342	0.0	0.0	0.236842	0.833333	1
1010	0.034529	0.273381	0.607219	0.066246	0.0	0.157895	0.944444	2
16850	0.841965	0.346351	0.602972	0.366982	0.544137	0.394737	0.777778	3
47550	0.837981	0.417266	0.0	0.0	0.0	0.289474	0.833333	4
53563	0.830013	0.191161	0.755839	0.107256	0.050066	0.394737	0.263889	1

Normalization calculation was made with the min and max values found. From the dataset, 0.1% of data was removed that contradicted the means value of each sensor. Then, the models were developed.

2.2 Electronics Components Used In Device

Arduino microprocessor has been used primarily as a control card in order to enable the sensors, circuits and other auxiliary materials. Arduino is an open source physical programming platform that includes an Atmel AVR microcontroller and additional peripherals. It can perform basic input and output applications using the Processing/wiring programming language, stands out with its features such as being open source, easy to use libraries, and easy availability. In the study, Arduino Uno R3 board based on Atmega328 microprocessor was used.

The device has been developed by integrating Grove - UV ultraviolet sensor (SeeedStudio, Shenzhen, China), (Figure 2.a), DHT11 temperature and humidity sensor (Aosong Electronic Co., Guangzhou, China), (Figure 2.b), LDR light detection sensor (Robotistan, Başakşehir, İstanbul), (Figure 2.c), 100K NTC thermistor (Shiheng Electronics, Nanjing City, China), Arduino rain sensor (Robotistan, Başakşehir, İstanbul), (Figure 2.d), capacitive soil moisture sensor (DFRobot, Shangai, China), (Figure 2.e) and SIM900 GSM/GPRS module (Elecrow, Shenzhen, China) to Arduino Uno R3 microcontroller (Arduino LCC, Boston, Massachusetts, USA). The picture of the developed system in operation is shown in Figure 3.

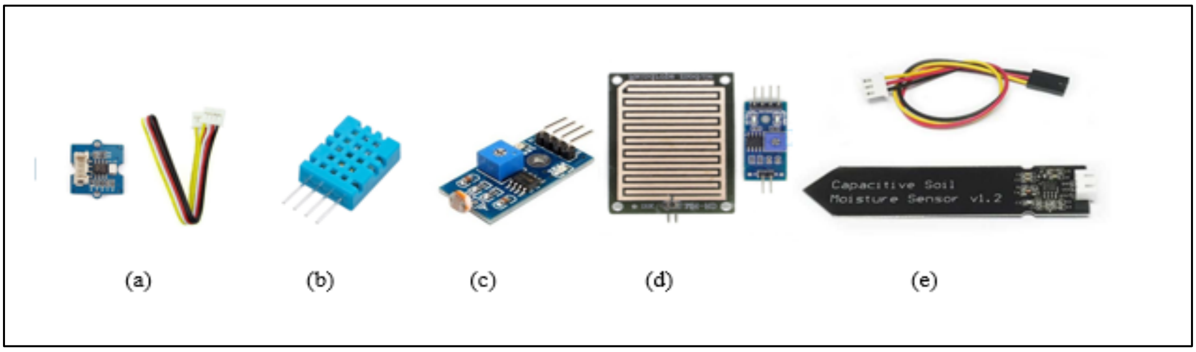


Figure 2. Grove-uv sensor (a), temperature and humidity sensor (b), ldr sensor (c), rain sensor (d), soil moisture sensor (e)

Figure 3. The developed device

A schematic representation of the data transfer and database management steps is given in Figure 4.

