

**DATA ANALYSIS OF ENVIRONMENTAL CONDITIONS  
INFLUENCING THE WORK OF LABORATORY EQUIPMENT  
AND APPLICATION OF MACHINE LEARNING MODELS FOR  
CLASSIFICATION OF POOR CONDITIONS**

*UDC ((004.3/.4:0.034.2)+004.78)*

**Andela Đorđević, Miroslav Milovanović, Marko Milojković,  
Miodrag Spasić**

University of Niš, Faculty of Electronic Engineering, Department of Control Systems,  
Republic of Serbia

**Abstract.** *Environmental conditions can have a crucial impact on the functioning of laboratory equipment. Electric components are sensitive to the influence of certain environmental factors such as temperature, humidity, vibrations, etc. Environmental factors should, therefore, be monitored to avoid their negative influence on the system and potential faults and failures they could cause. Unlike the traditional approaches which required the presence of special staff to monitor environmental factors and react if they are poor, the rise of the Internet of Things enhanced the application of intelligent solutions where human factor is not necessary. In this paper, research on data analysis, preprocessing and intelligent classification of environmental conditions has been conducted. The data was collected by sensors connected to Raspberry Pi. The applied monitoring system setup enabled long-distance monitoring of laboratory conditions through the internet and full applicability of fundamental IoT concepts. Since data preparation is an important step in the process of designing machine learning models, the collected data was analyzed and preprocessed in Python. Intelligent classification of environmental conditions was performed using machine learning models *k*-Nearest Neighbors and Random Forest. Grid search was used for model selection, and the performances of *k*-Nearest Neighbors and Random Forest machine learning models were compared. Experimental results show that these machine learning models can be successfully used for intelligent classification of environmental conditions.*

**Key words:** *Environmental conditions, Raspberry Pi, Internet of things, Data analysis, Machine learning*

---

Received November 18, 2022 / Accepted December 21, 2022

Corresponding author: Andela Đorđević

University of Niš, Faculty of Electronic Engineering, Department of Control Systems, Aleksandra Medvedeva 14,  
18000 Niš, Republic of Serbia

E-mail: [andjela.djordjevic@elfak.ni.ac.rs](mailto:andjela.djordjevic@elfak.ni.ac.rs)

## 1. INTRODUCTION

Environmental conditions can have a crucial impact on the operation of dynamic systems. Electric components are fundamental units of laboratory equipment, and as such, they can be sensitive to the influence of various external factors. Some of the most important factors to be monitored and controlled are high or low temperatures, humidity, vibrations, air pressure, etc. Beside affecting the functioning of the system, poor environmental factors can cause serious damage on the equipment [1]. For example, high or low temperatures can cause alterations of the electrical component's properties, which could lead to failures and faults of electronic devices. [2]. Also, humidity affects almost every kind of electrical equipment by causing corrosion, changes in electrical resistance, thermal conductivity, capacitance, etc. [2-4]. The effects of vibration should not be neglected, since they can lead to great physical damage on dynamic systems [1,5,6]. Therefore, monitoring and controlling environmental conditions is of crucial importance.

In order to detect undesirable environmental factors, traditional approaches used isolated monitoring units. The main drawback of this approach was the fact that these units were not integrated as a part of a bigger interconnected system. Interconnected intelligent systems of sensors and actuators, which can be applied for monitoring environmental factors, are becoming more and more common with the rise of Internet of Things (IoT) [7-9]. These intelligent approaches give an opportunity to avoid employing special staff to monitor environmental factors and react when the conditions are not satisfying. For example, Usamentiaga et al. proposed a special Industrial Internet of Things architecture for temperature monitoring and fault detection using infrared thermography [7], while Tu et al. proposed using Raspberry Pi to collect data on environmental conditions in a laboratory through various types of sensors and give an adequate response if the conditions are poor [8]. Another interesting research on IoT system applied for environment monitoring and control, can be found in [9]. The authors in [10] conducted research on intelligent classification of environmental conditions. Raspberry Pi was used to collect the data, which was later classified using machine learning models, such as Support Vector Machines, Logistic Regression,  $k$ -Nearest Neighbors and Decision Tree Classifier [10]. Machine learning models are commonly used as a tool for finding hidden patterns in large datasets, data classification, data mining, etc. Data analysis and preprocessing is an important step in the process of designing machine learning models [11]. Poorly conducted data preparation and analysis can lead to misleading results; thus, this step should be done with special care [11]. In this paper, a detailed analysis of the dataset created using Raspberry Pi was conducted, and the data was later classified using machine learning models Random Forest Classifier and  $k$ -Nearest Neighbors. In order to find the optimal parameters for each machine learning model, grid search was applied for model selection.

This paper is organized as follows. In Section 2, the description of an IoT system used for monitoring of environmental conditions in a laboratory is given. Also, a short description of Raspberry Pi and the sensors used for data acquisition is presented. Further information on sensing and data acquisition is given in Section 3. Data analysis is described in Section 4 in details, while Section 5 presents approaches used for intelligent classification of optimal and poor conditions. Experimental results are given in Section 6. Accuracy, number of false positives and number of false negatives of machine learning models are compared in this section. Experimental results show that both Random Forest Classifier and  $k$ -Nearest Neighbors can be successfully used for intelligent classification of environmental conditions. The authors give concluding remarks and discuss the future direction of their research in the last section.

