Machine Learning for identifying atomic species from optical emission spectra generated by an atmospheric pressure non-thermal plasma

Octavio Rosales Martínez, Allan A. Flores Fuentes, Antonio Mercado Cabrera, Rosendo Peña Eguiluz, Everardo Efrén Granda Gutiérrez, and Juan Fernando García Mejía

Abstract: An automatic recognition method of nine atomic species through ensemble classifiers based on decision trees from experimental data collected by optical emission spectroscopy (OES) is presented. Experimental spectra were obtained from OES of an atmospheric pressure non-thermal plasma (APNTP) generated in parallel circular plates dielectric barrier discharge reactor (DBDR). APNTP's emission was detected and acquired by a monochromator coupled to a photomultiplier and a data acquisition system. Data were organized in columns as relative intensity versus wavelength to generate a synthetic spectra dataset. The performance categorization of candidate classifiers was assessed using the F1 metric; after that, the grid-search hyperparameter optimization technique allowed the selection of the best combination to construct the final ensemble classifier. After the generation of the synthetic spectra dataset, they were evaluated using parametric statistics with analysis of variance (ANOVA) and non-parametric statistics with Friedman's tests. Subsequently, the critical distance was obtained by Nemenyi parametric profile, showing the best-classified groups with prediction accuracy of the species between 93% and 100% and a confidence value of 95% in the wavelength range from 200 to 890 nm. Finally, the automatic atomic species recognition test was carried out utilizing a set of nine files, each one

Octavio Rosales Martínez, Allan A. Flores Fuentes (correspondence), Rosendo Peña Eguiluz, Everardo Efrén Granda Gutiérrez, and Juan Fernando García Mejía
Centro Universitario Atlacomulco, Universidad Autónoma del Estado de México, Atlacomulco 50450, México.
e-mail: tavo_rmx@hotmail.com (O.R.M.); aafloresf@uaemex.mx (A.A.F.F.); rosendo.eguiluz@inin.gob.mx (R.P.E.); eegrandag@uaemex.mx (E.E.G.G.); fgarciam@uaemex.mx (J.F.G.M.)

Antonio Mercado Cabrera, Rosendo Peña Eguiluz
Plasma Physics Laboratory, Instituto Nacional de Investigaciones Nucleares, Carretera México-Toluca s/n, La Marquesa, 52750 Ocoyoacac, México.
e-mail: antonio.mercado@inin.gob.mx (A.M.C.); rosendo.eguiluz@inin.gob.mx (R.P.E.)

corresponding to an experimental spectrum obtained from an APNTP generated in three different argon-oxygen gas mixtures, where Ar I, O I, and O II species with predictions range from 73% to 100% (86.5 % mean). Further, the proposed method could be trained to analyze various species generated by some other type of electric discharge.

Keyword Machine learning; parametric and non-parametric statistics; optical emission spectroscopy; atmospheric pressure non-thermal plasma

1. Introduction

Atmospheric pressure non-thermal plasma (APNTP), also known as cold atmospheric plasma (CAP), is a weakly ionized gas resulting from the supply of a high alternating electric field between two electrodes separated by at least one layer of dielectric material and a gap [1, 2, 3]. This kind of plasma exhibits an interesting differential species temperature behavior, where the average temperature of electrons (T_e) is higher than that of heavy species (T_g). It has been noted that T_g is comprised in the temperature range from 20°C to 150°C [4, 5].

Applications of APNTP in environmental technology have an essential role in the abatement of recalcitrant pollutants, promoting their degradation by way of different chemical reactions in gases and liquids [6, 7, 8, 9]. Also, APNTP contributes as an emerging field in medical applications [10, 11], which has led to relevant innovations in the area, for instance: wound healing [12, 13], materials surface sterilization [14], blood coagulation [15], cancer treatment [16], skin treatment [17], biological medicine [18], food [19], among others [20, 21, 22].

Optical emission spectroscopy (OES) is a technique that detects the light emitted from different sources like stars, natural and laboratory-made plasmas, flames, electric discharges, etc. OES is a non-invasive method that provides the identity of excited chemical species of the source in function of the optical emission spectrum's wavelength; meanwhile, the spectrum intensity is proportional to the number of excited atoms or molecules of the present element. OES obtained from electromagnetic radiation comprehended in the wavelength range of 200 to 890 nm is a rich data source that exhibits photon emissions at specific wavelengths from different species of atomic elements and molecules [23, 24]. Through this procedure, the species contained in the resulting spectrum can be identified. In addition, based on several spectrum characteristics and applying diverse physic-chemical models, it is possible to calculate other parameters of the analyzed light source, such as electronic, vibrational, rotational temperature, and density of chemical species [25, 26, 27, 28].

OES is utilized for characterizing plasma discharges allowing us to know their typical parameters; for instance, the excitation temperature using the different spectral lines of a plasma can be estimated by the Boltzmann equation. In some

studies, the database of species and elements of the National Institute of Standards and Technology (NIST) is consulted [29].

In the literature, some artificial intelligence algorithms with back propagation artificial neural networks were implemented to identify 28 gamma ray isotopes to perform the automatic characterization of species. By using artificial neural networks (ANN) configured with 47 input neurons, 52 hidden layers, and 26 output neurons (where each one corresponds to a radioisotope), the ANN was trained with 409 input data sets and a total number of iterations of 500,000. As a result, isotope species identification achieves from 0.21 to 0.99 (due to external noise measurements) after carrying out 500,000 total iterations [30].

In this regard, the use of Machine Learning algorithms, such as decision trees, is proposed for the characterization of species with a local repository of species data reported by NIST, and a graphical user interface, to reduce and perform local queries for research. Thus, considering the importance of spectroscopy as a scientific tool for the study of plasmas, in this work, we propose to implement an automatic recognition method to predict the lines of nine atomic species (five at energy level I and four at energy level II) from experimental data collected by optical emission spectroscopy using ensemble-classifiers based on decision trees.

This system estimates the electronic temperature at the line of an element species by user intervention choosing three points: first is a maximum equivalent to the observed wavelength, and second two minimums as a base. After that, the area under the curve is computed by the trapezoid method, and a filter is used to smooth the curve. Finally, the electronic temperature is interpolated at least twice to reduce the estimation error [31].

1.1 Motivation

The need to improve the analysis not only in the species lines identification but also to calculate excitation temperature (among others) in spectra obtained from non-thermal plasmas reduces tasks and time for the user. This goal can be achieved by implementing proper machine learning (ML) techniques and a correct methodology to design and test the algorithm. A recurring problem in the application of ML techniques is unbalanced data. A study shows that decision trees with this type of data cause poor performance and proposes using Hellinger's distance as a corrective action, giving rise to nodes with higher-purity leaves. Hellinger's distance is used to quantify the similarity between two probability distributions [32]. Another problem in decision trees is overfitting; to solve it, there are pruning techniques. Besides, some tests based on the *Weka* software and decision trees show that pre-pruning limits the tree construction parameters, such as depth and number of observations. In post-pruning, the tree is designed first; later, the structure of the generated tree is analyzed, and the pruning proceeds [33]. These procedures aim to generalize the classification structure by reducing excessive complexity and then using it for new data and obtaining better predictions.

The choice of programming language is essential. *Python* is a multi-paradigm programming language; its main features are fast enough, flexible, expressive syntax, open source, cross-platform, portable, extensible, dynamically typed, modular, and general purpose. It also has general-purpose libraries with a scientific computing-oriented approach, such as *NumPy, SciPy, SymPy, Pandas, Matplotlib*, and *Scikit-learn*, which can interact with each other to generate powerful and high-quality applications. Another tool is *Jupyter Lab*; it is a work environment for programming languages such as *Julia*, *Python*, and *R*, among others. Its main features are the execution of code in cells, its own markup language, and interactivity. Moreover, its main advantage is the agile development of prototypes. Some drawback is the impossibility of using *Jupyter's ipywidgets* to create executable files as end-user applications. However, free software alternatives such as *Tkinter, WxPython*, and *PyQt,* among others, are used instead [34, 35, 36, 37, 38].

Taking expert considerations into account, the model generated with ML as an automatic identification tool performs; a) data pre-processing, b) line balancing of the species considered, c) estimation of the amount of data used in training, d) hyperparameter search, e) determination of the step size, width, and temperature of the generated synthetic spectra, f) optical shift correction of the experimental spectrum, and g) acquisition parameters of the spectrometer used. Moreover, as mentioned in previous paragraphs, Hellinger's technique (the last one suitable for unbalanced data) is used to improve the classifiers based on decision trees, but in our case, data balancing was performed with random oversampling at issue species. Because the choice species requires the expertise and experience of the user, a graphical interface is additionally created; it should be noted that the design of this interface is not presented in this work. In addition, a method for automatically estimating electronic excitation temperature is presented with all the parameters reported in the NIST online database.

Finally, the authors present through this work a contribution of supervised learning algorithms, based on criteria from previous studies, providing tools for the analysis of plasma discharges like the non-thermal plasma via optical emission spectroscopy technique, leaving as background information the use of ML in this area of knowledge.

2. Automatic recognition with Machine Learning

The automatic characterization of element species in spectra from cold plasma requires a sequence of steps detailed in methodology section 3. Whereas in Section 2.1, the theoretical concepts of optical emission spectroscopy and plasma temperature estimation equations are described. It is relevant to mention that automatic characterization refers to the fact that once an experimental spectrum has been loaded, optical shift correction, continuous background correction, peak detection, and species prediction are executed without user intervention.

2.1 Optical Emission Spectroscopy

Many techniques and instrumentation are currently available for the analysis and diagnosis of APNTP. A wide review of advances in atomic spectrometry and related techniques is presented in State of the Art by E. Hywel Evans *et al.* (2020), where about 194 references show developments in atomic spectrometry published from 2020 to 2021 [39]. The authors present an update on atomic spectrometry, a review of trends in atomic spectrometry, and related techniques. From this, it can be concluded the importance of plasma diagnosis through one or several techniques, as well as the requirement of specific software and plasma analysis, either atomic or molecular [40, 41, 42].

Optical emission spectroscopy is a non-invasive technique; therefore, it does not modify the characteristics of the plasma source for the species characterization of excited or ionized gasses [27]. For instance, this technique estimates the excitation temperature in the different spectral lines of argon and neon gas-produced plasma [28]. A typical way of identifying species manually by the user is by capturing light radiation emitted by a plasma source through optic fiber, a spectrometer, computer equipment, and software. This generates a representative spectral data file to identify the species and calculate its electronic, vibrational, rotational temperature, and density. In a case study, these parameters are calculated through the Boltzmann equation in plasma sources with mixed gasses of acetylene and air. Then, from a spectral line, the *SigmaPlot* software calculates the linear regression; this technique verifies the correspondence between the temperature found and the species under study [22].

Considering an approach of equilibrium state, the Boltzmann equation (1) can be used to estimate the temperature of a plasma from its experimental spectrum. Mainly, a method to estimate this temperature is by identifying the characteristic emission lines for a specific atomic element in the optical emission spectrum based on Boltzmann's plot method [31].

$$ln\left(\frac{\boldsymbol{I_{ki}}\cdot\boldsymbol{\lambda_{ki}}}{A_{ki}\cdot g_k}\right) = -\frac{E_k}{\boldsymbol{k_B}\cdot T} \qquad (1)$$

where subindex k is the upper energy level (integer), subindex i is the lower energy level (integer), I_{ki} is the emission intensity from the k energy level to the i energy level (a.u.), λ_{ki} is the corresponding wavelength (nm), A_{ki} is the transition probability function (s^{-1}), g_k is the statistical weight (a.u.), E_k is the electronic energy for the considered upper energy level (eV), k_B is the Boltzmann constant (8.6173×10^{-5} eV/K) and, T is the temperature (K).

Spectra from OES are commonly obtained as data blocks recorded as relative intensity versus wavelength. This information can be processed by some software tools which the user can manipulate and interpret data. Table 1 summarizes different software, either licensed or free, capable of studying, analyzing, and

characterizing matter via spectroscopy techniques. In this regard, the need for software, whether free or licensed, plays an important role in plasma diagnosis [22, 26, 27, 43, 44, 45, 46, 47, 48].

This is largely due to the increase in programming languages such as *Julia*, *R*, and *Python*, the latter being one of the most widely used to perform scientific algorithms largely with *SciPy 1.0*, as documented in the contributions up to the year 2020 by Virtanen *et al.* (2020).

Table 1 Different software to study, analyze, and characterize matter based on spectroscopy techniques.

Reference	Focus on application	Technique implemented	Software name	Type of license
Welz et al. (2015) [22]	Cold atmospheric plasma analysis	Non mentioned	SigmaPlot version 12.0	Licensed
Indrajit et al. (2011) [26]	Study for determination of heavy metals in fish species	Optical emission spectroscopy	21 CFR 11 version 4.1.0	Licensed
Kolpaková et al. (2011) [27]	Displaying measured emission spectra and identification of spectrum lines in glow discharge	Optical emission spectroscopy	Spectrum Analyzer	Licensed
Abbasi et al. (2015) [43]	Atmospheric-pressure plasma jet	Optical emission spectroscopy	Specair	Free
Gajdošík et al. (2021) [44]	Measuring glutamate and glutamine (Glx) and gamma-aminobutyric acid (GABA)	Magnetic resonance spectroscopy	Inspector	Free
McManus et al. (2017) [46]	Authentication of materials	Laser-induced breakdown spectroscopy	Quantagenetics®	Licensed
Navrátil et al. (2006) [47]	Spectra of various RF discharges in pure neon	Optical emission spectroscopy	Non mentioned	Free

Oeltzschner et al. (2020) [48]	Non mentioned	Magnetic resonance spectroscopy	Osprey	Free

As mentioned above, the estimation of plasma temperature is an important diagnostic parameter. Estimating the temperature of the species requires the knowledge and expertise of the user. To calculate this, the intensity and wavelength characteristics of each line of the species are required and then compared with the data reported in NIST. For this, it is necessary to consider a partial local thermodynamic equilibrium where the plasma temperature could be estimated from the Boltzmann plot [31]. The latter is based on a sage selection of groups of emission lines sufficiently spaced in energy values. In such a way that the NIST data corresponding to the species identified by the lines of the optical emission spectrum are downloaded to be entered into the next expression derived from Eq. (1).

$$y = ln\left(\frac{\boldsymbol{I_{ki}\cdot\lambda_{ki}}}{A_{ki}\cdot g_k}\right) \tag{2}$$

Thus,

$$m = \frac{1}{\boldsymbol{k_B}\cdot T} \tag{3}$$

Then plotting Eq. (2) with a linear regression against the energy corresponding to each considered wavelength, a straight line with slope *m* is generated so that the temperature *T* is obtained from Eq. (3). The result of this procedure is depicted in the plot of Fig. 1.