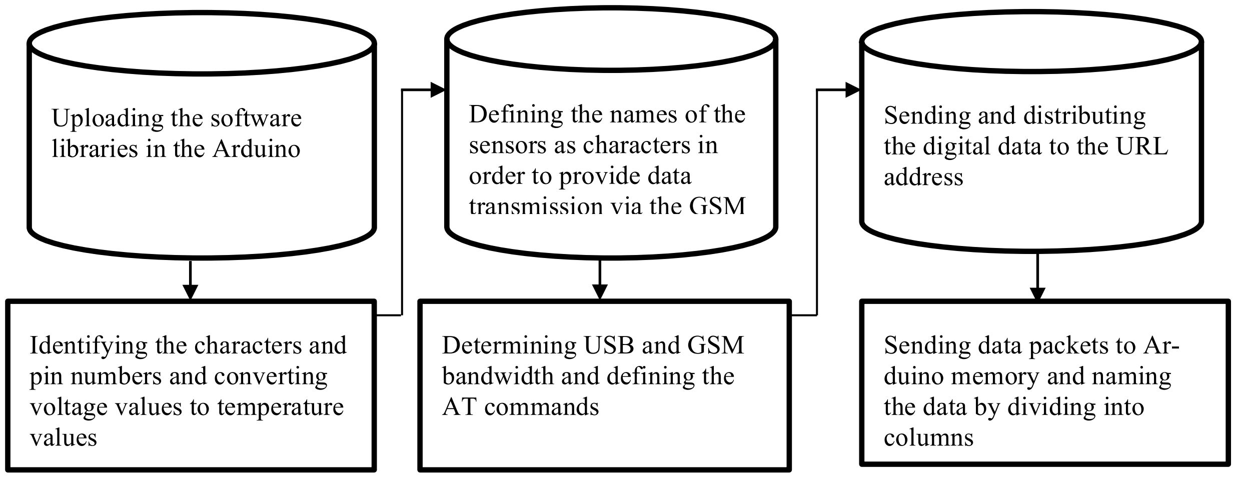


Figure 4. Data transfer and database development steps

2.3 Decision Tree

This method is a tool that approximates the goal functions and shows the learning function. A DT is a descriptive and predictive model in tree view. This model helps the decision maker to consider which factors to consider when making a decision, and to determine how each factor relates to the different outcomes of the decision in the past. The DTs are widely used among data mining classification models since they are cheap to set up in data mining, easy to interpret, easy to integrate with database systems, and satisfactory reliability (Pal and Mather, 2003).

In order to create a classification tree, there is a feature that best determines the examples in the learning set. With this feature, a branch and leaves of the tree are decomposed and a new sample set is created. From the examples on this decomposed branch, a new determinant feature is found and new branches are created (Covert and Thomas, 1991).

Entropy measurement is used to find the gain of information. Entropy is randomness, uncertainty, and the probability of an unexpected occurrence. To calculate the information gain of a system, let's say if S contains records from Ci class i = (1, … ,n), the information required to classify a record is calculated by equation 2.1.

$$Information\ (s_1, s_2, \ldots, s_n) \ = \sum_{i=1}^{n} \left(\frac{s_i}{s}\right) \left(log_2 \left(\frac{s_i}{s}\right)\right) \tag{2.1}$$

Equation 2.2 is used to find the entropy of a variable A with the values (a1, a2, …, av) it can take as many as (j = 1,2, ..., v).

$$\boldsymbol{Entropy\ (A)\ = \sum_{j=1}^{v} \left(\frac{s_{1j} + \cdots + s_{nj}}{s}\right) \times Information\ (s_1, s_2, \ldots, s_n)} \tag{2.2}$$

The information gain obtained by branching the tree using variable A is found by equation 2.3.

$$\boldsymbol{Gain\ (A)\ =\ Information\ (s_1, s_2, \ldots, s_n)\ -\ \mathbf{Entropy(A)}} \tag{2.3}$$

The Gini index assumes that all variables are continuous. The division is made over the variable with the discrimination that gives the lowest gini index value that disrupts this continuity. If a data set T contains

N records in n different classes, pi determines the relative frequency of class j in T and the Gini index is calculated by equation 2.4.

$$Gini\,(T) = 1 - \sum_{j=1}^{n} p_{j^2} \tag{2.4}$$

If the data set T1 and T2 are divided into two parts of size N1 and N2, respectively, the gini index is chosen from the variable with the lowest separation according to equation 2.5.

$$Gini_{separation} = \frac{N_1}{N} \times Gini\,(T_1) + \frac{N_2}{N} \times Gini\,(T_2) \tag{2.5}$$

Within the scope of the study, the flow chart of the decision tree model is given in Figure 5.