## 2. IOT SYSTEM FOR DETECTION OF ENVIRONMENTAL CONDITIONS IN A LABORATORY

### A. Raspberry Pi.

Raspberry Pi is a credit card sized microcomputer. The experiments were conducted on Raspberry Pi 3 B+, which has 1.4GHz 64-bit quad-core processor, dual-band wireless LAN, Bluetooth 4.2/BLE, faster Ethernet, and Power-over-Ethernet support (with separate PoE HAT) [12]. Raspberry Pi 3 B+ has Extended 40-pin GPIO header, Full-size HDMI, 4 USB 2.0 ports, CSI camera port for connecting a Raspberry Pi camera, DSI display port for connecting a Raspberry Pi touchscreen display, 4-pole stereo output and composite video port [12]. It has the opportunity of attaching various sensors, collecting data from them and sending it to a server.



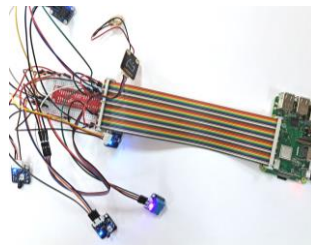
**Fig. 1** Raspberry Pi 3B+

### B. Sensors

In order to monitor environmental factors such as humidity, temperature, vibration, magnetic field, light intensity and fire detection, five sensors were used: Sunfounder Humiture Sensor (humidity and temperature sensor), DFRobot Digital Piezo Disk Vibration Sensor, Sunfounder Analog Hall Sensor, Sunfounder Photoresistor and Sun-founder Flame sensor.

### C. Environmental Conditions Monitoring System

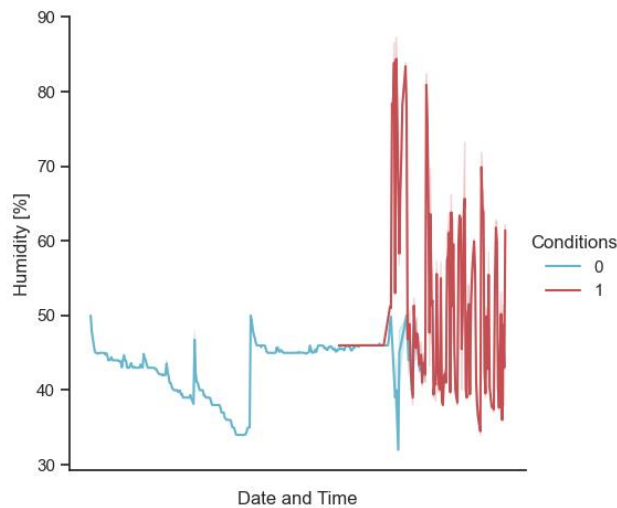
The hardware of the environmental conditions monitoring system consists of Raspberry Pi and sensors attached to it to collect data. Raspberry Pi is connected to the local laboratory computer utilized as a mediator between the device and sensors on one side and any computer with access to the local computer on the other. Such setup enables long-distance monitoring of laboratory conditions through the internet and full applicability of fundamental IoT concepts by the realized technical solution. The data analysis and preprocessing were done in Python, and intelligent classification of environmental conditions was performed using machine learning models.



**Fig. 2** System used for detection of environmental conditions

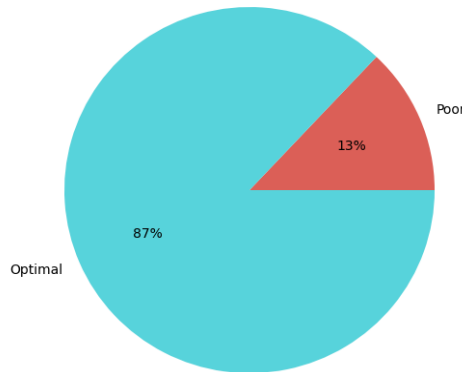
### 3. SENSING AND COLLECTING DATA

In order to create a complete environmental conditions dataset, firstly, information about environmental conditions was recorded when all the conditions were optimal. Secondly, after enough data samples about optimal conditions were collected, the disturbances in the form of poor environmental factors were introduced, concretely in the form of increased humidity and vibration levels. The datasets were then integrated in one unique dataset, which contains information on both optimal and poor conditions. During the recording, sampling time was 0.1 seconds. In total, 12872 samples of data were collected from which 11289 for optimal and 1683 for poor conditions. Fig. 3 shows the information from the humidity sensor throughout time. The blue line in Fig. 3 refers to the period of time when conditions were optimal, while the red line refers to the time period when disturbances were introduced.



**Fig. 3** Information from humidity sensor throughout time

Created dataset included information about the date and time when samples were recorded. Environmental factors are represented through *Humidity [%]* (containing information about humidity level), *Temperature [°C]* (temperature in degrees Celsius), *Vibration* (strength of vibration), *Hall* (strength of magnetic field), *Photoresistor* (information about light intensity) and *Flame* (existence of fire) columns. The *Conditions* column is of the main interest for the research objectives in this paper and represents the status of environmental conditions in a laboratory. Values in this column were added manually depending on environmental factors. If the values of environmental factors were in optimal range, the conditions were marked as “*optimal*”. On the other hand, if some of the environmental factor’s value was not in the optimal range, the conditions were marked as “*poor*”. Values in this column were later mapped in Python with numeric values 0 and 1 respectively. The graphic visualization of the values in *Conditions* column is presented with pie plot in Fig. 4.



**Fig. 4** Pie plot of the decision variable

The pie plot of the decision variable shows that there are much more rows with value “*optimal*” than “*poor*”. This means that significantly more information on factors’ values matching optimal than poor environmental conditions were collected. This disproportion could potentially make the classification task harder for some of the machine learning models.

#### 4. DATA ANALYSIS IN PYTHON

After the data had been collected, the dataset was analyzed and preprocessed in Python. In order to gain an insight into data types and number of rows with Non-Null objects, *Pandas dataframe.info()* function was used. This function was used to find out whether there were some missing values in the dataset. The names of the variables, number of Non-Null objects in them and their data types are presented in Table 1.