It should be noted that the lines selected to estimate the electronic excitation temperature must verify the next conditions:

1. Belonging to the same element.
2. Emission line groups are sufficiently spaced in energy values.

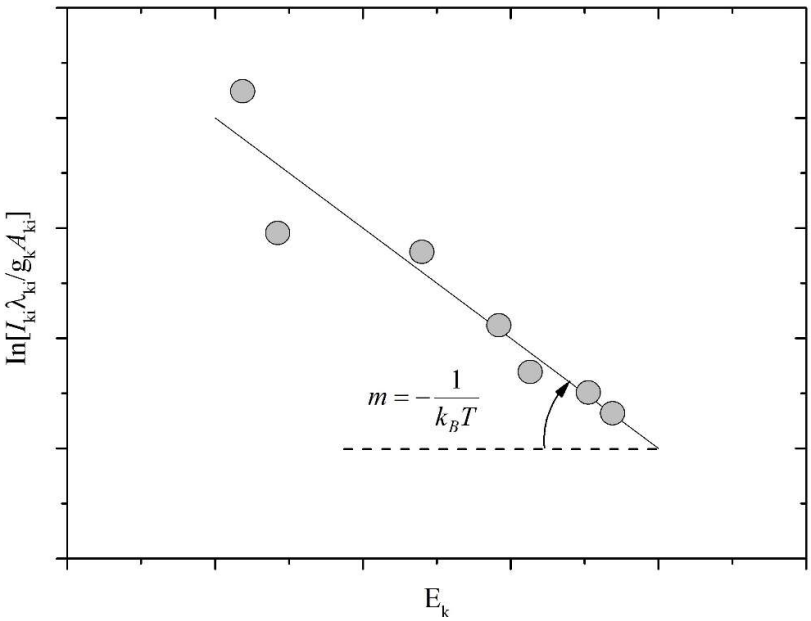


Fig. 1 Experimental electronic excitation temperature.

2.2 Characterization techniques based on Machine Learning

Machine Learning is an area of knowledge of Artificial Intelligence (AI), which through algorithmic and statistical inference, can find the pattern of a dataset. For this process, training data is used as input, for instance, in an algorithm. Then, a fitting model is created for the data, and finally, the output of the model is compared with the test data to evaluate its performance. Currently, the use of ML in pattern recognition, specifically in the application of spectroscopy, has increased because it allows the study and analysis to be more accurate and reduces expert error.

However, sometimes ML techniques are implemented with supervised knowledge that can be provided by decision trees, allowing automatic recognition by classifying datasets with previous data treatment. For instance, studies concerning spectrum analysis with different fields of application using algorithms are based on decision trees. In the study presented by O. Miettinen (2018), the classification of stellar objects with eight algorithms shows that the highest prediction percentages are achieved using the technique named Random Forest with 81 % and Gradient Boosting with 82 %. Results are obtained with 80 % data training and the 20 % remaining for testing with the use of 10-fold cross-validation on

unbalanced data whose range of proportion in the class to be predicted ranges from 1 % to 37.95 %. It is worth noting the influence of the reduced number of features to minimize overfitting and increase prediction accuracy [49]. Other ensemble systems like random forest and coarse Gaussian support vector machine classifiers were implemented for automatic galaxy classification using 4.4 million archives of spectroscopy study, achieving a prediction detection of 0.992 [50].

Other studies implement a neural network (NN) with forward propagation and error backpropagation training to a dataset of 22 spectra from organic samples (18 for training and 4 for testing) and 37 aerosol spectra (29 for training and 8 for testing) for automatic recognition, where one NN is used for each chemical element of interest [30].

A consistent problem in ML is the unbalanced data set; some studies showed that the bias toward the majority class could be reduced with decision tree classifiers (DTC), using algorithms insensitive to class size and generating statistically significant rules. Some authors propose measurement techniques such as CCP (new measure, class confidence proportion) and branch pruning. Through these two fundamental changes, a classifier that performs statistically better on classifiers can be assembled using typical decision tree techniques [32]. A SMOTE drawback is that an oversampling creates data samples that do not exist, like interpolation, which is not an appropriate technique for this type of problem because it would generate synthetic values unrelated to the data reported at NIST. Methods based on DTC are the boosting algorithms like gradient boosting classifier, AdaBoost classifier, and extreme gradient boosting classifier. These usually get better predictions than those typically implemented only with DTC because they improve internal error correction. Conversely, their computational consumption cost in training times is considerably higher than other algorithms [51, 52]. So, recently algorithms based on supervised and unsupervised learning have been widely used in scientific computing [53] and have been implemented in experimental spectroscopy for diagnosis and analysis [54, 55].

In Table 2, the state-of-the-art Machine Learning techniques applied to optical emission spectroscopy for plasma discharge applications are included. It can be established from the diversity of proposed methods, developed databases, and obtained results that: Machine Learning techniques are applied through either supervised or unsupervised algorithms. The datasets on which they carry out the identification of atomic species or molecules are obtained from the optical emission spectroscopy technique. The experimental results obtained for the validation of the different works are performance measures, critical indicators, and accuracy. The contributions mentioned range from complete systems such as software and some type of platform to own methodologies to predict chemical species in OES and other parameters of plasma discharges. In this regard, it can be established that the proposed method is comparable with related works since the identification of nine atomic species is presented from experimental OES of data acquired from an experimentally generated non-thermal plasma.

In addition, some observations in works cited in Table 2 are highlighted below:

1. The t-SNE method is used in unsupervised learning with clustering techniques [56].

2. The PCA method is applied to join and correlate project features. Also, the authors only used the expected wavelength and energy level, so PCA is not entirely appropriate [57, 58].
3. Some authors require large datasets for training, although running many experiments is expensive [59, 60].
4. The CNN are implemented for image identification and classification; the spectra are treated as maps [59, 61].

Finally, in relation to our proposal, the accuracy of the species prediction is between 93% and 100%, associated with a confidence value of 95% for the wavelength range from 200 to 890 nm (which is larger than that of one cited in Table 2), and for the nine species considered using synthetic data (972 spectra) the average accuracy for all classes at 95% confidence intervals is reached, the minimum accuracy with a mean of 0.93905 for Hg I at = 0.028 nm, and the maximum for N I, N II, and O I, at mean accuracy of 1.0 for at least one step size (Fig. 14). Furthermore, the proposed method could be trained to analyze other kinds of species generated by any other type of electric discharge utilizing data from NIST.

Table 2 State-of-the-art contributions of implemented Machine Learning techniques in optical emission spectroscopy for cold plasma applications.

Related work reference	Main characteristics	Methods	Instances or databases	Results	Contributions
Chen, H.F. *et al*. (2023) [56]	A novel manifold learning and tree-based ensemble classifier to predict residual stress of an aluminum nitride thin-film.	An unsupervised learning uniform manifold approximation and projection and t-distributed stochastic neighborhood embedding (t-SNE) were implemented. After dimensionality reduction, a tree-based ensemble model is trained and evaluated.	Data are available in the supplementary material of the proposal.	The results of performance measurement were true positive (TP) 34.32%, false positive (FP) 0.21%, and false negative (FN) 0.71%. Moreover, the critical indicators values obtained are recall 0.9797, precision	A data-driven recognizing system for AlN thin film residual stress by using manifold learning to pre-process OES data.

				0.9939, and F1-score 0.9867.	
Carter, J. A. *et al.* (2021) [57]	Evaluation of matrix effects caused by carbon and easily ionizable elements by machine learning tools in inductively coupled plasma optical emission spectrometry (OES)	Principal component analysis by supervised and unsupervised machine learning models training data via five repetitions of 10-fold cross-validation.	A dataset of 75/25 train/test data split.	The best predictive results of accuracy and R2 found were 0.970 and 0.856, respectively.	A potential ML predictor-based software capable of alerting users in real-time.
Rabasovic, M.S. *et al.* (2022) [58]	Machine Learning algorithms to classify optical emission spectroscopy of plasmas at different electron temperatures and excitation energies.	Principal Component Algorithms to reduce the dimensionality of the problem and clustering algorithms for the plasma electron temperature estimation.	Data are available in the supplementary material of this work.	It is reported classification of optical spectra of plasmas at different electron temperatures obtained with diverse excitation energies based on several clustering algorithms is reported.	Training computer software to recognize the spectra provided by a laser-triggered electric discharge spark at different electron temperatures by clustering algorithms.
Wang, C.Y. *et al.*	Real-time detection of volatile organic	VOC monitoring and detection is based on	A total of 64.000 spectra in the	CNN model is trained to classify	An alert platform sends an instant

(2021) [59]	compounds (VOC) detection and classification using optical emission spectroscopy (OES) of plasma by a convolutional neural network (CNN).	OES previously acquired for 5 seconds. Later the CNN is trained by extracting the characteristics from spectra to determine the critical spectra activating an alert at the user platform.	wavelength range of 300-685 nm were acquired.	volatile organic compounds with an accuracy of 99.9%.	message via email when a volatile organic compound is detected. The impact of AI by CNN techniques is feasible in plasma OES analysis for VOC classification.
Zhu, J. *et al*. (2022) [60]	A synergistic methodology of artificial intelligence (AI)-augmented ion mobility and mid-infrared spectroscopy (IMMS).	First, the calibration of data was made by the standard transform, SMOTE, and ALS. Afterward, the t-SNE was used for feature extraction. Finally, Deep Learning techniques such as LDA and DNN were used to assist gas classification and prediction.	A dataset was obtained at different gas concentrations: 1300, 800, 400, and 215 ppm.	A 99.08 % accuracy for a precise gas mixture concentration prediction.	A synergistic methodology of artificial intelligence (AI) augmented by ion mobility and mid-infrared spectroscopy (IMMS) for isopropyl alcohol (IPA) concentration prediction.
Li, L.N. *et al*. (2021) [61]	It is presented a discussion of the different ANN	Artificial Neural Network (ANN) methods such as	Multiple datasets available in the literature	A review of ANN schemes for multifarious	Prospects for the development of ANN-LIBS

	techniques is presented to carry out qualitative and quantitative analysis of the chemometrics in laser-induced breakdown spectroscopy (LIBS).	Functional Neural Networks (RBFNN), Convolutional Neural Networks (CNN), and Self-Organizing Maps (SOM), among others, were applied to classify, identify, and recognize patterns in laser-induced breakdown spectroscopy.	were discussed.	chemometrics applied to LIBS analysis in the past decades.	methodologies covering joint detection of the information from generalized spectra. In addition, the improvement and better combination of different ANNs for quantitative and qualitative analysis.
Lin, L. *et al*. (2021) [62]	Optimization of plasma medicine by optical emission spectroscopy with ANN.	Using ANNs in series to optimize the plasma chemical composition in real-time. 700 OES examples were used for the ANN training and 200 spectra for the testing.	A dataset of 900 OES was collected from the experimental measurements of a helium-guided cold atmospheric plasma jet.	Optimization of different combinations of N_2, O_2, H_2O, and He by means of ANN. The prediction of spatial resolution for; a) helium–air ratio, b) the mean electron temperature, and c) reactive species versus	A real-time diagnostic of target situations in vivo. The optimization focuses on self-adaptive plasma chemistry. A proposal to achieve intelligent plasma therapy.

				H_2O admixture.	
Kim, D.H *et al*. (2021) [63]	Fault detection and classification for advanced equipment control using information from optical emission spectroscopy (OES) produced by plasma glow discharge.	After data pre-processing, the abnormalities are detected in real-time through the isolation forest algorithm. Afterward, the Adaboost algorithm identifies the root cause. Finally, the DeepSHAP algorithm predicts the main parameters as gas flow rate and critical plasma information.	A dataset is obtained using the information provided by the sensors and plasma information from in situ OES data.	Increased identification accuracy of 99%, instead about 50% obtained with the conventional IPA recognition.	The best model recognition provides an accuracy of 93.6%. In addition, the performance comparison among two trained models denotes that the model defined with OES input data, plasma information (PI), and state variable identification (SVID) is the best option.

3. Method

The proposed workflow design shown in Fig. 2 is composed of a) the National Institute of Standards and Technology (NIST) online database and experimental dataset and comprises b) simulation of the synthetic spectra data frame, c) machine learning data frame, d) data correction data frame, e) graphical user interface (GUI) data frame, and f) Windows application for the user. This workflow scheme was implemented in *Jupyter Notebook* (an open-source web application) and QT5 multi-platform framework running in a GUI on Windows operating system.

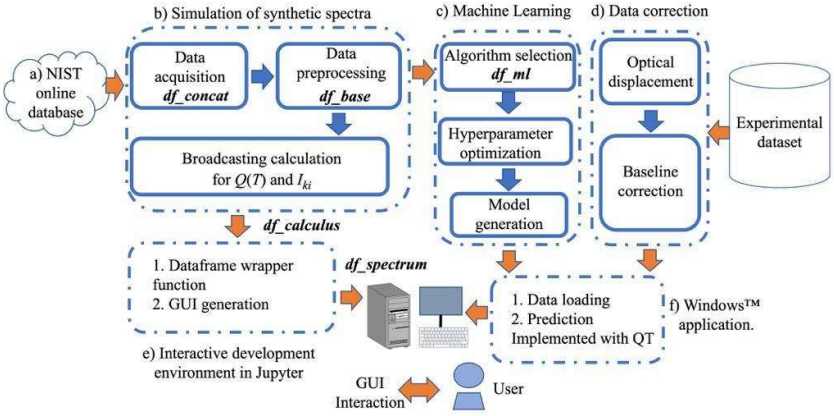


Fig. 2 Workflow design of the proposed automatic atomic species prediction.