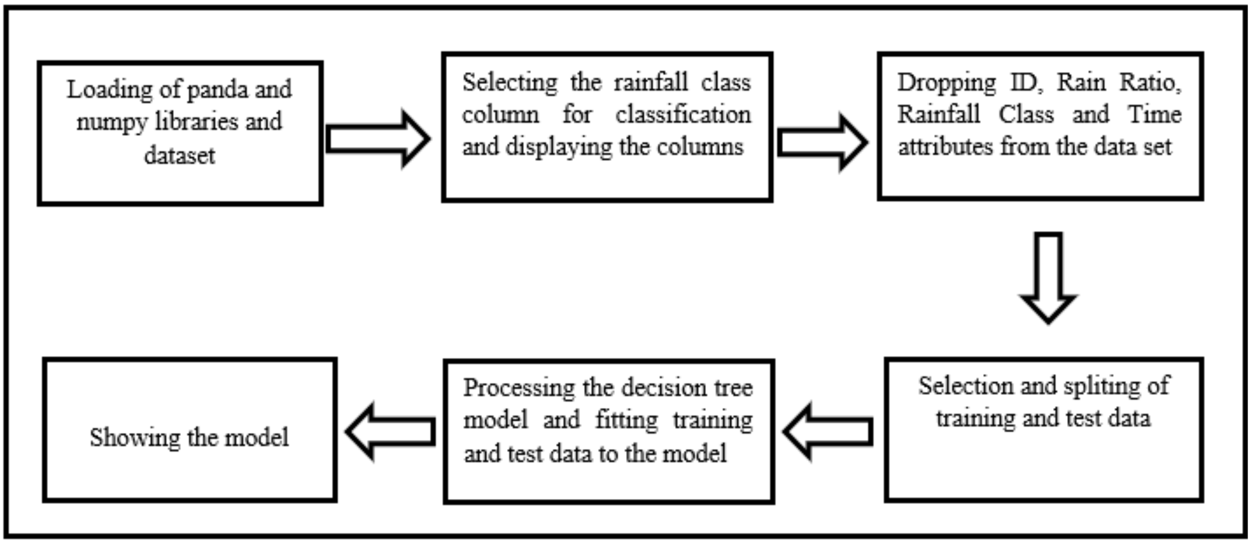


Figure 5. The flowchart of the decision tree

In the study, using the sklearn.model_selection library, 80% of the whole data set was chosen as training data and 20% as test data.

2.4 Artificial Neural Networks

The ANNs process the incoming data, compare it with the previously trained information and calculate how much the trained information and the incoming data correspond to each other. The success of the system increases in direct proportion to the success of introducing the examples related to the subject or problem. The networks are trained with pre-selected examples that have the characteristics of the problem to be defined. The response of the system is then reviewed. Training continues until the desired accuracy is reached. The networks increase their accuracy by using the information they learn not only during the training phase but also when it is used (Dongare, Kharde & Kachare, 2012).

An ANN consists of three segments including input, hidden, and output layers. Input layer is the input variables presented to model. Hidden layers are the collection of data processing units followed by y, output, layer. Data processing elements in the hidden layers process the data reaching the network and provide the desired outputs. Results vary depending on the characteristics and weights of the data processing units and their interconnections (Anitescu, Atroshchenko, Alajlan & Rabczuk, 2019). This process can be mathematically expressed and is found by equation 2.6.

$$y = f\left(\sum_{i=1}^{m} w_i\, x_i + b\, w_i\, x_i + b\right) \tag{2.6}$$

Where; wi is the weight vector, i is the number of the vectors (inputs), xi is the input vector, b is the bias, f is the transfer function and y is the output.

All models were developed in the Jupyter Notebook web-based Python program. Libraries run in DT model are pandas, numpy, sklearn.model_selection, DecisionTreeClassifier, sklearn.metrics, matplotlib and seaborn packages. In the ANN deep learning model, 80% of the dataset as training, 10% as test, and the remaining 10% as validation data was used. Libraries run in the ANN model are absolute_import, division, print_function, unicode_literals, tensorflow, tensorflow.keras, sklearn.model_selection, seaborn, tf.keras.utils.to_categorical, tf.keras.Sequential, tf.keras.layers.Dense, tf.keras.losses.BinaryCrossentropy, matplotlib packages.

3. Results and Discussion

The data obtained from the sensors were recorded temporally in Microsoft Excel format on the internet. Noisy data was eliminated with data pre-processing steps on Excel, and then the data set was converted to csv format. With this format, Decision Tree and Artificial Neural Networks models were created in the online interface of Jupyter Notebook. In both methods, alternative models were created with different parameters. The model with the lowest error between the dates of 14.01.2021-23.02.2021 and the combination including all sensor data were finally selected. In addition, maximum depth success was tested according to the new data sets in the study. The relationship between the data received from the sensors and the rain class outputs is given in Figure 6.