**Table 1** Names, number of Non-Null objects and data types of the variables

	Feature	Non-Null count	Data type
0	Date and Time	12972	Object
1	Humidity [%]	12972	int64
2	Temperature [°C]	12972	int64
3	Vibration	12972	float64
4	Hall sensor	12972	int64
5	Photoresistor	12972	int64
6	Flame	12972	int64
7	Conditions	12972	int64

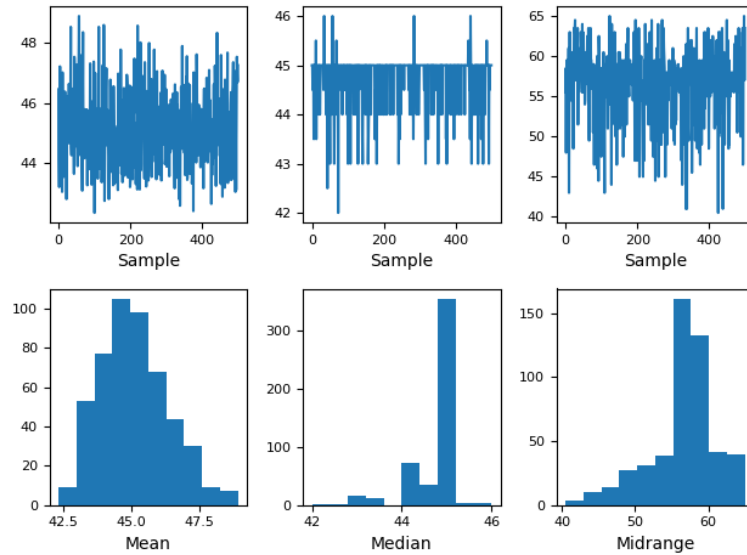
Data types: float 64 (1), int64 (6), object (1)  
Memory usage: 810.9+ KB

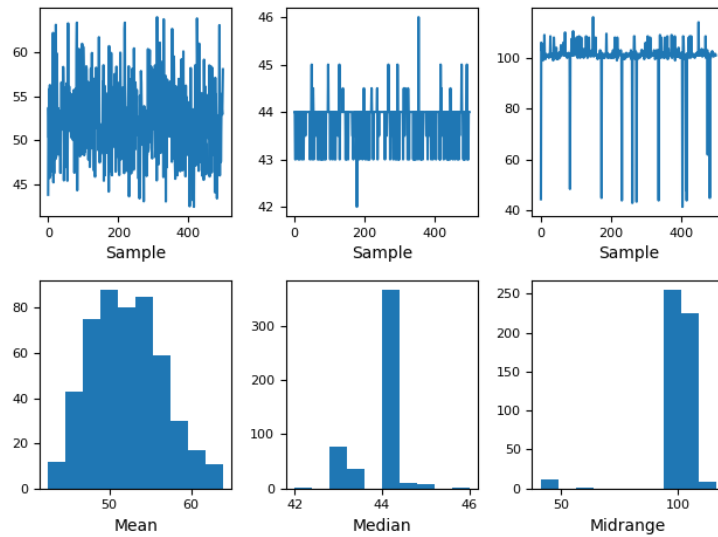
Number of samples, mean value, standard deviation, minimum and maximum value, the median, 25<sup>th</sup> and 75<sup>th</sup> percentile for each feature, were obtained using *Pandas dataframe.describe()* function. The results of this function are shown in Table 2.

**Table 2** Number of samples, mean value, standard deviation, min. value, 25<sup>th</sup> percentile, median, 75<sup>th</sup> percentile and maximum value of the features.

	Number of samples	Mean value	Standard deviation	Min. value	25%	50%	75%	Max. value
Humidity	12972	45.107	8.442719	32	41	45	46	95
Temperature	12972	25.303	1.330145	21	24	25	27	32
Vibration	12972	0.0868	0.451454	0	0	0	0	5
Hall	12972	134.001	0.055518	132	134	134	134	138
Photoresistor	12972	52.0438	30.87137	35	42	44	46	196
Flame	12972	0.00023	0.015206	0	0	0	0	1
Conditions	12972	0.12974	0.336031	0	0	0	0	1

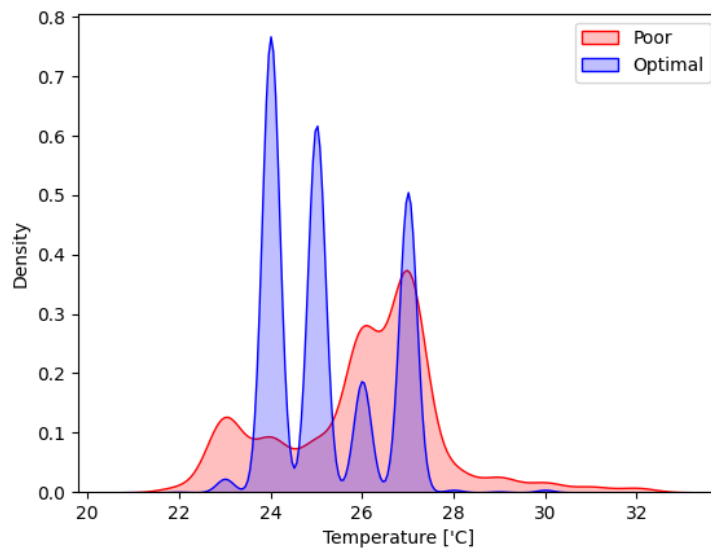
To check the uncertainty of statistics and choose the right representation for the features, bootstrap plots can be used. If the mean value of a feature differs from the median, bootstrap plot could give a clue which estimator of the two would be the superior location estimator. Based on the values in Table 2, all the features had almost the same locations of the mean and the median, except the values of *Photoresistor* features. In the case of this feature, the mean value was 52.0438, while the median value was 44. The bootstrap plots of *Humidity* and *Photoresistor* features are presented in Fig. 5 and Fig. 6 respectively. The bootstrap plot of the *Photoresistor* shows that the median has the smallest variance and would, therefore, be the superior location estimator.

