

Article

Quantitative and Qualitative Analysis of Multicomponent Gas Using Sensor Array

Shurui Fan *, Zirui Li, Kewen Xia and Dongxia Hao

Tianjin Key Laboratory of Electronic Materials Devices, School of Electronics and Information Engineering, Hebei University of Technology, Tianjin 300401, China; 201821902031@stu.hebut.edu.cn (Z.L.); kwxia@hebut.edu.cn (K.X.); 201831903026@stu.hebut.edu.cn (D.H.)

* Correspondence: fansr@hebut.edu.cn; Tel.: +86-139-0201-3210

Received: 3 August 2019; Accepted: 9 September 2019; Published: 11 September 2019



Abstract: The gas sensor array has long been a major tool for measuring gas due to its high sensitivity, quick response, and low power consumption. This goal, however, faces a difficult challenge because of the cross-sensitivity of the gas sensor. This paper presents a novel gas mixture analysis method for gas sensor array applications. The features extracted from the raw data utilizing principal component analysis (PCA) were used to complete random forest (RF) modeling, which enabled qualitative identification. Support vector regression (SVR), optimized by the particle swarm optimization (PSO) algorithm, was used to select hyperparameters C and γ to establish the optimal regression model for the purpose of quantitative analysis. Utilizing the dataset, we evaluated the effectiveness of our approach. Compared with logistic regression (LR) and support vector machine (SVM), the average recognition rate of PCA combined with RF was the highest (97%). The fitting effect of SVR optimized by PSO for gas concentration was better than that of SVR and solved the problem of hyperparameters selection.

Keywords: gas sensor array; cross-sensitivity; PCA; random forest; particle swarm optimization

1. Introduction

Gas is everywhere in our lives. The gas exhaled by humans contains a labeled gas that can indicate certain diseases. For example, a large amount of acetone appears in the exhalation of a diabetic patient [1], a large amount of ammonia appears in the exhalation of a uremic patient [2], and surfaces produce fungi and volatilize organic compounds after food deteriorates [3,4]. The generation of gas is closely related to changes occurring in the substances around it. Since it can be used as a basis for analyzing such changes, gas detection is particularly important.

Gas sensor arrays associated with machine learning algorithms are widely used in different fields, such as the use of an electronic nose to judge the quality of food [5], predict food additives in juice [6], evaluate paraffin samples [7], classify different essential oils [8], monitor air quality using drones in real time [9–14], analyze the spatial distribution of air pollutants [15], and predict future air quality [16,17]. In addition, they can be used to determine leak sources based on the gas concentration distribution [18]. However, the gas sensor element has cross-sensitivity, which makes it impossible to use a single gas sensor to effectively detect the composition of a gas mixture.

In light of this problem, a wide variety of machine learning algorithms have been used for gas identification or gas quantification, including kernel principal component analysis (KPCA) [19], linear discriminant analysis (LDA) [8], logistic regression (LR) [20], support vector machines for classification and regression (SVM [7] and SVR [16,21]), artificial neural networks (ANN) [14], and reservoir computing [22]. It is also a good method to select the appropriate parameters from the sensor response signals for the identification and concentration estimation of mixed gas [4,23]. A summary of

gas mixture analysis methods is shown in Table 1. There are three works on qualitative identification (QAL-ID), which have been applied to gas mixtures, volatile gas of paraffin, and essential oils. There are also reports on quantitative analysis (QTY-ANLS), which were applied to gas mixtures, emissions of LNG (liquefied natural gas) bus or food additives in fruit juice, and air pollutants.

Although the above strategies can, to a certain extent, be effectively used for mixture detection and prediction of concentration, they still pose problems. KPCA requires choosing the appropriate kernel function and parameter ξ , which reduces the training efficiency. The classification model based on ANN requires a large number of training samples to achieve good training results, and is prone to overfitting and local optimum. In addition, the structure of a neural network is generally determined by an empirical method, which leads to a certain degree of gas identification accuracy decline. In many previous works, SVR has been shown to outperform other competing methods in regression tasks for gas quantification [24,25]. However, the hyperparameters of this algorithm are determined using the grid search method [6], which traverses the subspace of the specified value parameter to select the optimal value. Since the value space of a hyperparameter is not restricted, in many cases, any real value can be taken, and the choice of the subspace is not simple.

To avoid such problems, the main objective of this study was to propose a gas mixture analysis method to be applied to a gas sensor array. This proposal must include qualitative identification and quantitative analysis for gas mixtures, which would make it possible to use a gas sensor array to effectively detect the composition of a gas mixture.

Table 1. Summary of gas mixture analysis methods.