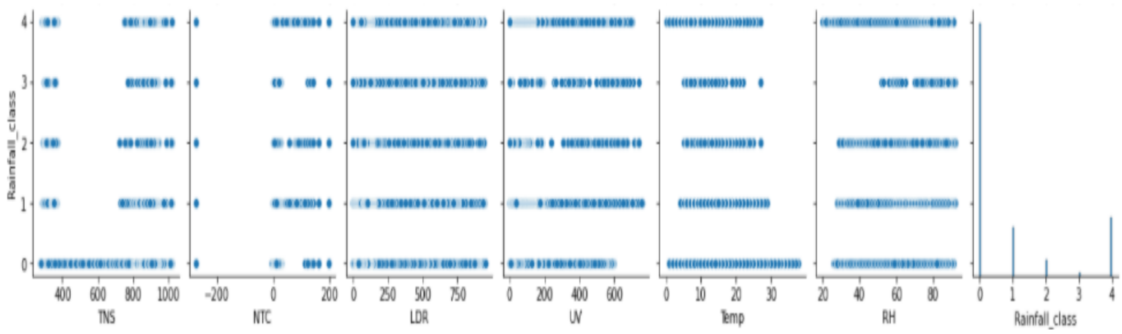


Figure 6. Relationship between sensor data and precipitation classes

As can be seen in Figure 6, the precipitation class mostly coincided with 0 and 4 classes during the measurement period. The reason for this is that the device is sensitive to wet and dry conditions. Thus, it will contribute to the determination of the precipitation intensity and later to increase the accuracy score in the modelling phase.

For the Decision Tree Model, all attribute values are defined in variable X of the model, and rainfall_class values are defined in variable y, as provided in Table 1. The DT is formed as given in Figure 7.

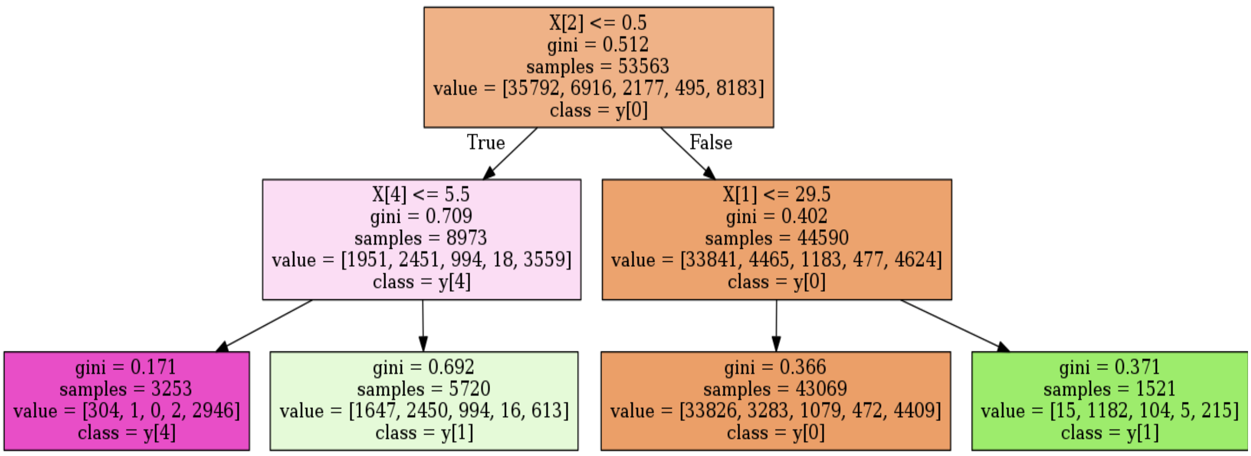


Figure 7. The schematic representation of the decision tree

Gini represents the probability of occurrence of the class, sample values represent the number of samples, and class represents the classification. The reason why the root of the tree starts from the 0 class in the Figure 7 is that there is a precipitation parameter that corresponds to the 0 class at most in the data set. The lowest gini values of the 0 and 4 classes in the leaves of the tree structure show that these classes are correctly separated in the model according to their frequencies.

In comparing the classification algorithms included in the test set, the performance report is shown in Table 4. The performance of the Decision Tree model between precipitation forecast outputs and actual values is compared. The resulting confusion matrix is given in Figure 8.

Table 4

Model report of decision tree

Accuracy	Precision	Recall	F-1 Score	Model Score	Cohen-Kappa Score
0,94	0,98	0,97	0,97	0,96	0,99

As can be seen from Table 4, the performance indicators of the model, such as accuracy, precision recall F-1 score and cohen score, all took high values between 0.94-0.99, showing that the model gave successful results in classification.