**Fig. 5** Bootstrap plot of *Humidity* [%]



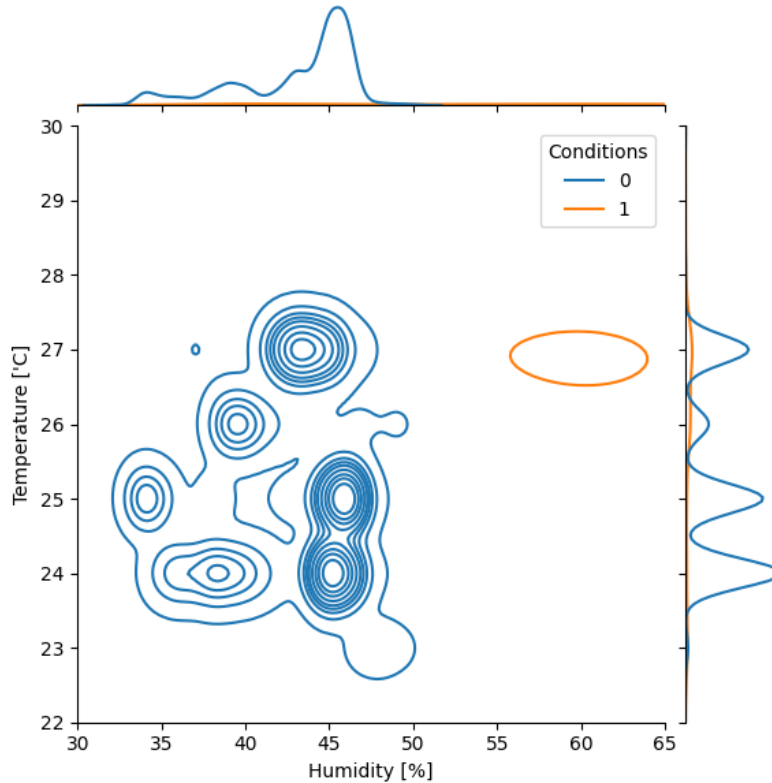
**Fig. 6** Bootstrap plot of *Photoresistor*

Data distribution is an important thing to consider when analyzing the dataset. In order to check the distribution of the features, distribution plots were examined for each of the features (for example, Fig. 7 shows the distribution plot of temperature).



**Fig. 7** Distribution plot of *Temperature*: the blue line represents the distribution of optimal conditions, while the red line represents the distribution of poor conditions.

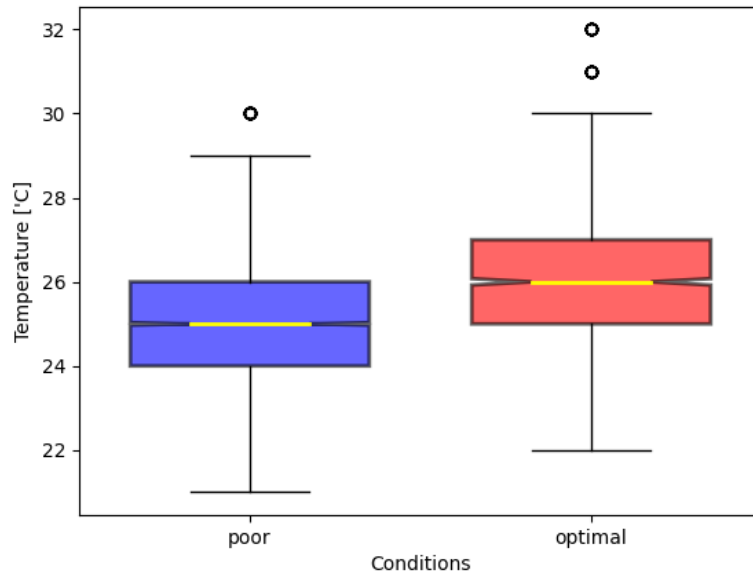
The figure clearly shows that the information from the temperature sensor does not follow normal distribution. Another thing which can be concluded based on the distribution plot of the temperature feature, is the fact that it would be hard to classify environmental conditions simply based on this parameter, since the blue area representing optimal conditions and the red area representing poor conditions are both not following normal distribution, and they overlap. This can also be seen on the Humidity-Temperature joint plot (Fig. 8)



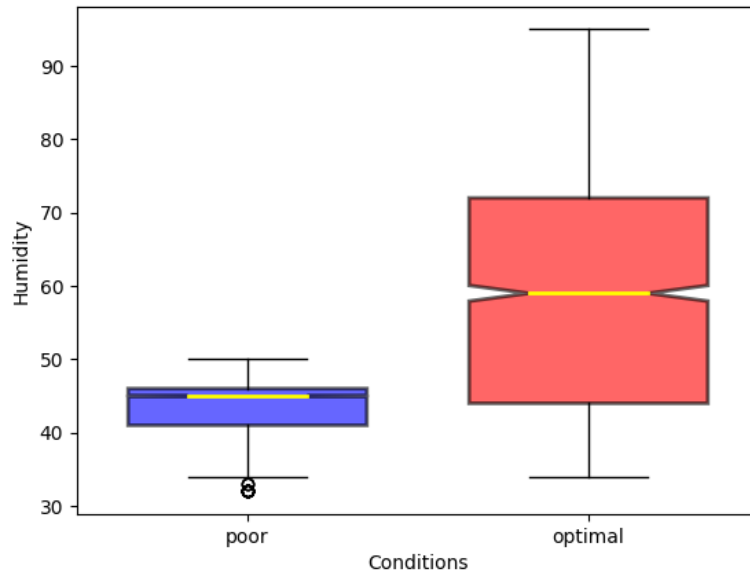
**Fig. 8** Joint plot of *Humidity* and *Temperature*