The simulation of synthetic spectra data frame requires information from the NIST online database that can be downloaded from the web cloud (https://physics.nist.gov/PhysRefData/ASD/lines_form.html). Obtained data were processed to create a local repository with about 118 atomic elements of the periodic table by using *Wget* of *Python*. An implemented algorithm generated a utilizable URL including parameters as the spectrum for each atomic element and subspecies, transition probability of each line, statistical weight, energy level information, wavelength data, wavelength units, and CSV text output format.

```
https://physics.nist.gov/cgi-bin/ASD/lines1.pl?spectra=Ar&limits_type=0&low
_w=&upp_w=&unit=1&submit=Retrieve+Data&de=0&format=2&line_out=1&en_unit=1&o
utput=0&page_size=15&show_obs_wl=1&order_out=0&max_low_enrg=&show_av=2&max_
upp_enrg=&tsb_value=0&min_str=&A_out=0&intens_out=on&max_str=&allowed_out=1
&forbid_out=1&min_accur=&min_intens=&conf_out=on&term_out=on&enrg_out=on&J_
out=on&g_out=on
```

Fig. 3 URL for download replacement of Argon (Ar) specie.

Thus, a pseudocode defined as Algorithm 1 was implemented to get this information through a character string from the NIST. It is important to mention that Algorithm 1 must be used with a substitution string in *Python*, as shown in Fig. 3.

Algorithm 1 Pseudocode for downloading element species from the NIST.

Input: element_list, directory_name, url,
Output: $element_i$.csv
To $element_i$ in element_list **do**:
$file\_name_i$ ←**concatenate** ($element_i$, 'csv')
url_i ← **replace** (url, $element_i$)
download_save (url_i, **concatenate** (directory_name + $file\_name_i$))
end

Synthetic data spectra were simulated in *Jupyter Notebook* and served to build a local dataset enabling automatic spectrum lines recognition [33, 34, 38, 64]. The

simulation of 972 synthetic spectra with the transition probability function of each line, statistical weight, energy level information, wavelength data, wavelength units, and the temperature was achieved in time execution of about 9.85 ms, at a standard deviation of 0.18 ms. Some samples of the mentioned simulation can be found at https://github.com/orosalesm/synthetic.

This process is complemented by Algorithm 2, called ***df_concat,*** conforming to the class ***df_base*** that is ready to calculate the model of the synthetic spectra.

Algorithm 2 Pseudocode to concatenate elements.

Input: file_list_csv, file_interest_columns
Output: concatenated_elements.csv
df ← create_dataframe ()
to $file_i$ in file_list_csv **do**:
$file\_temp_i$ ← **download** ($file_i$)
$file\_temp_i$ ← **filter** ($file\_temp_i$, file_interest_columns)
df ← **concatenate** (df, $file\_temp_i$)
end
save (df, concatenated_elements.csv)

The class ***df_base*** contains the parameters used for calculating the relative intensity. At the same time, the class ***df_calculus*** determines the index and range of NIST values. The temperature-dependent partition function can be calculated using [65]:

$$Q(T) = \sum_{k=0}^{n} \quad g_k e^{\frac{-E_k}{k_b T}} \tag{4}$$

Where n is the last populated level (a.u.). While the intensity of the chosen line is determined by [66, 67]:

$$I = \frac{2\left(\frac{g_k A_{ki}}{Q\lambda_c} e^{\frac{-E_k}{k_b T}}\right)}{\pi} \cdot \frac{w}{4(\lambda_S - \lambda_c)^2 + w^2} \tag{5}$$

Where, λ_S is the wavelength between each pair of adjacent points (nm), with a defined step increment in nm, λ_C is the NIST's wavelength in the selected range (nm), and w is the total width of the line (nm).

The resulting class ***df_spectrum*** is constantly updated based on the ***df_base*** and the ***df_calculus***. The interactive GUI implemented with the *ipgwidgets* library of *Jupyter Notebook* [64] allows the user to select the wavelength range to be displayed, the atomic specie to be studied, the energy level, the medium of the

discharge (atmospheric pressure or vacuum) and to adjust the intensity level to scale the required sections of the spectrum. Fig. 4 shows an experimental spectrum of 200 to 890 nm obtained from an Ocean Optics™ HG-1 calibration light source type Ar I and Hg I mixture at 15,000 K. It was reconstructed from 33 data files provided by a monochromator Acton SP2500. The importance of the broadcasting technique from the *NumPy* library of *Python* consists of performing matrixial mathematical operations, reducing the calculation time because some elements have more than one specie. Usually, the required execution time to generate a spectrum does not exceed one second.

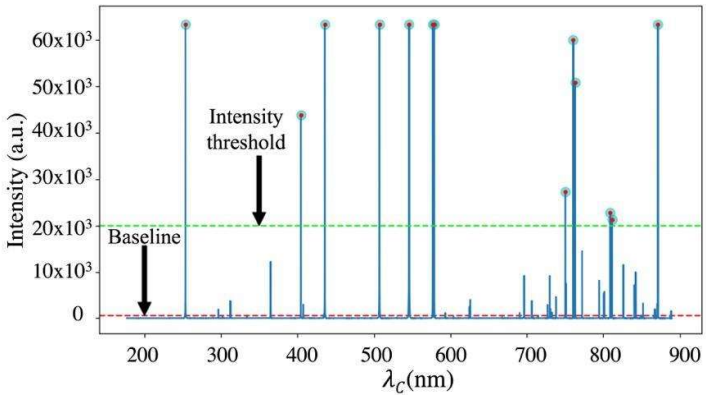


Fig. 4 HG-1 calibration light source experimental spectrum plotted in *Jupyter Notebook*.

As shown in Fig. 4, the intensity threshold value is set by the user. The automatic line detection function establishes the wavelength and line intensity of any peak exceeding this limit. It should be noted that the detection function is essential for specie detection but not for temperature calculation. Line detection is performed by modifying the data obtained with the class named *signal* of the module *SciPy* [64]. This modification consists of detecting the wavelength value of the extremes of the overall width of each line. With these extreme values, we calculate the center of each line and consider it as the FWHM; the result is observed in Fig. 5.

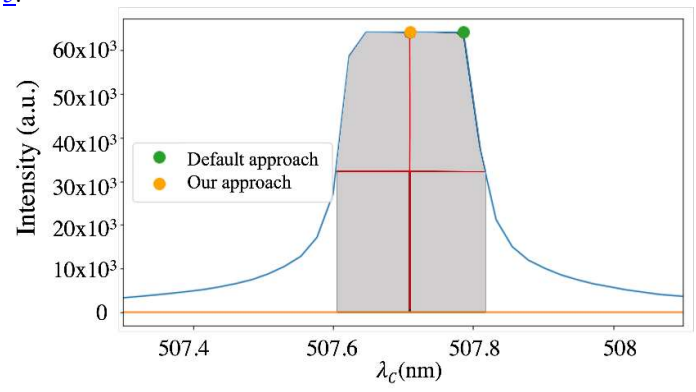


Fig. 5 Proposed approach to obtain the FWHM.

There are two issues to be solved with the acquired spectrum data: a) the baseline definition and b) the optical wavelength displacement correction. In many licensed software suites, the baseline correction procedure is implemented as a semi-manual task, where the user indicates (on a computer screen) the start and the end of the line regions while the software constructs a (piecewise linear) connecting the plot as an estimation of the baseline. Later, the obtained curve is subtracted from the original data. In this work, the procedure was carried out with the algorithm described in [68, 69, 70].

While the optical displacement is corrected using the following process: A reference data array containing the wavelength information from the atomic species of the calibration source should be established. Each experimental data array is constituted by 33,792 pairs of values matching a wavelength value with an intensity obtained with an Ocean Optics™ HG-1 calibration light source. It is necessary to identify the wavelength of the lines from the experimental spectrum and save them in another data array which is concatenated with the reference array to calculate the wavelength difference among both spectra.

Finally, a regression technique (either linear or polynomial) can be selected to obtain the corrected spectrum, which consists of the following steps:

1. Wavelengths with their respective species are obtained from the Ocean Optics™ HG-1 calibration lamp data sheet and saved in the array.
2. Plot the calibration lamp data and identify the wavelengths at which the peaks are found and saved into an array.
3. Steps 1 and 2 are repeated for each experimental run and concatenated.
4. Once all wavelengths are obtained from the Ocean Optics™ HG-1 calibration lamp experimental runs, their difference is calculated and stored in separate arrays.
5. Each coefficient of linear regression is calculated for each experimental execution.
6. Then, the experimental execution coefficients of a second-degree polynomial regression are calculated.
7. Finally, with the coefficients, the prediction functions defined by equations (6) and (7) are implemented to observe their behavior concerning optical displacement.

$$fl(x\_exp_i) = \beta 1 + \beta 2 \cdot x\_exp_i \tag{6}$$

$$fp(x\_exp_i) = \beta 1 + \beta 2 \cdot x\_exp_i + \beta 3 \cdot x\_exp_i{}^2 \tag{7}$$

Where, $fl(x\_exp_i)$ and $fp(x\_exp_i)$ are the wavelength linear and polynomial corrected vectors, respectively $\beta 1$, $\beta 2$, and $\beta 3$, are the linear and polynomial coefficients, while x_exp_i, and $x\_exp_i{}^2$ are the wavelength vectors from detected lines in the HG-1 experimental calibration lamp.

The percentage of the variance between the actual value and the predicted value is estimated by means of the determination coefficients (R^2) as:

$$R^2 = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \underline{y})^2} \quad (8)$$

Where, y_i and $\hat{y}_i$ are the actual and predicted value, and $\underline{y}$ is the median. Table 3 shows the values of R^2.

Table 3 R^2 values for each experimental run of the HG-1 calibration lamp.

Regression	R^2	α	$\beta 1$	$\beta 2$	$\beta 3$
Linear	0.791903	0.570907	0.000333	-	-
Polynomial	0.805007	-0.48287	0	-0.000022	0.0000003115681

Coefficients in Table 3 denote that the linear regression has a lower R^2 value than the polynomial regression. However, the difference is small; therefore, either linear or polynomial is implemented for the optical displacement correction. Fig. 6 shows the results obtained before and after the correction procedure for both the optical wavelength displacement and the continuous background spectrum baseline of the experimental curve.

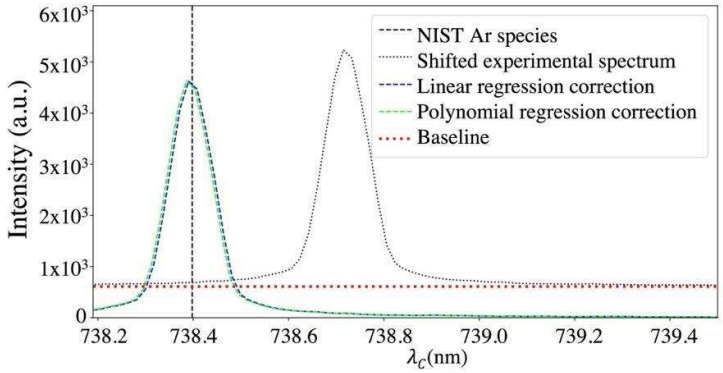


Fig. 6 Baseline and optical displacement corrections in the HG-1 calibration light source experimental spectrum generated in QT5.

To develop the interactive GUI, the *ipywidgets* library of *Jupyter Notebook* functions is required [64]. The controls are defined based on the user's needs. The options chosen for the GUI are included in Table 4; it should be noted that these can be configured initially or modified later.

Table 4 *Jupyter Notebook* initially selected controls.

Control	Initial value	Mini-mum	Maxi-mum	St ep	Description
FloatSlider	0.2	0	2	0. 1	'Wide:'
IntSlider	5000	0	50000	50 0	'Tempera-ture:'
Checkbox	False	-	-	-	'Normal-ized:'
Dropdown	'Ne'	-	-	-	'Specie:'
RadioButtos	'Air'	-	-	-	'Kind'
SelectMulti-ple	[1]	-	-	-	'Energy level:'
IntRang-eSlider	[200, 890]	0	2000	-	'Range:'

Other functional requirements of the GUI are:

1. Import spectra in CSV format.
2. Automatic characterization of the species of the elements He, N, O, Ar, and Hg.
3. Baseline and optical shift correction.
4. Estimation of the electronic excitation temperature.
5. Exportation of characterization with labels in PNG (Portable Network Graphics) and CSV format.

The minimum hardware requirements are:

1. Processor: Pentium Core i5 4570.
2. Memory: 8GB.
3. Operating System: Windows 8□, 64 bits.

In this chapter, the characteristics and requirements of the GUI are presented, but not its design because it is not the aim of this work. However, Fig. 4 and Fig. 6 are obtained from the designed interface.

3.1 Ensemble-classifier based on Decision Trees Algorithms

The proposed APNTP analysis requires two previous stages. In the first step, the following four parameters are set: 1) the studied elements: Ar, He, Hg, N, and/or O; 2) the ionization degree 1 and/or 2; 3) atmosphere type: 0 for vacuum and 1 for air and 4) the wavelength range available from 200 until 890 nm. The second step is to balance data among the nine possible classes (Ar I/0, Ar II/1, He I/2, Hg I/3, Hg II/4, N I/5, N II/6, O I/7, and O II/8) because the majority class is O II with 326 species, and the minority class is Hg I with 35 species. The balance between

classes is done with random over-sampling. As a result, the total of species is 2,934 providing 326 species per class element.