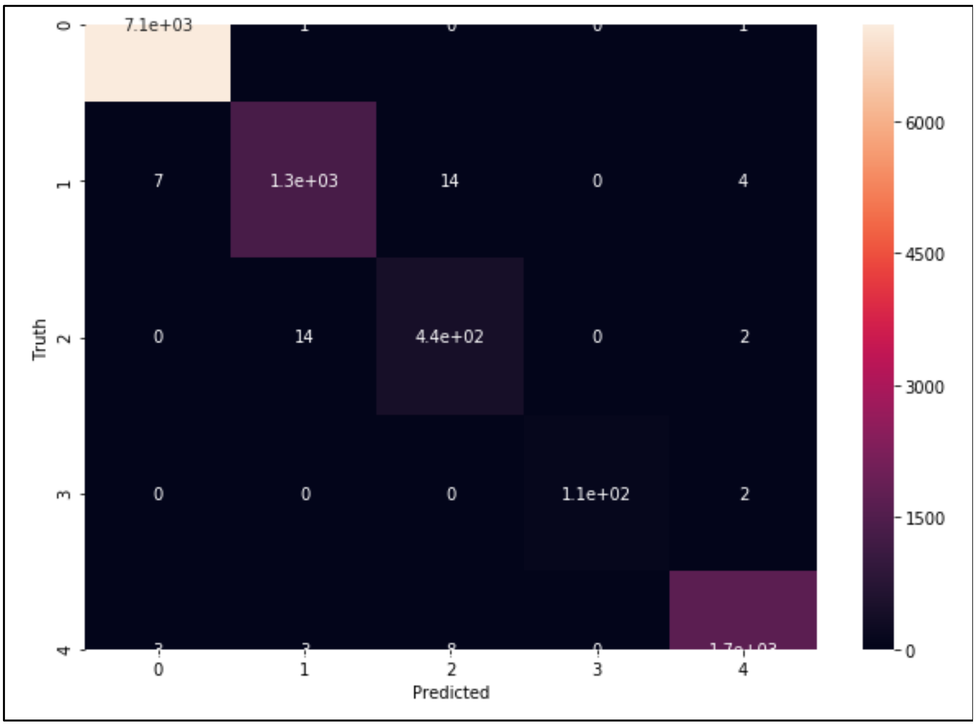


Figure 8. The confusion matrix of the model

The performance of the Decision Tree model between precipitation predicted outputs and truth values is compared. Precipitation forecasts in classes 0, 1 and 4 with the highest frequencies were also correctly reflected in the confusion matrix metric, showing that the classification performance of the model was high.

ANN deep learning classification was made according to the classes in Table 1. Four hidden layers with varying numbers of neurons were created. The number of neurons at each hidden layer and activation functions of the layers are given in Figure 9. These parameters were selected based on trial and error method. The combination of model parameters that yielded the best accuracy was selected.

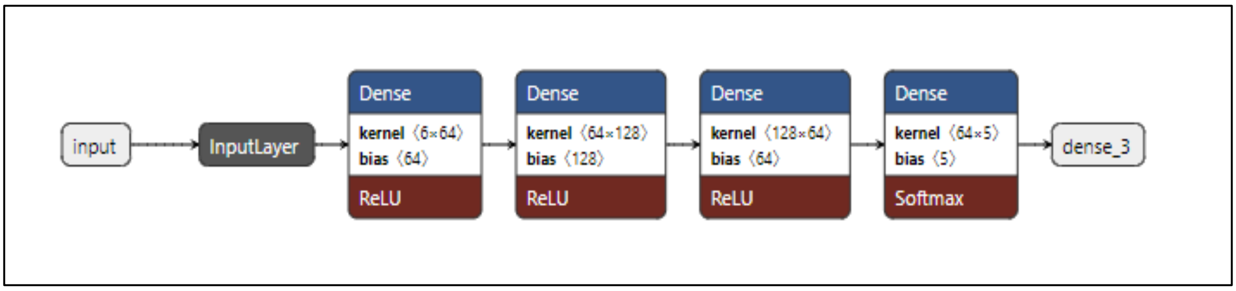


Figure 9. Neural network structure

In the fitting phase of the model, the epoch number of 100 and BinaryCrossentropy are chosen as the loss function. In addition, the data were distributed in 32 groups. The relationship of training and validation data to epoch and accuracy rates is shown in Figure 10.