In order to locate outliers, boxplots were used. Boxplots graphically represent the distribution of the data through minimum and maximum value, the first quartile, median and the third quartile. If outliers are present in the data, that implies that some unusual values, which significantly differ from the rest of the dataset are present. In this case, outliers would imply that some environmental factors had unexpected values. Figures 9 and 10 represent boxplots of *Temperature* and *Humidity*.





**Fig. 9** Boxplot of *Temperature*



**Fig. 10** Boxplot of *Humidity*

Since the distribution of the data was not normal, in order to check the exact values and locations of outliers in the columns, interquartile range method was used [13,14]. As expected, the most outliers were found in humidity and vibration data, while the number of outliers for the rest of the features was insignificant.

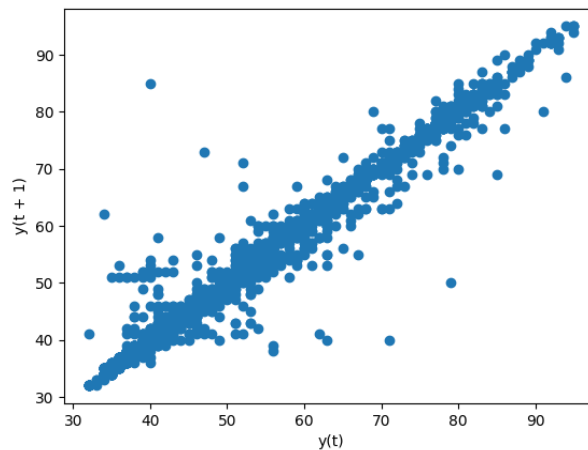
In the end, the mutual relations between the variables were analyzed. In order to examine the influence of the parameters on the decision variable, as well as relationships between columns, correlation heatmap function was used to find the correlation index. Correlations between the features and the decision variable are presented in Table 3.

**Table 3** Correlation between the decision variable and features.

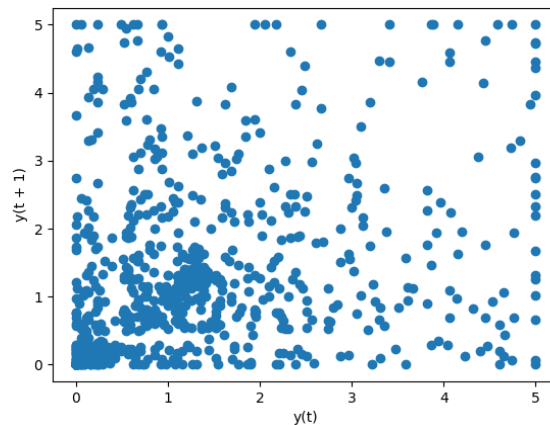
	Humidity	Vibration	Temperature	Hall	Flame	Photoresistor
Conditions	0.64	0.48	0.2	0.058	0.039	-0.038

The correlation between the data acquiesced from the photoresistor and environmental conditions was insignificant. Also, information from this sensor was irrelevant for the research, and therefore, this feature was dropped from the dataset. Even though the correlation between the data acquiesced from the flame and analog hall sensor and the decision variable was also small, these features were not dropped from the dataset since they can contain useful information about the conditions. For example, flame feature gives an insight on whether a fire exists in a laboratory. If it does, this automatically means that the conditions are poor, which implies that, in reality, the data from flame sensor directly affects the decision variable. Two features with the highest correlation with the decision variable were *Humidity [%]* and *Vibration*.

Finally, autocorrelation of the features was examined. If autocorrelation exists, that means that there is a relationship between the value of a feature at a certain point in time, and its past points. The variables in which autocorrelation was present were *Humidity* and *Temperature* (this was expected because the transition between higher and lower temperatures, for example, cannot be immediate). In order to illustrate this, lag plots of *Humidity*, in which autocorrelation is present, and lag plot of *Vibration* feature, which is much more random, are presented in Fig. 11 and Fig. 12.



**Fig. 11** Lagplot of *Humidity*



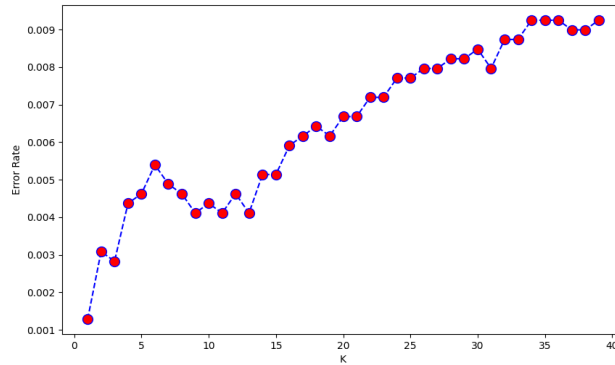
**Fig. 12** Lagplot of *Vibration*

## 5. INTELLIGENT CLASSIFICATION OF ENVIRONMENTAL CONDITIONS

After the data was analyzed and preprocessed, training and test datasets for machine learning models were formed. Training dataset consisted of 70% of total data samples. After fitting the models, their performance was tested on the remaining 30% of dataset. The machine learning models used in the experiments were  $k$ -Nearest Neighbors and Random Forest Classifier. Both  $k$ -Nearest Neighbors (KNN) and Random Forest Classifier are popular machine learning models which give satisfying results for classification problems [15-18]. In order to select optimal models, grid search was performed. The performance of KNN algorithm depends heavily on the number of neighbors used in the classification process. The models with lower values of  $k$  used in the classification process tend to lead overfit the data and fail to generalize [18]. Generally, the value of the bias is lower, but the variance is higher than when larger number of neighbors is used. The influence of the number of neighbors on the capability of KNN model to classify the conditions was examined. Grid search was applied to choose the right value of  $k$ . In order to select Random Forest classifier model, attention was focused on the variations of two parameters: number of estimators and minimum number of samples per leaf node. To choose the optimal values of these parameters, grid search was applied.