The proposed algorithms were coded, as shown in Liu *et al.* (2010), to predict the balanced classes of the nine element species. The results showed that decision tree-based algorithms achieve a good performance.

3.1.1. Unbalance data

As mentioned above, imbalanced data is a constant problem in applications of machine learning techniques. The methodology to deal with this issue consisted, on the one hand, filtering data to manage the same content. Second, the balancing of classes. This stage focuses on loading the *final_concatenated_elements.csv* file in a *DataFrame* to perform a filter based on four parameters adjusted to the established requirements (see Fig. 2, stage (b) Simulation of synthetic spectra). The options with their filters are:

1. Species: ['Hg,' 'Ar,' 'N,' 'O', 'He']
2. Degrees of ionization: [1, 2]
3. Type: [0, 1]
4. Wavelength range: [200, 890]

Based on these criteria, it is necessary to process the data according to its class, as described below, because class imbalance affects predictions. In this regard, the data work is not balanced; of the nine classes, there is a total of 1,284 species, and O II is the majority class with 326 species, while the minority class is the element Hg I with 35 species.

This class imbalance can be treated with random undersampling and random oversampling techniques [71, 72]. Initially, there were 3,899 species, and Ar II was the majority class with 1,768 species; oversampling randomly selects samples from each minority class until reaching the same number of species from the majority class; this allows the inclusion of species with minority classes in each generated tree. Since O II has 326 species, by applying oversampling, each class reaches this value, and the result is the balance of classes, with a total sum of 2,934 species for the nine classes.

3.1.2 Hyperparameter optimization

A specific space of possibilities is required for hyperparameter optimization. There are available techniques as the principles of experimental design that are: a) randomization, b) replication, c) blocking, and d) stratification are required. For this stage, the user should define the properties of each classifier, which are: a) the number considered for each division of the decision tree (*max_features*, quantity

4), b) the type of criteria (*criterion*, quantity 3), c) the maximum number of levels (*max_depth*, quantity 26), d) the number of decision trees used (*n_estimators*, quantity 10), e) sample selection method (*boostrap*, quantity 2), f) the minimum number of samples required to divide the node (*min_samples_split*, quantity 3), and g) for each leaf of the node (*min_samples_leaf*, quantity 3). Then, the total exploration space is equal to 17,820 for Decision Tree Classifier, 27,664 in Bagging Classifier, and 3,243,240 for Random Forest Classifier, and Extremely Randomized Trees, respectively. The selection criterion of these properties was an essay from the experiment based on the no-free lunch theorem [73,74]. This set of hyperparameters was explored using the techniques *GridSearch* and *RandomGridSearch*. To ensure the reproducibility conditions of this experimental design, the utilized hardware for this processing task was constituted by a processor Intel i7 3770, 16GB of RAM Kingston HyperX 1600MHz, storage ADATA SSD 480 GB, Lenovo Mahobay motherboard. At the same time, the software was integrated with NumPy V1.18.0, Pandas V0.25.3, Matplotlib V3.1.2, SciPy V1.4.1, and Scikit Learn V0.22.1.

The variance and standard correlation were considered to estimate the times for exploration of the algorithms and the configuration of the experiment to define the best combination of parameters, as can be noted in Table 5.

Table 5 Estimated time for exploration by the algorithm and its configuration for one hundred repetitions.

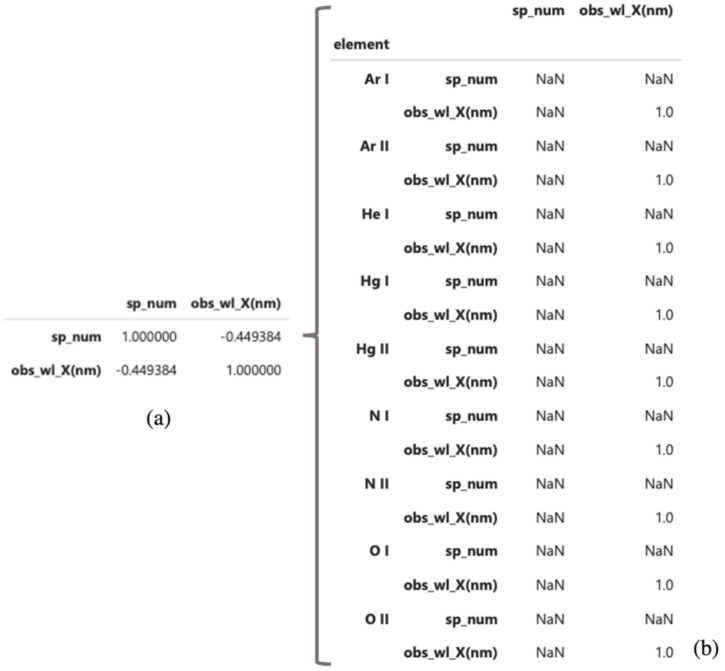
Algorithm	Hyperparameter space	Splits	Combinations	Days
DT	$17x10^3$	2	$3.563x10^6$	0.058
		3	$5.345x10^6$	0.089
		5	$8.911x10^6$	0.142
BG	$27x10^3$	2	$5531x10^6$	0.106
		3	$8.292\ x10^6$	0.124
		5	$13.830x10^6$	0.237
RF	$3.243x10^6$	2	$648.649x10^6$	10.860
		3	$972.970x10^6$	15.021
		5	$1.622x10^9$	25.419
ERT	$3.243x10^6$	2	$648.640x10^6$	9.763

3	972.971x10^9	15.667
5	1.6221x10^9	25.966

Then, based on the main advantages of the operational characteristics of a decision tree, it can be established that [51]:

1. It is a non-parametric model; this makes it consistent in the results.
2. It supports heterogeneous data (continuous, discrete, ordered, and categorical).
3. It is quick to train and predict. Its complexity is determined with $\Theta = (Nlog_2N)$ and $\Theta = (N^2)$ on average and the worst-case values, respectively.
4. It is easy to interpret plotting when the tree is short; otherwise it becomes complex.
5. It has low bias and typically high variance, meaning if the training data varies slightly, the resulting tree and predictions can change significantly.

In this case, the solution was to combine several trees in a single model. The first step was to create an ensemble of this predictor where each instance is trained with a random subset of training samples using the bootstrap aggregation method, whose advantage is that it reduces the variance of algorithms with high variance [75]. Although the models built with this technique may have structural similarities that result in correlated predictions, this was corrected by means of random forest, which consists of building several decision trees adding randomness to the construction process to give rise to a forest. As an advantage, overfitting, variance, and correlation in the predictions are reduced [51]. To improve another property of the algorithm, the extremely randomized trees algorithm was applied. As a result, the speed in the build process was increased because the division threshold is random instead of being calculated; this result is included in Table 5.



	sp_num	obs_wl_X(nm)
sp_num	1.000000	-0.449384
obs_wl_X(nm)	-0.449384	1.000000

(a)

element		sp_num	obs_wl_X(nm)
Ar I	sp_num	NaN	NaN
	obs_wl_X(nm)	NaN	1.0
Ar II	sp_num	NaN	NaN
	obs_wl_X(nm)	NaN	1.0
He I	sp_num	NaN	NaN
	obs_wl_X(nm)	NaN	1.0
Hg I	sp_num	NaN	NaN
	obs_wl_X(nm)	NaN	1.0
Hg II	sp_num	NaN	NaN
	obs_wl_X(nm)	NaN	1.0
N I	sp_num	NaN	NaN
	obs_wl_X(nm)	NaN	1.0
N II	sp_num	NaN	NaN
	obs_wl_X(nm)	NaN	1.0
O I	sp_num	NaN	NaN
	obs_wl_X(nm)	NaN	1.0
O II	sp_num	NaN	NaN
	obs_wl_X(nm)	NaN	1.0

(b)

Fig. 7 Resulting correlation values for the two different considered configurations, a) sp_num and obs_wl_X(nm), and b) obs_wl(nm) against specie.

The hyperparameter search random forest has a maximum time for exploration of 25,418 days, while extremely randomized trees require 25,967 days. As a result, the variance was reduced with the chosen bagging decision tree techniques. The second step was to determine the correlation value calculated with NIST data, *e.g.,* sp_num and obs_wl_X(nm), which for the species 'O II,' 'Hg II,' 'Ar II,' 'Ar I,' 'N I,' 'O I' showed a negative magnitude (see Fig. 7.a). This can be interpreted as follows: the longer the wavelength of the element, the lower the probability of finding a higher energy level. If now the data of obs_wl(nm) against the element is obtained, the behavior of Fig. 7.b was observed. Where the value 1.0 indicates a strong and positive correlation with itself because of the data variability. It should be noted that NaN, in *Python,* means a data type used to represent any value that is undefined or unpresentable. The reason the data was analyzed in this manner is that when it is necessary to predict an experimental spectrum, the data contains only the wavelength and, depending on the energy applied to the reactor, it is possible to deduce the energy level of I or II, or both. This feature is user configurable.

Another analysis result is provided in Table 6, where all possible energy levels were configured for each specie. The results obtained were interesting by themselves, however in our case, it does not apply, since for any experimental spectrum, an intensity is obtained for a wavelength and the data [Element, Aki(s^-1), Ek(eV), g_k, Type] are unknown *a priori*.

Table 6 Correlation values as configured for each specie and NIST parameters to Ar, He, Hg, N, and O.

specie	parameters	sp_num	obs_wl_X(nm)	Aki(s^-1)	Ek(eV)	g_k	Type
Ar	sp_num	1.0	-0.39937	0.154136	0.87129	0.129467	0.03324
	obs_wl_X(nm)	-0.39937	1.0	-0.17975	-0.55939	-0.10452	-0.08734
	Aki(s^-1)	0.154136	-0.17975	1.0	0.106096	-0.00249	-0.42428
	Ek(eV)	0.87129	-0.55939	0.106096	1.0	0.222255	0.244182
	g_k	0.129467	-0.10452	-0.00249	0.222255	1.0	0.16871
	Type	0.03324	-0.08734	-0.42428	0.244182	0.16871	1.0
He	sp_num	NaN	NaN	NaN	NaN	NaN	NaN
	obs_wl_X(nm)	NaN	1.0	-0.16582	-0.1472	0.287644	0.022618
	Aki(s^-1)	NaN	-0.16582	1.0	-0.42571	-0.08564	-0.37164
	Ek(eV)	NaN	-0.1472	-0.42571	1.0	0.176709	0.343768
	g_k	NaN	0.287644	-0.08564	0.176709	1.0	0.266761
	Type	NaN	0.022618	-0.37164	0.343768	0.266761	1.0
Hg	sp_num	1.0	-0.5334	0.112828	0.751765	0.252883	-0.24655
	obs_wl_X(nm)	-0.5334	1.0	-0.2664	-0.43908	-0.1085	0.524227

	Aki(s^-1)	0.112828	-0.2664	1.0	0.03626	0.082145	-0.46912
	Ek(eV)	0.751765	-0.43908	0.03626	1.0	0.401445	-0.18972
	g_k	0.252883	-0.1085	0.082145	0.401445	1.0	-0.08421
	Type	-0.24655	0.524227	-0.46912	-0.18972	-0.08421	1.0
N	sp_num	1.0	-0.26713	0.233188	0.842736	-0.0802	0.193697
	obs_wl_X(nm)	-0.26713	1.0	-0.26666	-0.1616	0.148963	0.713135
	Aki(s^-1)	0.233188	-0.26666	1.0	0.053472	-0.14093	-0.33605
	Ek(eV)	0.842736	-0.1616	0.053472	1.0	0.021282	0.374962
	g_k	-0.0802	0.148963	-0.14093	0.021282	1.0	0.146715
	Type	0.193697	0.713135	-0.33605	0.374962	0.146715	1.0
O	sp_num	1.0	-0.46216	0.139433	0.890054	-0.07965	-0.076
	obs_wl_X(nm)	-0.46216	1.0	-0.33163	-0.42655	0.10049	0.663108
	Aki(s^-1)	0.139433	-0.33163	1.0	0.082076	-0.05053	-0.44227
	Ek(eV)	0.890054	-0.42655	0.082076	1.0	-0.0313	-0.05859
	g_k	-0.07965	0.10049	-0.05053	-0.0313	1.0	0.04336

Type	-0.076	0.663108	-0.44227	-0.05859	0.04336	1.0

Actually, Fig. 14 in Section 5 provides better results by accuracy than correlation and variance. Thus, the correlation was not presented in a general way as an object of study for all the elements but only for its explanation. In fact, in state-of-the-art studies, research focuses mainly on the accuracy of the prediction for species or molecules in another case.

The results of the estimated time for exploration by the algorithm and its configuration are given in Table 5. The hyperparameters space was explored using a design of experiments based on repeated-stratified cross-validation (RSCV). The experiment configuration to validate the RSCV was performed with *n_splits* = 2, 3, and 5 (number of blocks for the cross-validation), *n_repetitions* = 100 (number of iterations for the repeated cross-validation), and *random_state*= 2020 (seed used to generate the random state).

The algorithms: a) Decision Tree (DT), b) Bagging (BG), c) Random Forest (RF), and d) Extremely Randomized Tree (ERT) included in Table 5 were evaluated using the F1 metric [33, 74] which validate each output by a confusion matrix (See Fig. 8). The last is based on the following four options: a) True Negative (TN), negative class and false prediction; b) False Positive (FP, error type I), negative class and true prediction, c) False Negative (FN, error type II), positive class and false prediction, and d) True Positive (TP), positive class and true prediction. This confusion matrix was implemented to verify the output of each classification algorithm. Since the classification problem presented in this work is greater than two classes, the equations derived from the confusion matrix shown in Table 7 were used [74]. From the set of equations in Table 7, the metric F1 is defined as the harmonic mean of Precision and Recall as follows:

$$F1 = 2 \times \frac{Precision \cdot Recall}{Precision + Recal} \tag{9}$$

In fact, this is a multiclass problem; the final expression is:

$$\underline{F1} = \frac{1}{c}\sum_{i=1}^{c} F1_i \tag{10}$$

Where $\underline{F1}$ is the average value, and c the total number of classes.