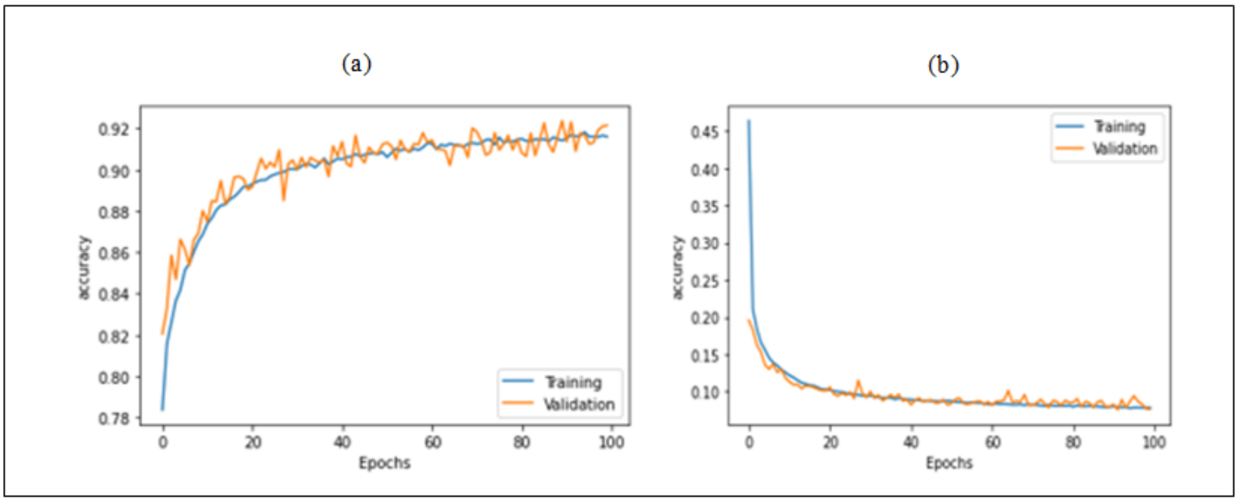


Figure 10. The relationship of training and validation values with accuracy and epoch number (a), The relationship of training and validation loss values to accuracy and epoch number (b)

Epoch is the number of iterations during the training phase of the model. As can be concluded from Figure 10, around 100 epochs the accuracy of the model becomes consistent at 0.92. Obtained accuracies were determined using test data. Comparison table with similar studies was given in Table 5.

Table 5

Comparison table with similar studies

Author	Gagne, McGover, & Xue (2014)	Huang, Lin, Huang & Xing (2017)	Young & Liu (2015)	Manandhar, Dev, Lee, Meng & Winkler (2019)	Ortiz-Garcia, Salcedo-Sanz & Casanova-Mateo (2014)	Shah, Garg, Sisodiya, Dube & Sharma (2018)
Dataset	Center for Analysisand Prediction of Storms (CAPS) 2010 SSEF system	Beijing Meteorological Information Center	Water Resources Agency, Taiwan	The weather station in Singapore	Madrid-Barajas International Airport automatic station	National Centers for Environmental Prediction
Year	2014	2014	2015	2019	2014	2018
Study Group (Model)	Logistic Regression and Random Forest	K-nearest neighbor (KNN)	ANN	Support Vector Machine (SVM)	Support Vector Machine (SVM)	Random Forest
Accuracy	0.78 and 0.77	0.49	0.92	0.80	0.80	0.71

As seen in Table 5, precipitation forecasts were made using different datasets and models. As a result of 6 studies, the highest accuracy was found to be 0.92 in Young & Liu (2015). When the said study and this study were compared, the Decision Tree model, which had a higher model score, was found to be 0.96.

4. Conclusion

Developing sensor and artificial intelligence technologies provide many conveniences in agriculture and environmental sectors as well as in other sectors. In this context, sensing environmental parameters with sensors, processing and making them meaningful with artificial intelligence helps in solving many agricultural problems. Agriculture is a production system that is largely dependent on natural conditions. Therefore, it is vital to measure natural or environmental factors and store these measurements in a database. The most important of these environmental factors are temperature, relative humidity and precipitation. Another important issue in agricultural studies is the development of methods, models or systems that will provide estimation of these environmental factors. In this study, a portable weather station is developed that will measure the main climatic parameters affecting agricultural production systems, transfer these data via wireless systems and store them in the cloud database. It is aimed develop a system and the database working together and also low cost. In addition, meteorological data obtained from the developed prototype system are evaluated in popular artificial intelligence models and precipitation forecasts are made. At the end of the study, The DR score was 0.96 and the ANN score was 0.92. The obtained accuracies were reached by using the test data. The results of the research revealed that both ANN deep learning and DT models can predict precipitation using the aforementioned sensor data. It has been demonstrated that the devices and models developed with this study can make precipitation predictions with high scores and instantly. In the next stage, it is aimed to increase the number and quality of sensors in the developed system, to collect data in a longer period and to make it more functional by integrating it with irrigation systems.