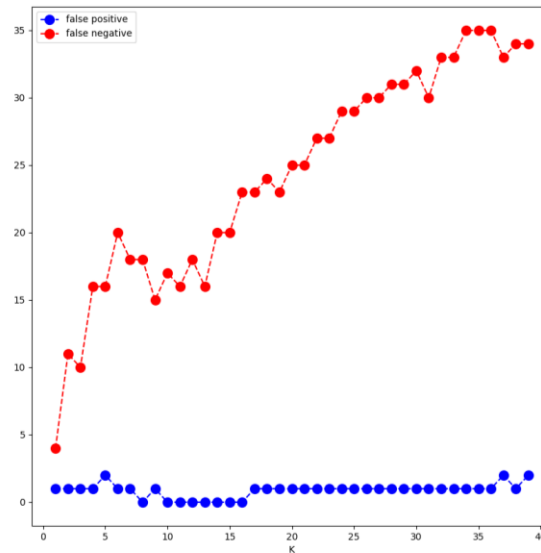
## 6. RESULTS

The prediction of the decision variable “Conditions” was performed based on the values of environmental factors in the test dataset, and the results were compared to the real values. Firstly, grid search was applied in order to set the parameters of the models. Accuracy of the prediction of  $k$ -Nearest Neighbors for different  $k$  values were compared in order to choose the best model. Fig. 14 shows the error rate for different values of number of neighbors.



**Fig. 13** Error rate for different  $k$  values

It can be concluded that the error rate of the models becomes higher as  $k$  increases. In order to investigate this further, the relationship between  $k$  values and the number of false positives (the number of optimal conditions falsely classified as poor) and the number of false negatives (the number of poor conditions falsely classified as optimal) was examined. Fig. 15 shows the number of false positives (blue line) and the number of false negatives (red line) depending on  $k$ .



**Fig. 14** Number of false positives (blue) and number of false negatives (red) for different  $k$  values

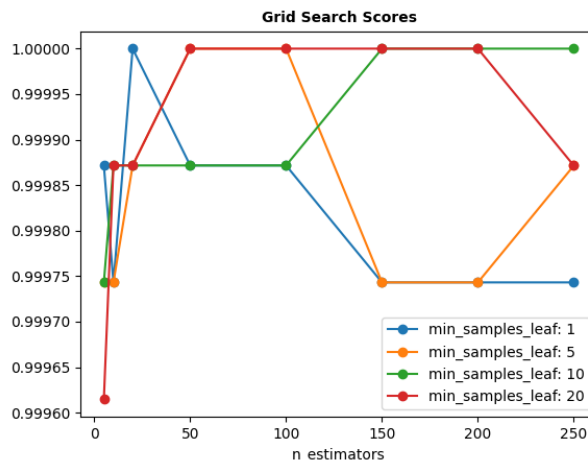
While the number of false positives is smaller than 3 for all  $k$  values, the number of false negatives increases. The figure shows that the algorithm had a bigger problem with classifying poor than optimal conditions. This problem occurs due to the fact that the majority of the collected data referred to the optimal conditions.

The accuracy, number of false positives and number of false negatives of  $k$ -Nearest Neighbors models for  $k = 1, 3, 5, 10, 20, 40$  is presented in Table 4. The best accuracy and the smallest number of false negatives were obtained for  $k = 1$ . Still,  $k = 1$  may not be the optimal solution, since it can lead to overfitting and poor generalization of data. A smaller number of neighbors leads to a smaller bias, but the larger variance. On the other hand, for larger values of  $k$ , variance decreases, but the bias increases. Therefore, a trade-off should be made to avoid overfitting or underfitting the data. Bearing that in mind,  $k = 10$  was chosen as the optimal number of neighbors in this case.

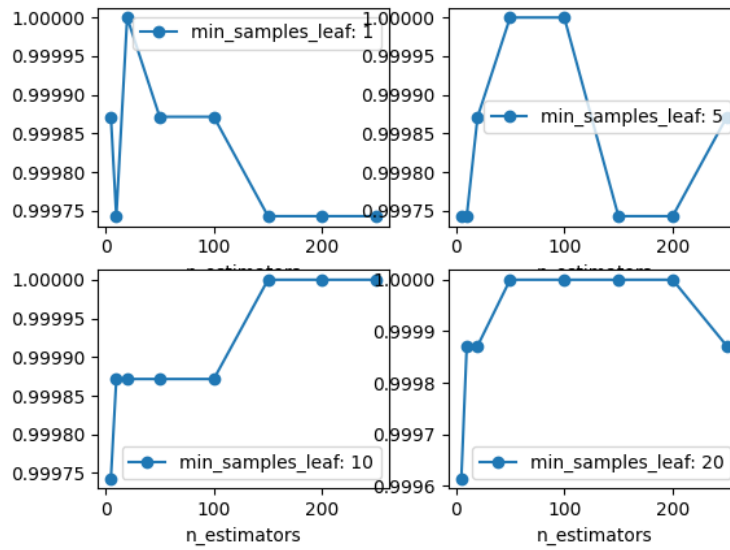
**Table 4** Accuracy, number of true negatives, number of false positives, number of true positives and number of false negatives for different values of  $k$ .