Table 7 Equations derived from the confusion matrix [74].

Name	Binary expression	Multiclass expression	Description

Accuracy	$= \frac{VP + VN}{N}$ $= 1 - Error$	$= \frac{VP + \sum VN}{N}$ $= 1 - Error$	Percentage of correct classifications.
Error	$= \frac{FP + FN}{N}$ $= 1 - Accuracy$	$= \frac{\sum FP + \sum FN}{N}$ $= \frac{(p' - VP) + (p - VP}{N}$ $= 1 - Accuracy$	Percentage of incorrect classifications.
VP-rate Recall Sensitivity	$= \frac{VP}{p}$	$= \frac{VP}{p}$	Percentage of the current class classified correctly. If the class is Ar I, how often does it predict Ar I?
FP-rate	$= \frac{FP}{n}$ $= 1 - Specificity$	$= \frac{\sum FP}{\sum n}$ $= \frac{p' - VP}{\sum n}$ $= 1 - Specificity$	The percentage is of FP relative to the current class. If the class is not Ar I, how often does it predict Ar I?
Precision	$= \frac{VP}{p'}$	$= \frac{VP}{p'}$	Percentage of the current class correctly predicted. If he predicts Ar I, how often is he correct?
Specificity	$= \frac{VN}{n}$ $= 1 - FPrate$	$= \frac{\sum VN}{\sum n}$ $= 1 - FPrate$	Percentage of the other classes classified correctly. If the class is not Ar I, how often do you predict it is not Ar I?

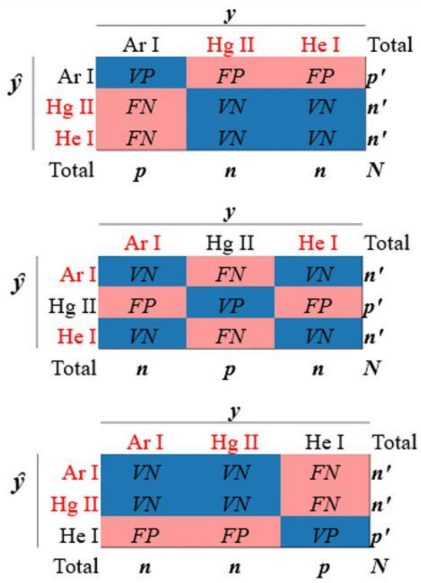


Fig. 8 Confusion matrix for classification of three classes.

3.1.3. Line detection.

Line detection is performed by choosing a data segment that exceeds an intensity threshold and determining the position of the maximum value; this process is repeated in turn, and thus the lines are chosen. This procedure is detailed using the pseudocode of Algorithm 3. As shown in Fig. 4, a clear-green dotted line is a threshold the user determines from which the peaks are detected. Also, the points in red color with blue shadow are the lines detected with Algorithm 3.

Algorithm 3. Line detection pseudocode.

```
Input: intensities, threshold, window_size
Output: line_index
intensities = intensities > threshold
beginnin ← 0
end ← window_size
i ← 0
to window_size in intensities do:
  window_size ← window_size [beginning,
end]
  line_index [i] ← arg_max (window_size)
  beginnin ← fin + 1
  end ← window_size + 1
```

```
12:         i ← i + 1
        end
        return line_index
```

4. Results

Finding the best combination of parameters for a specific algorithm is one of the most time-consuming tasks when exploring a specific possibility space. Within the cross-validation techniques, there is stratified repeated cross-validation, which is a technique that covers all the principles of the design of experiments. The configuration options used for repeatedly stratified cross-validation are included in Table 8.

Table 8 Options for repeatedly stratified cross-validation.

Parameter	Option	Description
n_splits	[2,3, 5]	The number of blocks used in cross-validation.
n_repeats	[100]	The number of times the cross-validation is repeated.
random_state	2020	Seed is used to generate the random state.

The computational cost is multiplied by the value of *n_repeats* for a possible combination of each algorithm in the hyperparameter space shown in section 3.1.2. Table 5 summarizes the combinations of each hyperparameter space with the total time achieved by the respectively applied algorithm. As a result of the hyperparameter optimization, the best combination of hyperparameters for every single classifier is provided in Table 9.

After that, the hyperparameters were assigned to their corresponding algorithm and tested again with the data training with cross-validation of 2, 3, 5, and 10 times with repeated cross-validation (RCV) and repeated-stratified cross-validation (RSCV). Fig. 9 shows that ERT is slightly better in all tests, while the other tree algorithms have similar performance. The median (vertical line within the box) and media (with the triangle) overlap, denoting an equilibrated data distribution.

Table 9 Best hyperparameters by each classifier

Hyperparameter	DT	BG	RF	ET
max_freatures	None	-	$\log_2$	None

Criterion	entropy	-	gini	entropy
max_depth	20	-	33	33
min_samples_split	2	-	2	2
min_samples_leaft	1	-	1	1
base_estimator	-	DTC	-	-
max_samples	-	0.89	-	-
n_estimators	-	100	90	100
Bootstrap	-	True	True	False
bootstrap_features	-	False	-	-

Thus, the parametric ANOVA test was performed to verify if the distribution medians in the model were the same (H0). In addition, the non-parametric Friedman test allows estimating if the medians of the distributions in the models are similar (H0), as shown in Table 10, for a significance level $\alpha = 0.05$.

The best algorithm performance depends on the cross-validation used to test, as it is depicted in Fig. 9, where the higher the cross-validation number, the higher the F1 metric prediction value. Although Friedman's p-values are correct in Table 10, it is recommendable to consider a null value. For instance, these values are used with chaotic systems such as the pendulum and double pendulum. But, in most cases, the data sets are normally distributed.

Table 10 Comparison of p-values in H0 by the Friedman and ANOVA test.

	Experiment	Friedman p-value	Friedman H0	ANOVA p-value	ANOVA H0
RCV	2x100	1.850×10^{-70}	False	0	False
	3x100	3.434×10^{-104}	False	0	False
	5x100	8.229×10^{-174}	False	0	False
	10x100	0	False	0	False
RSCV	2x100	1.350×10^{-69}	False	0	False
	3x100	5.495×10^{-104}	False	0	False
	5x100	7.328×10^{-173}	False	0	False
	10x100	0	False	0	False

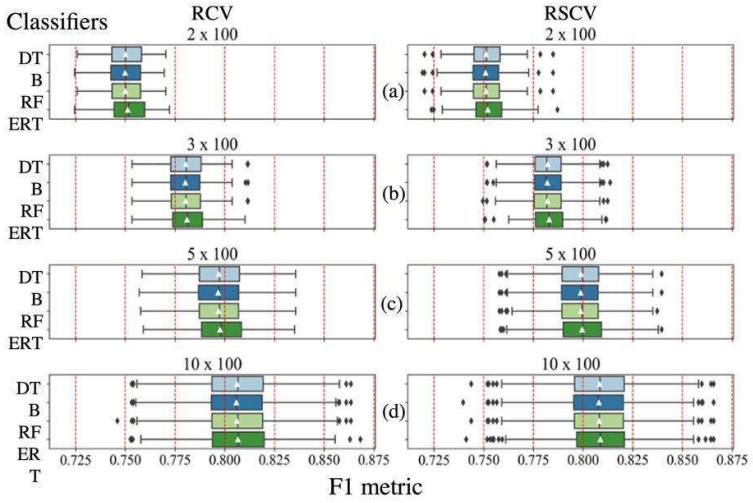


Fig. 9 Box-and-whisker plots with RCV and RSCV for the studied classifier algorithms with 2, 3, 5, and 10 times cross-validations ((a), (b), (c), and (d) respectively), each one of them repeated 100 times.

In addition, data sets validated utilizing RSCV have higher dispersion than those validated with RCV; this fact is denoted by the presence of outliers in all the RSCV studied cases. For both validation methods, when repeated ten times, wider distribution results are obtained. Thus, when they are assembled, the F1 metric prediction increases. The algorithm hyperparameters effects are constantly visualized during the training of the NIST training database and tested by cross-validation test. This characteristic defines the number of samples to train the final model and if this fits the data correctly.

Both Friedman H0 and ANOVA H0 result tests were false because one or more distributions for each validation were different. So, the algorithm assembled was accomplished with the four classifiers: DT, BG, RF, and ERT. To show subtle differences among the four algorithms, the Nemenyi test was applied with significance alpha $\alpha = 0.05$. Thus, the prediction position for each sample was obtained, and the critical distance (CD) was calculated, showing that the models were located in the range of permissible CD because the higher the number of divisions of RCV, the lower the value of CD (Fig. 10).

It does not imply that each estimator provides the same prediction error for the same observation. The automatic learning curve obtained with the assembled model was calculated by soft voting from the NIST database, and it can be observed in Fig. 11. The light blue color on the cross-validation test curve corresponds to standard deviation values; since the more data considered, the smaller the deviation obtained. As a result, the final model correctly predicts the data it was trained with.

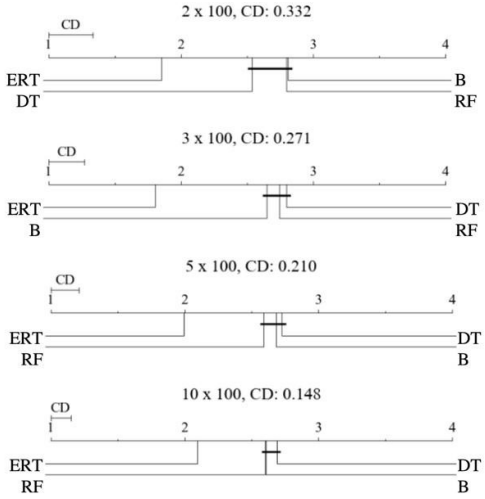


Fig. 10 Nemenyi test results for the considered classifiers.

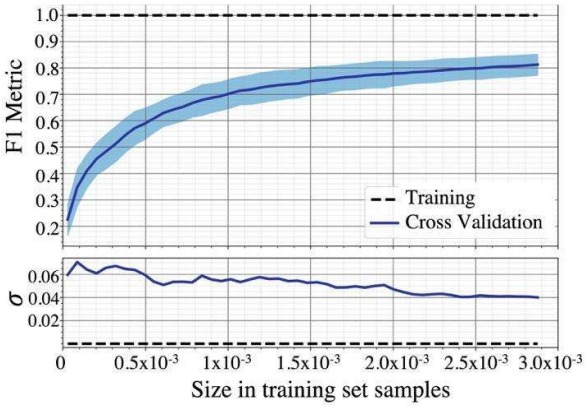


Fig. 11 Learning curve for the ensembled and trained model with the NIST data.

After creating the ensemble classifier and considering the different atomic species, several receiver operating characteristics (ROC) curves were obtained to compare the TP against the FP rate, where the closer the resulting curve is to the upper left corner, the better the classifier. Thereby, each curve can be integrated as an area under curve (AUC) value as they are provided between parentheses in Fig. 12. The species He I, Hg I, N I, and O I have a TP rate near the unit, the best-attained approach was for O I with about 0.9999, while the worst case was for O II with about 0.8257. Finally, the AUC for the metric F1 was 0.9442.

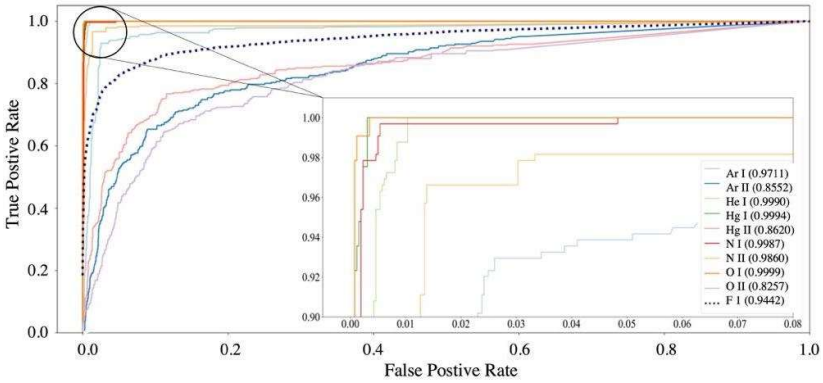


Fig. 12 ROC curves with cross-validation to voting model and a detail of the left upper corner.

5. Discussion

Table 11 provides the selected combination of attributes based on the experimental design theory. The variables were chosen to get the experimental results in the wavelength range from 200 nm to 890 nm. The goal of the selected combination is to generate the synthetic spectra dataset (972, that is, 108 spectra by specie) and evaluate the accuracy of the final ensembled model with the best hyperparameters. These synthetic spectra aim to evaluate the accuracy (accuracy shown in Table 7 is used) of the final model generated with the best hyperparameters found.

It is important to mention that the choice of confidence intervals to calculate the precision per species for each of wavelength λ_S, FWHM, and temperature T (see Table 11) was set at 0.95 using the bootstrapping technique, which is a sampling approximation that results in a Gaussian distribution, after which the mean and standard deviation can be obtained with the chosen confidence interval. Thus, accuracy was privileged over precision because the former applies to balanced data and the latter to unbalanced data.

Table 11 Selected combination of attributes for synthetic spectra.