Author Contributions

Hakkı Fırat Altınbilek: Performed ANN and DT analysis.

Hakan Nar: Collected data.

Sefa Aksu: Designed the hardware component.

Ünal Kızıl: Planned the study wrote the manuscript.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

Abyaneh, H.Z., Varkeshi, M.B., Golmohammadi, G., & Mohammadi, K. (2016). Soil temperature estimation using an artificial neural network and co-active neuro-fuzzy inference system in two different climates. *Arabian Journal of Geosciences, 9(5),* 377. https://doi.org/10.1007/s12517-016-2388-8

Anitescu, C., Atroshchenko, E., Alajlan, N., & Rabczuk, T. (2019). Artificial neural network methods for the solution of second order boundary value problems. *Computers, Materials & Continua, 59(1),* 345–359. https://doi.org/10.32604/cmc.2019.06641

Ardiansyah, A. Y., Sarno, R., & Giandi, O. (2018). Rain detection system for estimate weather level using Mamdani fuzzy inference system. *International conference on information and communications technology* (pp. 848-854).Yogyakarta, Indonesia. Retrieved from: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8350711

Bhagat, S.K., Tung, T.M., & Yaseen, Z.M. (2020). Development of artificial intelligence for modeling wastewater heavy metal removal: state of the art, application assessment and possible future research. *Journal of Cleaner Production, 250,* 119473. https://doi.org/10.1016/j.jclepro.2019.119473

Bochtis, D., Liakos, G.K., Busato, P., Moshou, D., & Pearson, S. (2018). Machine learning in agriculture: a review. *Sensors,* 18(8)*,* 2674. https://doi.org/10.3390/s18082674

Bounsaythip, C., & Esa, R.R. (2001). Overview of data mining for customer behavior modeling. *VTT Information Technology Research Report,* 1, 1-53.

Covert, T. M., & Thomas, J. A. (1991). *Elements of information theory.* New York: Wiley & Sons.

Dongare, A.D., Kharde, R.R., & Kachare, A.D. (2012). Introduction to artificial neural network. *International Journal of Engineering and Innovative Technology, 2(1),* 189-194. Retrieved from: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.1082.1323&rep=rep1&type=pdf

Durrani, A., Khurram, M., & Khan, H. R. (2019). Smart weather alert system for dwellers of different areas, in Proc. *16th International Bhurban Conference Applied Sciences & Technology* (pp. 333-339). Islamabad, Pakistan. https://doi.org/10.1109/ıbcast.2019.8667190

Gagne, D. J., McGovern, A., & Xue, M. (2014). Machine learning enhancement of storm-scale ensemble probabilistic quantitative precipi- tation forecasts. *Weather and Forecasting, 29(4)*, 1024-1043. https://doi.org/10.1175/WAF-D-13-00108.1

Huang, M., Lin, R., Huang, S., & Xing, T. (2017). A novel approach for precipitation forecast via improved K-nearest neighbor algorithm. *Advanced Engineering Informatics*, 33, 89–95. https://doi.org/10.1016/j.aei.2017.05.003

Hwang, J., Orenstein, P., Cohen, J., Pfeiffer, K., & Mackey, L. (2019). Improving subseasonal forecasting in the western U.S. with machine learning. *The 25th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD'19)* (pp. 2325-2335). New York, USA. https://doi.org/10.1145/3292500.3330674

Joshi, P., Mistry, N., Khan, A., Motekar, H., & Chaugule,A. (2021). Weather data accumulation using Arduino. *International Journal of Innovative Research in Computer and Communication Engineering.* 9(4), 1880-1887.