	Accuracy [%]	Number of true negatives	Number of false positives	Number of true positives	Number of false negatives
$k = 1$	99.8715313464	3395	1	492	4
$k = 3$	99.7173689620	3395	1	486	10
$k = 5$	99.5375128469	3394	2	480	16
$k = 10$	99.5632065776	3396	0	479	17
$k = 20$	99.3319630010	3395	1	471	25
$k = 40$	99.0750256937	3394	2	462	34

Another machine learning model used for the prediction was Random Forest Classifier. In this case, grid search was applied to find the optimal number of estimators ( $n\_estimators$ ) and the minimum number of samples per leaf ( $min\_samples\_leaf$ ). Models with  $n\_estimators = 10, 25, 50, 75, 100, 150, 200, 250$  and  $min\_samples\_leaf = 1, 5, 10, 20$  were compared. Fig. 15 and Fig. 16 show the accuracy of different models depending on the number of estimators and minimum samples per leaf, while Fig. 17 shows the mean fitting time.

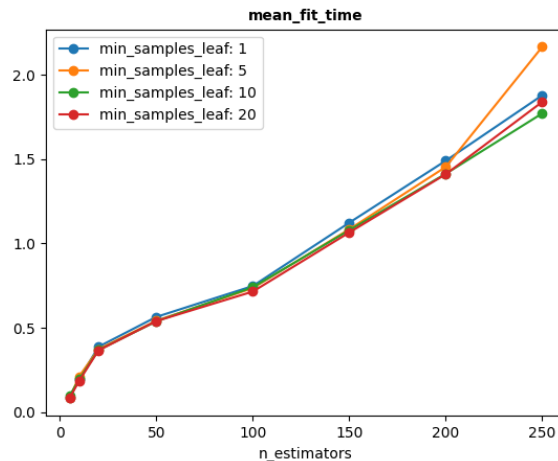


**Fig. 15** Accuracy of Random Forest Classifiers depending on the number of estimators and minimum samples per leaf



**Fig. 16** Accuracy score of Random Forest Classifiers depending on number of estimators for  $\text{min\_samples\_leaf}=\{1,5,10,20\}$

The default value of the number of samples per leaf is one. In general, this number is increased to avoid overfitting the data (tree pruning), and therefore poor prediction. The graph shows a slight decrease of the accuracy of the models with the increase of the number of estimators for  $\text{min\_samples\_leaf}=1$  and  $\text{min\_samples\_leaf}=5$ , while the accuracy of the models with  $\text{min\_samples\_leaf}=10$  and  $\text{min\_samples\_leaf}=20$  slightly increases.



**Fig. 17** Mean fitting time for each of the Random Forest Classifiers

As expected, the fitting time increased as the number of estimators increased, while the number of samples per leaf did not affect it as much.

The model which proved to be the best choice was the one with 50 estimators and one sample per leaf. Table 5 shows the accuracy, number of true negatives, number of false positives, number of true positives and number of false negatives for the best  $k$ -Nearest Neighbors and Random Forest Classifier model.

**Table 5** Accuracy, number of true negatives, number of false positives, number of true positives and number of false negatives for  $k$ -Nearest Neighbors and Random Forest Classifier.

	Accuracy [%]	Number of true negatives	Number of false positives	Number of true positives	Number of false negatives
k Nearest Neighbors (k=10)	99.5632065776	3396	0	479	17
Random Forest Classifier (n_estimators=50, min_samples_leaf=1)	99.9807284641	3396	0	495	1

Even though the accuracy of Random Forest Classifier is only slightly better than the accuracy of  $k$ -Nearest Neighbors,  $k$ -Nearest Neighbors model showed a greater tendency to falsely classify *poor* conditions as *optimal*, which could seriously influence the performance of the equipment and electrical components in the laboratory. Because of that tendency and since it falsely classified only one poor condition as optimal, Random Forest Classifier would be the better choice in this case. Still, the difference in classification is small and both models performed with good accuracy.

## 7. CONCLUSION AND FUTURE WORK

This paper investigates data analysis, data preprocessing and intelligent classification of environmental conditions in a laboratory. Laboratory equipment is susceptible to the influence of environmental conditions. Raspberry Pi and several sensors measuring environmental factors were used for monitoring environmental conditions in a laboratory. Firstly, the data recording was done when the conditions were optimal. Afterwards the disturbances in the form of poor environmental factors were introduced. These recordings were used to form a dataset, which was later analyzed and preprocessed in Python. When the data preparation was completed, intelligent classification of environmental conditions was conducted using  $k$ -Nearest Neighbors and Random Forest Classifier. In order to select optimal models, a grid search was used to set the number of neighbors for the KNN model, and number of estimators and minimum samples per leaf for Random Forest Classifier model. The accuracy of both KNN and Random Forest Classifier models was satisfying, but the results have shown that the KNN algorithm had a bigger problem with classifying poor conditions. This issue occurred because the majority of the collected data referred to the optimal conditions. Therefore, Random Forest Classifier proved to be a better choice in this case, even though the difference in classification accuracy was almost insignificant.

This work is only the beginning of our research. In the future research, we plan on working on a new method for detecting direct influence of environmental conditions on the performance of dynamical systems. Our objective is to propose intelligent approaches to prevent the influence of the disturbances on operation and performance of dynamic systems. In our future research, we will try to analyze and process the detected environmental disturbances which affect the work of the system, using new intelligent algorithms. The results would be used as an input in the control logic and would be employed to minimize the negative influence of environmental conditions.