Attribute	Options	Quantity
Species	Ar I, Ar II, He I, Hg I, Hg II, N I, N II, O I, O II	9
λ_S (nm)	0.01; 0.02; 0.03; 0.05; 0.1; 0.2	6
FWHM (nm)	0.01; 0.03; 0.05; 0.1; 0.3; 0.5	6
T(K)	1,000 K; 10,000 K; 20,000 K;	3

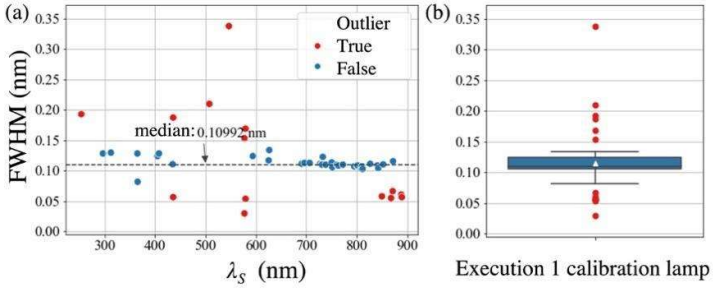
The effect of the selected combination in the synthetic spectra is resumed as follows:

1. As the value of λ_S increases, displacement of lines is induced either to the right or left direction, as well as deformation and superposition of lines.
2. The increase of FWHM causes the horizontal lines to broaden, gradually adding to adjacent ones or overlapping each other and distorting the spectrum.
3. The increase of T promotes the appearance of certain lines that are proportional to the energy present in the spectrum.

As a result, the effect in predictions is established as follows:

1. The accuracy observed for energy levels I and II is greater than 0.9 for all cases of FWHM and T in the case of O II; this property is compromised for temperatures lesser than 20,000 K. Low-intensity lines cannot always be detected.
2. This procedure provides accuracy predictions ranging from 0.8 to 1.0 for all the cases.
3. It is observed that the lower standard deviation values are at $\lambda_S = 0.012$ nm.

Fig. 13 (a), (c), and (e) show the scatter plots where the red dots represent the outliers for FWHM, and the blue ones, the non-outliers, while the dotted line represents the median value. It is observed in Fig. 13 (b), (d), and (f), the box and whisker plots that serve to verify whether the detection of outliers is correct, the white triangle represents the median value, the vertical line inside the blue box represents the median; the red and blue colors have equivalent meaning to scatter charts of Fig. 13 (a), (c), and (e). Fig. 13 (c) shows that the lowest median FWHM value (0.10973 nm) was obtained from the execution of three calibration source experiments. The results of accuracy predictions grouped by λ_S in all classes are shown in Fig. 14. Each black point represents the mean value, the vertical length of the colored lines shows the standard deviation, while the internal black lines are confidence intervals at 0.95 defined by bootstrapping and calculating the limits with the 0.025 and 0.975 percentiles respectively.



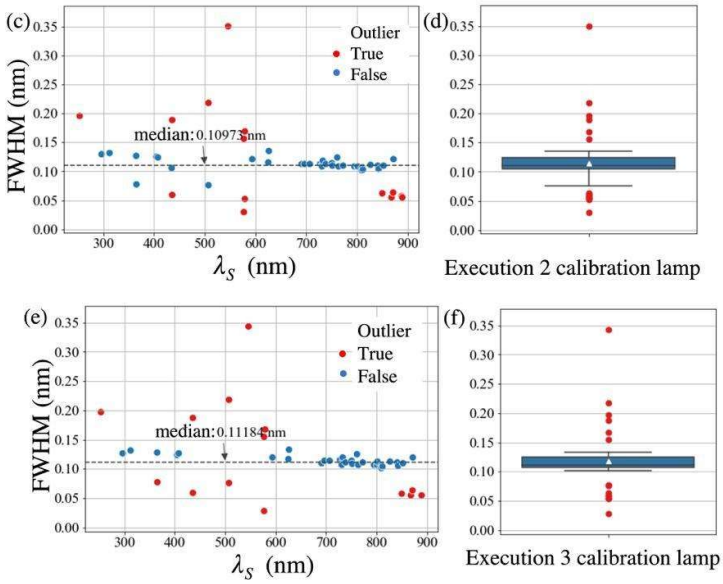


Fig. 13 Detection of outliers for the FWHM for each experimental spectrum of the HG-1, (a), (b) to 0.10992, (c), (d) to 0.10973, and (e), (f) to 0.11184.

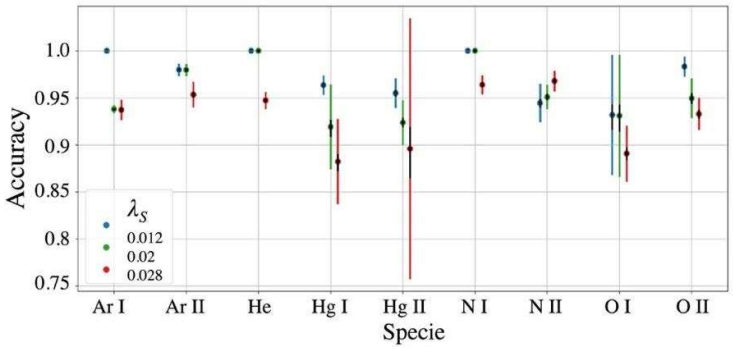


Fig. 14 Accuracy of predictions grouped by λ_S for each specie.

The confidence intervals of each species are lower than their respective standard deviation. He I, N I, N II, and O I provide an accuracy of 1.0 in at least one step size, while Hg I has the highest standard deviation at 0.028 nm. This stands that some values deviate from their mean value (0.93905) along the confidence interval, indicating that 0.95 of the predictions have an accuracy between 0.93273 and 0.94423. Table 12 provides the statistics generated from the prediction accuracy with the synthetic spectra for the conditions of λ_S and FWHM.

Table 12 Statistics of λ_S and FWHM obtained with synthetic spectra dataset.

Species	λ_S (nm)	Quantity	Median	σ	Confidence	Confidence	Confidence interval

					interval boot-strapping 95% high	interval bootstrap-ping 95% low	bootstrap-ping 95% difference
Ar I	0.012	99	0.99160	0.00285	0.99216	0.99104	0.00112
	0.02	99	0.98292	0.00452	0.98381	0.98202	0.00182
	0.028	99	0.99140	0.00238	0.99188	0.99092	0.00096
Ar II	0.012	99	0.99501	0.00243	0.99546	0.99451	0.00094
	0.02	99	0.99218	0.00245	0.99269	0.99172	0.00092
	0.028	99	0.98034	0.00439	0.98126	0.97957	0.00171
He I	0.012	99	0.97371	0.00781	0.97519	0.97228	0.00289
	0.02	99	0.99970	0.00282	1	0.99912	0.00087
	0.028	99	0.97371	0.00781	0.97519	0.97228	0.00289
Hg I	0.012	100	0.97972	0.01366	0.98233	0.97704	0.00528
	0.02	100	0.9801	0.01379	0.98292	0.97734	0.00557
	0.028	100	0.93901	0.02971	0.94422	0.93272	0.01145
Hg II	0.012	100	0.99472	0.01255	0.99643	0.99181	0.00462
	0.02	100	0.97902	0.02323	0.98305	0.97401	0.00904
	0.028	100	0.97606	0.02622	0.98070	0.97066	0.01009
N I	0.012	100	1	0	1	1	0
	0.02	100	1	0	1	1	0
	0.028	100	0.97932	0.00482	0.98028	0.97841	0.00183
N II	0.012	100	1	0	1	1	0

	0.02	100	0.99010	0.00865	0.99180	0.98835	0.00344
	0.028	100	0.99006	0.00866	0.99183	0.98832	0.00346
O I	0.012	100	1	0	1	1	0
	0.02	100	1	0	1	1	0
	0.028	100	1	0	1	1	0
O II	0.012	72	0.99672	0.00412	0.99757	0.99582	0.00174
	0.02	72	0.99162	0.00811	0.99332	0.98978	0.00354
	0.028	72	0.98997	0.01058	0.99213	0.98751	0.00465

Figure 15 shows the effect of increasing temperature T on attained accuracy. Predictions with an accuracy greater than 0.94 are obtained with temperatures under 5,000 K. It is also observed that the increase in temperature has a positive effect on the accuracy of predictions, which is noticeable in classes: Ar I, He I, Hg I, and Hg I.

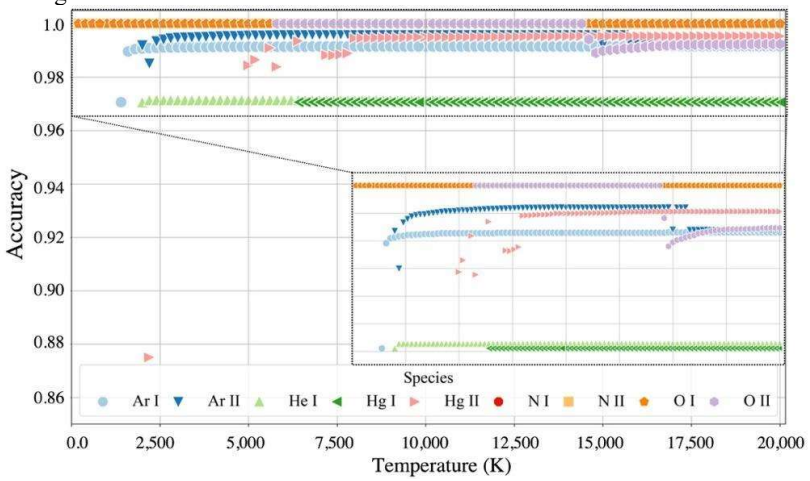
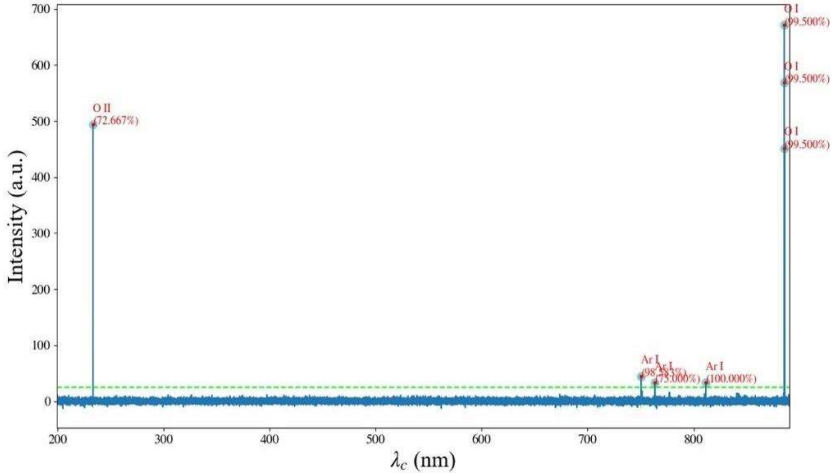


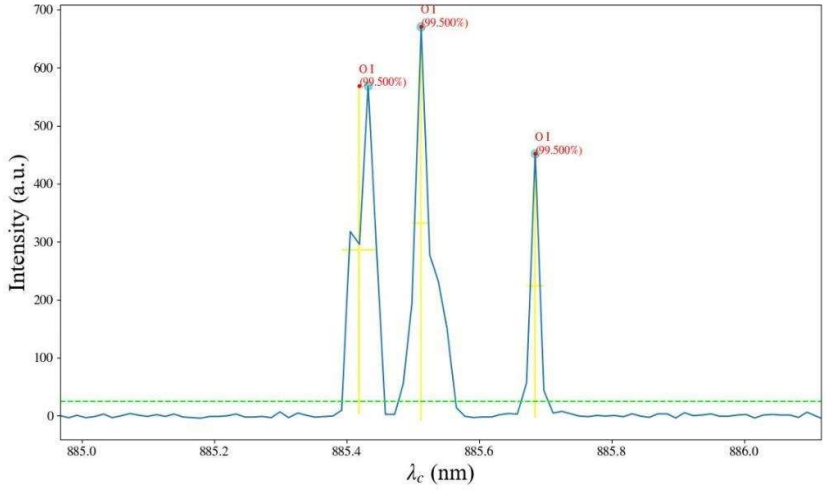
Fig. 15 Accuracy effect predictions at the increase in Temperature (K).

Finally, the results of the species identification for three different APNTP experimental spectra produced in a DBDR are provided in Fig. 16. The three graphs denote a fully uploaded experimental spectrum obtained when the appropriate settings are provided through the user GUI, which consists of a) baseline, b) area, c) wavelength range, d) line magnitude, and e) threshold detection method. The process finishes running the prediction; the GUI displays the spectrum with labels and

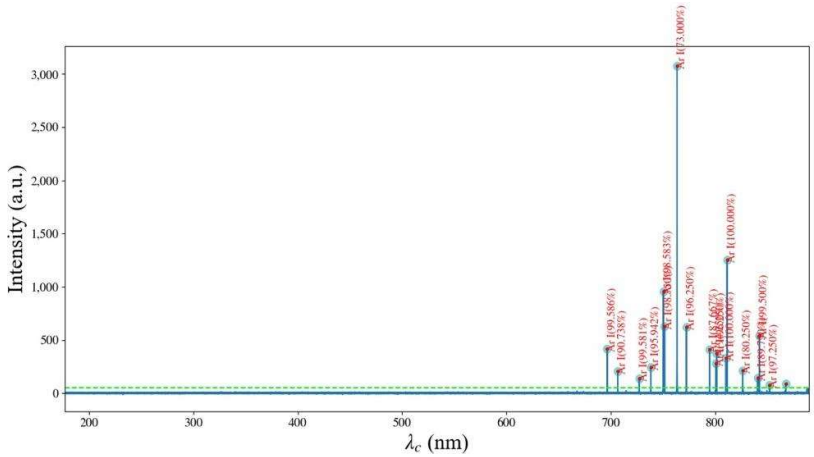
accuracy percentages. Fig. 16 (b), (d), and (f) are enlarged in a narrow wavelength range to show better detail of their previous image.