Kumari, S., Kasliwal, M. H., & Valakunde, N. D. (2018). An android based smart environmental monitoring system using IoT. In Singh M., Gupta P., Tyagi V., Flusser J., & Ören T. (Eds.), *Advances in Computing and Data Sciences. ICACDS 2018. Communications in Computer and Information Science, vol 906* (pp. 535-544). Springer, Singapore. https://doi.org/10.1007/978-981-13-1813-9_53

Manandhar S., Dev, S., Lee, Y.H., Meng, Y.S., & Winkler S. (2019). A data-driven approach for accurate rainfall prediction. *IEEE Transactions on Geoscience and Remote Sensing,11,*9323-31.

Mehr, A. D. (2021). Drought classification using gradient boosting decision tree. *Acta Geophysica*, *69*, 909–918. https://doi.org/10.1007/s11600-021-00584-8

Mohapatra, G., Rakesh, V., Purwar, S., & Dimri, A.P. (2021). et al. Spatio-temporal rainfall variability over different meteorological subdivisions in India: analysis using different machine learning techniques. *Theoretical and Applied Climatology,* 145(1-2), 673-686. https://doi.org/10.1007/s00704-021-03644-7

Murray J., & Mackinnon, N.G. (1999). Data mining and knowledge discovery in databases- an overview. Australian and New Zeland *Journal of Statistics, 41*(3), 255-275. p.260. https://doi.org/10.1111/1467-842X.00081

Ortiz-García, E.G., Salcedo-Sanz, S., & Casanova-Mateo, C. (2014). Accurate precipitation prediction with support vector classifiers: A study including novel predictive variables and observational data. *Atmospheric Research*, *139*, 128–136. https://doi.org/10.1016/j.atmosres.2014.01.012

Pal, M., & Mather, P.M. (2003). Support vector machines for classification in remote sensing. *International Journal of Remote Sensing, 26,* 1007-1011. https://doi.org/10.1080/01431160512331314083

Shah, U., Garg, S., Sisodiya, N., Dube, N., & Sharma, S. (2018). Rainfall Prediction: Accuracy Enhancement Using Machine Learning and Forecasting Techniques. *Fifth International Conference on Parallel, Distributed and Grid Computing (PDGC)* (pp.776-782). Solan, India. Retrieved from: https://ieeexplore.ieee.org/document/8745763

Sirohi, K., Tanwar, A., Himanshu, Jindal, P. (2016). Automated irrigation and fire alert system based on Hargreaves equation using weather forecast and Zigbee protocol. *IEEE 2nd International Conference on Communication, Control and Intelligent Systems (CCIS)* (pp. 13-17). Mathura, India. https://doi.org/10.1109/CCIntelS.2016.7878191

Teanby, N. A. (2006). An icosahedron-based method for even binning of globally distributed remote sensing data. *Computers & Geosciences*, *32(9), 1442–1450*. https://doi.org/10.1016/j.cageo.2006.01.007

Ucgun, H., & Kaplan, Z.K., (2017). Arduino Based Weather Forecasting Statio. *2nd International Conference on Computer Science and Engineering* (pp. 972-977). Guilin, China. https://doi.org/ 10.1109/UBMK.2017.8093397

Tian, X., & Chen, M. (2021). Descriptor selection for predicting interfacial thermal resistance by machine learning methods. *Scientific Reports*, *11*, 739 (2021). https://doi.org/10.1038/s41598-020-80795-z

Valada, A., Kohanbash, D., & Kantor, G.A. (2010). Design and Development of a Wireless Sensor Network System for Precision Agriculture. *Technical Report:CMU-RI-TR-10- 21, Robotics Institute, Carnegie Mellon University, USA*. Retrieved from: https://www.ri.cmu.edu/publications/design-and-development-of-a-wireless-sensor-network-system-for-precision-agriculture/

Yeditha, P. K., Kasi, V., Rathinasamy, M., & Agarwal, A. (2020). Forecasting of extreme flood events using different satellite precipitation products and wavelet-based machine learning methods. *Chaos: An Interdisciplinary Journal of Nonlinear Science, 30(6),* 063115. https://doi.org/10.1063/5.0008195

Young, C.-C., & Liu, W.-C. (2015). Prediction and modelling of rainfall–runoff during typhoon events using a physically-based and artificial neural network hybrid model. *Hydrological Sciences Journal*, *60*(12), 2102–2116. https://doi.org/10.1080/02626667.2014.959446

Zhang, N., Wang, M., & Wang, N. (2002). Precision agriculture-a worldwide overview. *Computers and Electronics in Agriculture, 36(2-3),* 113–132. https://doi.org/10.1016/S0168-1699(02)00096-0