**Acknowledgement:** *Funded by the European Union under project Smart Products and Services Engineering (SPaSE) 101047566-ERASMUS-JMO-2021-MODULE. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or EACEA. Neither the European Union nor the granting authority can be held responsible for them. This work has also been supported by the Ministry of Education, Science and Technological Development of the Republic of Serbia.*

#### REFERENCES

- [1] N. Driendl, F. Pauli and K. Hameyer, "Influence of Ambient Conditions on the Qualification Tests of the Interturn Insulation in Low-Voltage Electrical Machines," in *IEEE Transactions on Industrial Electronics*, vol. 69, no. 8, pp. 7807-7816, Aug. 2022, doi: 10.1109/TIE.2021.3108721.
- [2] F. A. L. Souza, P. C. T. Pereira, H. de Paula, B. J. C. Filho and A. V. Rocha, "Motor drive systems reliability: Impact of the environment conditions on the electronic component failure rates," *2014 IEEE Industry Application Society Annual Meeting*, 2014, pp. 1-8, doi: 10.1109/IAS.2014.6978463.
- [3] M. S. S. Chani, K. S. Karimov, A. M. Asiri, M. M. Rahman and T. Kamal, "Effect of Vibrations, Displacement, Pressure, Temperature and Humidity on the Resistance and Impedance of the Shockproof Resistors Based on Rubber and Jelly (NiPc-CNT-Oil) Composites," in *Gels*, vol. 8, no. 4, 226, Apr. 2022, doi: 10.3390/gels8040226.
- [4] M. Shooshtari, A. Salehi and S. Vollebregt, "Effect of Humidity on Gas Sensing Performance of Carbon Nanotube Gas Sensors Operated at Room Temperature," in *IEEE Sensors Journal*, vol. 21, no. 5, pp. 5763-5770, 1 March 1, 2021, doi: 10.1109/JSEN.2020.3038647.
- [5] R. A. Amy, G. S. Aglietti and G. Richardson, "Reliability Analysis of Electronic Equipment Subjected to Shock and Vibration – A Review," *Shock and Vibration*, vol. 16, no. 1, 2009, pp. 45-59, doi: 10.3233/SAV-2009-0453.
- [6] A. M. Veprik, "Vibration protection of critical components of electronic equipment in harsh environmental conditions," *Journal of Sound and Vibration*, vol. 259, no. 1, 2003, pp. 161-175, doi: 10.1006/jsvi.2002.5164.
- [7] R. Usamentiaga, M. A. Fernandez, A. F. Villan and J. L. Carus, "Temperature Monitoring for Electrical Substations Using Infrared Thermography: Architecture for Industrial Internet of Things," in *IEEE Transactions on Industrial Informatics*, vol. 14, no. 12, pp. 5667-5677, Dec. 2018, doi: 10.1109/TH.2018.2868452.
- [8] Z. -x. Tu, C. -c. Hong and H. Feng, "EMACS: Design and implementation of indoor environment monitoring and control system," *2017 IEEE/ACIS 16th International Conference on Computer and Information Science (ICIS)*, 2017, pp. 305-309, doi: 10.1109/ICIS.2017.7960010.
- [9] A. A. Jaber, F. K. I. Al-Mousawi and H. B. Jasem, "Internet of things based industrial environment monitoring and control: a design approach," in *International Journal of Electrical & Computer Engineering (2088-8708)*, pp. 4657-4667, vol. 9, no. 6, Dec. 2019, doi: 10.11591/ijece.v9i6.pp4657-4667.
- [10] A. D. Đorđević, M. B. Milovanović, M. T. Milojković, M. D. Spasić, "Intelligent Classification of Environmental Conditions Influencing the Work of Laboratory Equipment", *Proceedings of the XVI International Conference on Systems, Automatic Control and Measurements, SAUM 2022, Niš, Serbia, November 17.-18., 2022.*, Publisher: Faculty of Electronic Engineering, Niš, Faculty of Mechanical Engineering, Niš, Serbia (Accepted for publication)
- [11] S. Garcia, J. Luengo and F. Herrera, "Data preprocessing in data mining", vol. 72, Cham, Switzerland: Springer International Publishing, 2015.



- [12] *Raspberry Pi 3 Model B+*, [Online]. Available: <https://www.raspberrypi.com/products/raspberry-pi-3-model-b-plus/>, visited in June 2022.
- [13] D. L. Whaley III, "The interquartile range: Theory and estimation", PhD Thesis, East Tennessee State University, 2005.
- [14] H. M. Kaltenbach, "A concise guide to statistics", Springer Science & Business Media, 2011.
- [15] L. Jiang, Z. Cai, D. Wang and S. Jiang, "Survey of Improving K-Nearest-Neighbor for Classification," Fourth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD 2007), 2007, pp. 679-683, doi: 10.1109/FSKD.2007.552.
- [16] O. Kramer, "Dimensionality reduction with unsupervised nearest neighbors.", vol. 51. Berlin: Springer, 2013.
- [17] S. Sathe and C. C. Aggarwal, "Nearest Neighbor Classifiers Versus Random Forests and Support Vector Machines," 2019 IEEE International Conference on Data Mining (ICDM), 2019, pp. 1300-1305, doi: 10.1109/ICDM.2019.00164.
- [18] R. Primartha and B. A. Tama, "Anomaly detection using random forest: A performance revisited," 2017 International Conference on Data and Software Engineering (ICoDSE), 2017, pp. 1-6, doi: 10.1109/ICoDSE.2017.8285847.