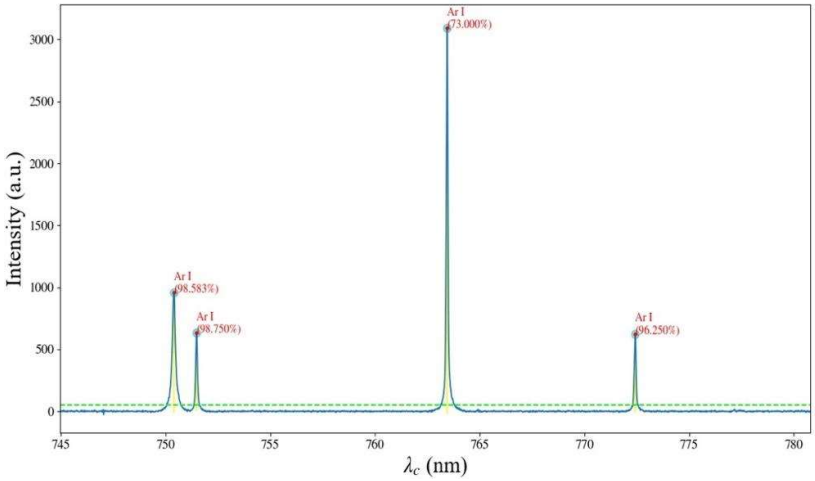
(a)



(b)


Intensity (a.u.)
λ_c (nm)
Ar I (73.000%)
Ar I (100.000%)
Ar I (98.583%)
Ar I (96.250%)
Ar I (99.586%)
Ar I (90.738%)
Ar I (99.581%)
Ar I (95.942%)
Ar I (80.250%)
Ar I (99.500%)
Ar I (97.250%)

(c)


Intensity (a.u.)
λ_c (nm)
Ar I (73.000%)
Ar I (98.583%)
Ar I (98.750%)
Ar I (96.250%)

(d)

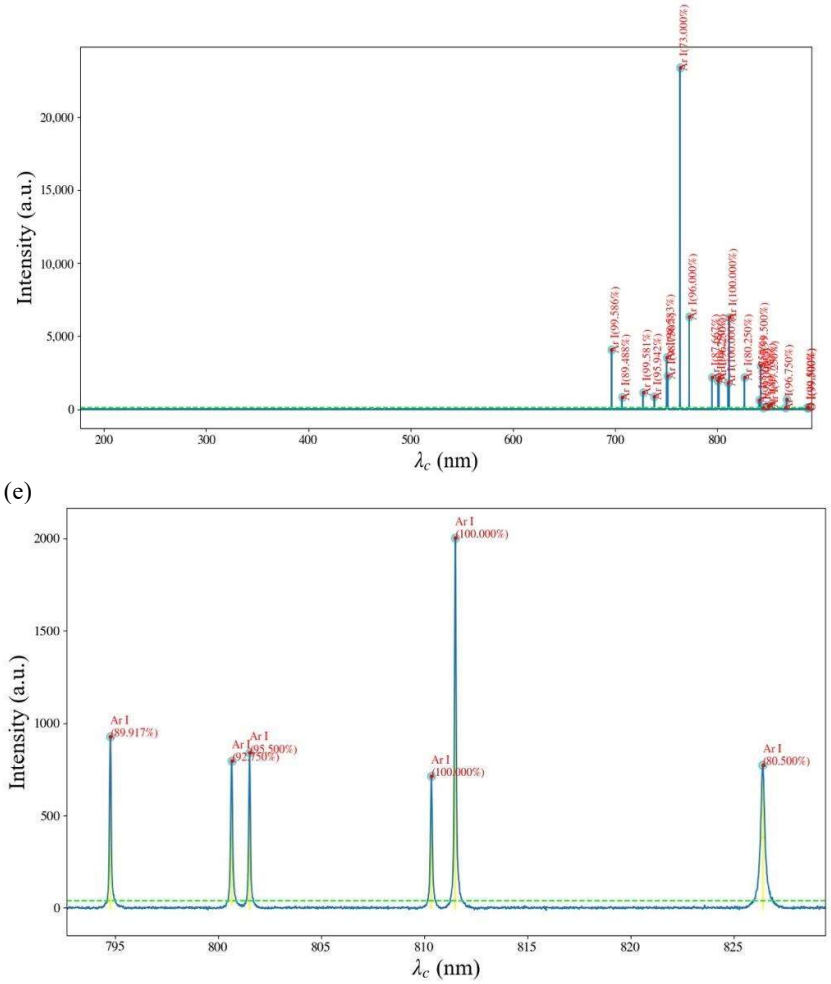


Fig. 16 Species detection produced by an APNTP at three different gas mixtures and voltage discharge parameters: (a) 10% Ar-90% O_2 at 17.5 kV, (c) 100% Ar-0% O_2 at 13.5 kV, (e) 90% Ar-10% O_2 at 13.4 kV, and their respectively zoomed area (b), (d), and (f) in a narrow wavelength range.

From Fig. 16 (a), (c), and (e), it is possible to observe Ar I, O I, and O II species with predictions ranging from 73% to 100% for the gas mixtures containing argon and oxygen. The minimum accuracy value of 73 % was due to a change introduced in the attributes established in Table 11.

7. Conclusion

A machine learning tool based on decision trees algorithms was ensembled to achieve the automatic recognition of atmospheric pressure non-thermal plasma species from optical emission spectroscopy data defined in the wavelength range from 200 nm to 890 nm. A confidence of 95% was attained, and a prediction accuracy from 0.93905 to 1.0 for all the studied species. The synthetic spectra generated from data reported by NIST were stored in a local repository to be used as training data for the decision tree-based system. The optical displacement and baseline correction made from experimental spectra results derived from the data acquired by the monochromator were suitably modified. The optimization of hyperparameters was carried out using cross-validation and the *GridSearch* technique, with searching times of approximately 26 days. Stratified repeat cross-validation and repeat cross-validation were applied to probe the optimization of the hyperparameters. Both results show similar performance, tested by Friedman and ANOVA techniques. The proposed automatic atomic species prediction was validated by modifying the parameters: λ_S, FWHM, and T, leading to results not reported in the literature for this kind of species until now. The proposed application allows: a) to correct the optical displacement and continuous background, b) to estimate the temperature of the species, and c) to integrate the ensembled model to predict the lines of the species. Thus, integrating machine learning techniques and optical spectroscopy identification in an interactive development environment provides precious information on species produced by APNTP. Finally, this methodology can be used to train models with other synthetic species and predict the species by identifying them from an optical spectrum emitted from plasma discharges based on the results obtained from synthetic data and experimental spectra with high-efficiency results.

Acknowledgements

The author disclosed receipt of the following financial support for the research, authorship, and/or publication of this chapter book. This work was partially supported by Consejo Mexiquense de Ciencia y Tecnología (CoMeCyT), México, through the program “Estancias de Investigación Especializadas COMECyT, Edo-Méx.”, grant No. EESP2021-0019.

References

1. Raizer, Y.P., Allen, J.E.: Gas discharge physics. Springer, Berlin (1997)

2. Roth R.J.: Industrial Plasma Engineering: Volume 1: Principles (1st ed.), IOP, Tennessee (1995)
3. Zhang, H., Sang, L., Wang, Z., Liu, Z., Yang, L., Cheng, Q.: Recent progress on non-thermal plasma technology for high barrier layer fabrication. Plasma Sci. Technol. 20(6), 063001 (2018). https://doi.org/10.1088/2058-6272/aaacc8
4. Nehra, V., Kumar, A., Dwivedi, H.K.: Atmospheric non-thermal plasma sources. Int. J. Eng. 2(1), 53–68 (2008)
5. Roth R.J.: Industrial Plasma Engineering: Volume 2: Applications to Nonthermal Plasma Processing (1st ed.), CRC Press, Boca Raton (2001). https://doi.org/10.1201/9781420034127
6. Baloul, Y., Aubry, O., Rabat, H., Colas, C., Maunit, B., Hong, D. Paracetamol degradation in aqueous solution by non-thermal plasma. Eur. Phys. J. Appl. Phys. 79, 30802 (2017). https://doi.org/10.1051/epjap/2017160472
7. Mercado-Cabrera, A., Jaramillo-Sierra, B., Peña-Eguiluz, R., López-Callejas, R., Valencia-Alvarado, R., Rodríguez-Méndez, B.G., Muñoz-Castro, A.E.: Chlorobenzene Degradation in Simultaneous Gas–Liquid Phases Assisted by DBD Plasma. IEEE T. Plasma Sci. 47(1), 86–94 (2019). https://doi.org/10.1109/tps.2018.2877057
8. Prysiazhnyi, V., Brablec, A., Čech, J., Stupavská, M., Černák, M.: Generation of Large-Area Highly-Nonequlibrium Plasma in Pure Hydrogen at Atmospheric Pressure. Contrib. Plasm. Phys. 54(2), 138–144 (2014). https://doi.org/10.1002/ctpp.201310060
9. Van Impe, J., Smet, C., Tiwari, B., Greiner, R., Ojha, S., Stulić, V., Vukušić, T., Režek Jambrak, A.: State of the art of nonthermal and thermal processing for inactivation of micro-organisms. J. Appl. Microbiol. 125(1), 16–35 (2018). https://doi.org/10.1111/jam.13751
10. Kuchenbecker, M., Bibinov, N., Kaemlimg, A., Wandke, D., Awakowicz, P., Viöl, W.: Characterization of DBD plasma source for biomedical applications. J. Phys D: App. Phys. 42(4), 045212 (2009). https://doi.org/10.1088/0022-3727/42/4/045212
11. Sladek, R.E.J., Stoffels, E., Walraven, R., Tielbeek, P.J.A., Koolhoven, R.A.: Plasma Treatment of Dental Cavities: A Feasibility Study. IEEE T. Plasma Sci. 32(4), 1540–1543 (2004). https://doi.org/10.1109/tps.2004.832636
12. He, R., Li, Q., Shen, W., Wang, T., Lu, H., Lu, J., Lu, F., Luo, M., Zhang, J., Gao, H., Wang, D., Xing, W., Jia, W., Liu, F.: The efficacy and safety of cold atmospheric plasma as a novel therapy for diabetic wound in vitro and in vivo. Int. Wound J. 17(3), 851–863 (2020). https://doi.org/10.1111/iwj.13341
13. Xu, G.M., Shi, X.M., Cai, J.F., Chen, S.L., Li, P., Yao, C.W., Chang, Z.S., Zhang, G.J.: Dual effects of atmospheric pressure plasma jet on skin wound healing of mice. Wound Repair Regen. 23(6), 878–884 (2015). https://doi.org/10.1111/wrr.12364
14. Heuer, K., Hoffmanns, M.A., Demir, E., Baldus, S., Volkmar, C.M., Röhle, M., Fuchs, P.C., Awakowicz, P., Suschek, C.V., Opländer, C.: The topical use of non-thermal dielectric barrier discharge (DBD): Nitric oxide related effects on human skin. Nitric Oxide.44, 52–60 (2015). https://doi.org/10.1016/j.niox.2014.11.015
15. Kisch, T., Schleusser, S., Helmke, A., Mauss, K.L., Wenzel, E.T., Hasemann, B., Mailaender, P., Kraemer, R.: The repetitive use of non-thermal dielectric barrier discharge plasma boosts cutaneous microcirculatory effects. Microvasc. Res. 106, 8–13 (2016). https://doi.org/10.1016/j.mvr.2016.02.008
16. Ishaq, M., Evans, M.M., Ostrikov, K.K.: Effect of atmospheric gas plasmas on cancer cell signaling. Int. J. Cancer. 134(7), 1517–1528 (2013). https://doi.org/10.1002/ijc.28323
17. Heinlin, J., Isbary, G., Stolz, W., Morfill, G., Landthaler, M., Shimizu, T., Steffes, B., Nosenko, T., Zimmermann, J., Karrer, S.: Plasma applications in medicine with a special focus on dermatology. J. Eur. Acad. Dermatol. Venereol. 25(1), 1–11 (2010). https://doi.org/10.1111/j.1468-3083.2010.03702.x

18. Brun, P., Pathak, S., Castagliuolo, I., Palù, G., Brun, P., Zuin, M., Cavazzana, R., Martines, E.: Helium Generated Cold Plasma Finely Regulates Activation of Human Fibroblast-Like Primary Cells. PLoS ONE. 9(8), e104397 (2014). https://doi.org/10.1371/journal.pone.0104397
19. Patriarca, M., Barlow, N., Cross, A., Hill, S., Robson, A., Taylor, A., Tyson, J.: Atomic spectrometry update: review of advances in the analysis of clinical and biological materials, foods, and beverages. J. Anal. At. Spectrom. 37(3), 410–473 (2022). https://doi.org/10.1039/d2ja90005j
20. Graves, D.B.: Oxy-nitroso shielding burst model of cold atmospheric plasma therapeutics. Clin. Plasma Med. 2(2), 38-49 (2014). https://doi.org/10.1016/j.cpme.2014.11.001
21. Tanaka, H., Ishikawa, K., Mizuno, M., Toyokuni, S., Kajiyama, H., Kikkawa, F., Metelmann, H.R., Hori, M.: State of the art in medical applications using non-thermal atmospheric pressure plasma. Rev. Mod. Plasma Phys. 1(1) (2017). https://doi.org/10.1007/s41614-017-0004-3
22. Welz, C., Emmert, S., Canis, M., Becker, S., Baumeister, P., Shimizu, T., Morfill, G.E., Harréus, U., Zimmermann, J.L.: Cold Atmospheric Plasma: A Promising Complementary Therapy for Squamous Head and Neck Cancer. PLoS ONE. 10(11), e0141827 (2015). https://doi.org/10.1371/journal.pone.0141827
23. Laux, C.O., Spence, T.G., Kruger, C.H., Zare, R.N.: Optical diagnostics of atmospheric pressure air plasmas. Plasma Sources Sci. Technol. 12(2), 125–138 (2003). https://doi.org/10.1088/0963-0252/12/2/301
24. Moon, S.Y., Choe, W.: A comparative study of rotational temperatures using diatomic OH, O2 and N2+ molecular spectra emitted from atmospheric plasmas. Spectrochim. Acta Part B At. Spectrosc. 58(2), 249–257 (2003). https://doi.org/10.1016/s0584-8547(02)00259-8
25. Argoti, A., Fan, L.T., Cruz, J., Chou, S.T.: Introducing the stochastic simulation of chemical reactions using the Gillespie algorithm and MATLAB: Revisited and augmented. Chem. Eng. Educ. 42(1), 35–46 (2008)
26. Indrajit, Sen, Ajay Shandil, and Shrivastava, V. S.: Study for Determination of Heavy Metals in Fish Species of the River Yamuna (Delhi) by Inductively Coupled Plasma-Optical Emission Spectroscopy (ICP-OES), Adv. Appl. Sc. Res. 2(2), 161-166 (2011).
27. Kolpaková A., Kudrna P., Tichý M.: Study of plasma system by OES (optical emission spectroscopy). In: Safranková J. (ed). 20th Annual Conference of Doctoral Students. Prague, Czech Republic, May 31 to June 3, 2011, pp. 180–185 (2011)
28. Watson, S., Nisol, B., Lerouge, S., Wertheimer, M.R.: Energetics of Molecular Excitation, Fragmentation, and Polymerization in a Dielectric Barrier Discharge with Argon Carrier Gas. Langmuir. 31(37), 10125–10129 (2015). https://doi.org/10.1021/acs.langmuir.5b02794
29. Hamed, S.S.: Spectroscopic Determination of Excitation Temperature and Electron Density in Premixed Laminar Flame. Egyp. J. Solids. 28(2), 349–357 (2005). https://doi.org/10.21608/ejs.2005.149334
30. Yoshida, E., Shizuma, K., Endo, S., Oka, T.: Application of neural networks for the analysis of gamma-ray spectra measured with a Ge spectrometer. Nucl. Instrum. Methods Phys. Res. A: Accel. Spectrom. Detect. Assoc. Equip. 484(1-3), 557–563 (2002). https://doi.org/10.1016/s0168-9002(01)01962-3
31. Kunze, H. J. (ed.): Introduction to Plasma Spectroscopy. Springer, Berlin (2009). https://doi.org/10.1007/978-3-642-02233-3
32. Liu, W., Chawla, S., Cieslak, D.A., Chawla, N.V.: A Robust Decision Tree Algorithm for Imbalanced Data Sets. In: Parthasarathy, S., Liu, B., Goethals, B., Pei, J., Kamat C. (eds.) Proceedings SIAM International Conference on Data Mining. pp. 767-777, Society for Industrial and Applied Mathematics, Philadelphia (2010). https://doi.org/10.1137/1.9781611972801.67

33. Barga, R., Fontama, V., Tok, W.H. (ed): Predictive Analytics with Microsoft Azure Machine Learning. Berkeley (2015) https://doi.org/10.1007/978-1-4842-1200-4
34. Kumar, R.: Future for scientific computing using Python. Int. J. Eng. Technol. Manag. Res. 2(1), 30–41 (2020). https://doi.org/10.29121/ijetmr.v2.i1.2015.28
35. Oliphant, T.E.: Python for Scientific Computing. Comput. Sci. Eng. 9(3), 10–20 (2007). https://doi.org/10.1109/mcse.2007.58
36. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blodel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Venderplas, J., Passos, A., Cournapeau, D.: Scikit-learn: Machine learning in Python. J. Mach. Learn. Res. 12, 2825-2830 (2011)
37. Wang G., Peng, B.: Script of Scripts: A pragmatic workflow system for daily computational research. PLOS Comp. Biol. 15(2): e1006843. https://doi.org/10.1371/journal.pcbi.1006843
38. Yu, W., Carrasco-Kind, M., Brunner, R.J.: Vizic: A Jupyter-based interactive visualization tool for astronomical catalogs. Astron. Comp. 20, 128–139 (2017). https://doi.org/10.1016/j.ascom.2017.06.004
39. Hywel, E.E., Pisonero, J., Clare, M.M.S., Rex, N.T.: Atomic spectrometry update: review of advances in atomic spectrometry and related techniques. J. Anal. At. Spectron. 37, 942-965 (2022). https://doi.org/10.1039/d2ja90015g
40. Jones, R.D., Stalling, D.L., Davis, J., Jurkovich, P., LaPointe, K.: Software validation for medical device manufacturing. Qual. Assur. J. 7(4), 242–247 (2003). https://doi.org/10.1002/qaj.245
41. Martinez-Urreaga, J., Mira, J., Gonzáles-Fernández, C.: Introducing the Stochastic Simulation of Chemical Reactions: Using the Gillespie Algorithm and MATLAB. Chem. Eng. Educ. 37(1), 14-19 (2003)
42. Shi, S., Finch, K., She, Y., Gamez, G.: Development of Abel's inversion method to extract radially resolved optical emission maps from spectral data cubes collected via push-broom hyperspectral imaging with sub-pixel shifting sampling. J. Anal. At. Spectrom. 35(1), 117–125 (2020). https://doi.org/10.1039/c9ja00239a
43. Abbasi, H., Nazeri, M., Mirpour, S., Farahani, N.J.: Measuring electron density, electric field and temperature of a micro-discharge air plasma jet using optical emission spectroscopy. In: Proceedings of 2nd International Conference on Knowledge-Based Engineering and Innovation (KBEI), Teheran, Iran, 5–6 November (2015) https://doi.org/10.1109/KBEI.2015.7436207
44. Gajdošík, Martin, Karl Landheer, Kelley, M. Swanberg, Christoph, J.: INSPECTOR: free software for magnetic resonance spectroscopy data inspection, processing, simulation, and analysis. Sci. Rep. 11, 2094 (2021). https://doi.org/10.1038/s41598-021-81193-9
45. García, L.A., Restrepo, E., Jiménez, H., Castillo, H.A., Ospina, R., Benavides, V., Devia, A.: Diagnostics of pulsed vacuum arc discharges by optical emission spectroscopy and electrostatic double-probe measurements. Vacuum. 81(4), 411–416 (2006). https://doi.org/10.1016/j.vacuum.2006.06.005
46. McManus, C.E., Dowe, J., McMillan, N.J.: Quantagenetics® analysis of laser-induced breakdown spectroscopic data: Rapid and accurate authentication of materials. Spectrochim. Acta Part B At. Spectrosc. 145, 79–85 (2018). https://doi.org/10.1016/j.sab.2018.04.010
47. Navrátil, Z., Trunec, D., Šmíd, R., Lazar, L.A.: Software for optical emission spectroscopy-problem formulation and application to plasma diagnostics. Czech. J. Phys. 56(Suppl 2), B944–B951 (2006). https://doi.org/10.1007/s10582-006-0308-y
48. Oeltzschner, G., Zöllner, H.J., Hui, S.C.N., Mikkelsen, M., Saleh, M.G., Tapper, S., Edden, R.A.E.: Osprey: Open-source processing, reconstruction & estimation of magnetic

resonance spectroscopy data. J. Neurosci. Methods. 343, 108827 (2020).. https://doi.org/10.1016/j.jneumeth.2020.108827
49. Miettinen, O.: Protostellar classification using supervised machine learning algorithms. Astrophys. Space Sci. 363(9), 2-15 (2018). https://doi.org/10.1007/s10509-018-3418-7
50. Bai, Y., Liu, J., Wang, S., Yang, F.: Machine Learning Applied to Star–Galaxy–QSO Classification and Stellar Effective Temperature Regression. Astron. J. 157 (1), 9 (2018). https://doi.org/10.3847/1538-3881/aaf009
51. Breiman, L.: Random Forest. Mach. Learn. 45, 5–32 (2015). https://doi.org/10.1023/A:1010933404324
52. Espinosa Zúñiga, J.J.: Aplicación de algoritmos Random Forest y XGBoost en una base de solicitudes de tarjetas de crédito. Ing. Invest. Tecnol. 21(3), 1–16 (2020). https://doi.org/10.22201/fi.25940732e.2020.21.3.022
53. Virtanen, P. et al.: SciPy 1.0: fundamental algorithms for scientific computing in Python. Nat. Methods. 17(3), 261–272 (2020). https://doi.org/10.1038/s41592-019-0686-2
54. Mesbah, A., Graves, D.B.: Machine learning for modeling, diagnostics, and control of non-equilibrium plasmas. J. Phys D: Appl. Phys. 52(30), 30LT02 (2019). https://doi.org/10.1088/1361-6463/ab1f3f
55. Meza-Ramirez, C.A., Greenop, M., Ashton, L., Rehman, I.U.: Applications of machine learning in spectroscopy. Appl. Spectrosc. Rev. 56(8-10), 733–763 (2020). https://doi.org/10.1080/05704928.2020.1859525
56. Chen, H.F., Yang, Y.P., Chen, W.L., Wang, P.J., Lai, W., Fuh, Y.K., Li, T.T.: Predicting residual stress of aluminum nitride thin-film by incorporating manifold learning and tree-based ensemble classifier. Mater. Chem. Phys. 295, 127070 (2023). https://doi.org/10.1016/j.matchemphys.2022.127070
57. Carter, J.A., O'Brien, L.M., Harville, T., Jones, B.T., Donati, G.L.: Machine learning tools to estimate the severity of matrix effects and predict analyte recovery in inductively coupled plasma optical emission spectrometry. Talanta, 223, 121665 (2021). https://doi.org/10.1016/j.talanta.2020.121665
58. Rabasovic, M.S., Marinkovic, B.P., Sevic, D.: Time resolved study of laser triggered electric discharge spark in atmosphere: Machine learning approach. Adv. Space Res. 71, 1331-1337 (2023). https://doi.org/10.1016/j.asr.2022.04.046
59. Wang, C.Y., Ko, T.S., Hsu, C.C.: Interpreting convolutional neural network for real-time volatile organic compounds detection and classification using optical emission spectroscopy of plasma. Anal. Chim. Acta. 1179, 338822 (2021). https://doi.org/10.1016/j.aca.2021.338822
60. Zhu, J., Ji, S., Ren, Z., Zhang, Z., Ni, Z., Liu, L., Zhang, Z., Song, A., Lee, C.: Artificial intelligence-augmented, triboelectric-induced ion mobility for mid-infrared gas spectroscopy (2022). https://doi.org/10.21203/rs.3.rs-1939335/v1
61. Li, L.N., Liu, X.F., Yang, F., Xu, W.M., Wang, J.Y., Shu, R.: A review of artificial neural network based chemometrics applied in laser-induced breakdown spectroscopy analysis. Spectrochim. Acta Part B At. Spectrosc. 180, 106183 (2021). https://doi.org/10.1016/j.sab.2021.106183
62. Lin, L., Yan, D., Lee, T., Keidar, M.: Self-Adaptive Plasma Chemistry and Intelligent Plasma Medicine. Adv. Intell. Syst. 4(3), 2100112 (2021). https://doi.org/10.1002/aisy.202100112
63. Kim, D.H., Hong, S.J.: Use of Plasma Information in Machine-Learning-Based Fault Detection and Classification for Advanced Equipment Control. IEEE Trans. Semicond. Manuf. 34(3), 408–419 (2021). https://doi.org/10.1109/tsm.2021.3079211
64. Randles, B.M., Pasquetto, I.V., Golshan, M.S., Borgman, C.L.: Using the Jupyter Notebook as a Tool for Open Science: An Empirical Study. In Proceedings 2017 ACM/IEEE Joint Conference on Digital Libraries (JCDL), Toronto, Canada, 19–23 June (2017). . https://doi.org/10.1109/jcdl.2017.7991618

65. De Galan, L., Smith, R., Winefordner, J.D.: The electronic partition functions of atoms and ions between 1500 °K and 7000 °K. Spectrochim. Acta B. 23(8), 521–525 (1968). https://doi.org/10.1016/0584-8547(68)80032-1
66. Flannigan, D.J.: Spreadsheet-Based Program for Simulating Atomic Emission Spectra. J. Chem. Educ. 91(10), 1736–1738 (2014). https://doi.org/10.1021/ed500479u
67. Ingle, J.D., Crouch, S.R. (eds.): Spectrochemical Analysis. Prentice Hall, Upper Saddle River (1988)
68. He S., Zhang, W., Liu, L., Huang, Y., He, J., Xie, W., Wu., P., Du, C.: Baseline correction for Raman spectra using an improved asymmetric least squares method. Anal. Methods. 6(12), 4402-4407 (2014). https://doi.org/10.1039/C4AY00068D
69. Jiang, X., Li, F., Wang, Q., Luo, J., Hao, J., Xu, M.: Baseline correction method based on improved adaptive iteratively reweighted penalized least squares for the x-ray fluorescence spectrum. Appl. Opt. 60(19), 5707 (2021). https://doi.org/10.1364/ao.425473
70. Baek, S.J., Park, A., Ahn, Y.J., Choo, J.: Baseline correction using asymmetrically reweighted penalized least squares smoothing. Analyst. 140(1), 250-7 (2015). https://doi.org/10.1039/c4an01061b
71. García, V., Sánchez, J. S., Marqués, A. I., Florencia, R., & Rivera, G. Understanding the apparent superiority of over-sampling through an analysis of local information for class-imbalanced data. Expert systems with applications, 158, 113026 (2020). https://doi.org/10.1016/j.eswa.2019.113026
72. Rivera, G., Florencia, R., García, V., Ruiz, A., & Sánchez-Solís, J. P. News classification for identifying traffic incident points in a Spanish-speaking country: A real-world case study of class imbalance learning. Applied Sciences, 10(18), 6253 (2020). https://doi.org/10.3390/app10186253
73. Adam, S.P., Alexandropoulos, S.A.N., Pardalos, P.M., Vrahatis, M.N.: No Free Lunch Theorem: A Review. In: Demetriou, I., Pardalos, P. (eds) Approximation and Optimization. Springer Optimization and Its Applications, vol 145, pp. 57-82 Springer, Cham (2019). https://doi.org/10.1007/978-3-030-12767-1_5
74. Asch, V.V.: Macro-and micro-averaged evaluation measures. [[BASIC DRAFT]]. Comp. Sci. (2013)
75. Géron, A.: Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems. Second Ed. O'Reilly Media, Sebastopol (2019).