Methodology	Application
Two-dimensional wavelet transformation feature extraction + linear-SVM classifier [26]	QAL-ID for gas mixture
PCA and partial least squares (PLS) feature extraction + SVM, RF, extreme learning machine (ELM) [7]	QAL-ID for volatile gas of the paraffin
Fuzzy adaptive resonant theory map (ARTMAP) and linear discriminant analysis (LDA) [8]	QAL-ID for gas mixture of essential oils
Two-dimensional wavelet transformation feature extraction + PLS regression [26]	QTY-ANLS for gas mixture
Least-squares support vector machine-based (LSSVM-based) nonlinear regression [24]	QTY-ANLS for gas mixture
Reservoir computing [22]	QTY-ANLS for gas mixture
Gradient boosted regression tree [27]	QTY-ANLS for emissions of LNG bus
Long Short-Term Memory(LSTM) [28]	QTY-ANLS for gas in coal mine
SVM, RF, extreme learning machine (ELM), and partial least squares regression (PLSR) [6]	QTY-ANLS for food additives in the fruit juice
Genetic algorithm + SVR [21]	QTY-ANLS for gas chromatography
Neural network [12]	QTY-ANLS for air pollutants
Empirical wavelet transformation (EWT)-multi-agent evolutionary genetic algorithm (MAEGA)-nonlinear auto regressive models (NARX) [17]	QTY-ANLS for air pollutants
SVR [16]	QTY-ANLS for PM2.5
Principal component correlation analysis (PCCA) and LSTM [29]	QTY-ANLS for natural gas
Multiple regression (MR) and SVR [25]	QTY-ANLS for methane

2. Gas Mixture Analysis Method

As shown in Figure 1, the mixed gas analysis method proposed in this paper is mainly divided into two parts: qualitative identification and quantitative analysis. We used PCA combined with

random forest (RF) as a tool for qualitative identification, and the quantitative analysis adopted SVR optimized by the particle swarm optimization (PSO) algorithm (PSO + SVR).

For the qualitative identification, we chose 1/10 of a dataset to determine the number of principal components. After that, 9/40 of the dataset were extracted features utilizing PCA, which was used to build the random forest model. Then, the feature set extracted from another 9/40 was used for testing the generated model, through which we obtained the identification results. For the quantitative analysis, 9/40 of the dataset was used for training the optimization model (PSO + SVR), after which the combination of C and γ are obtained. By applying the combination of C and γ to SVR, we obtained the regression model. Finally, the quantitative analysis results for the last 9/40 of the dataset were obtained.

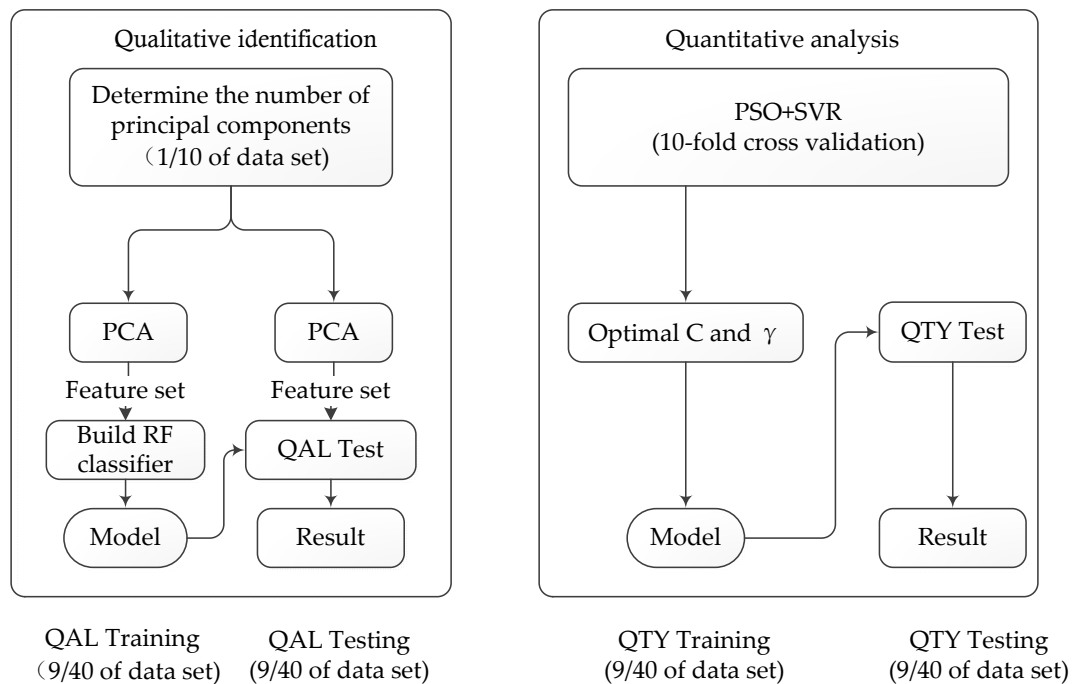


Figure 1. Gas mixture analysis method.

3. Qualitative Identification Method for Gas Mixture

3.1. Principal Component Analysis

Feature extraction is an important topic and the basis of pattern recognition and machine learning [30]. Principal component analysis is a method of feature extraction. The basic idea of it is to transform the original features into a group of new features in order of importance, from the largest to the smallest, through a set of orthogonal vectors [29]. These new features are linear combinations of the original features and they are unrelated to each other. We have provided a working process of principal component analysis.

Consider the original sample $X = [x_1, x_2, \dots, x_M] \in \mathbb{R}^{M \times N}$, where N is the number of variables, M is the number of samples, and $x_i \in \mathbb{R}^N (i \in M)$ represents the i th N -dimensional sample.

Firstly, the data of each dimension are decentralized. That is, the characteristics of each dimension are subtracted from their average values, as shown in Equation (1). $x_i^j (i \in M, j \in N)$ is the i th sample of the j th variable, and x_i^{j*} is the decentralized value of x_i^j . Secondly, the covariance matrix of X^* is calculated using Equation (2), and the eigenvalues and eigenvectors of it are obtained by eigenvalue decomposition. Then, the eigenvalues are sorted from largest to smallest as $\lambda_1, \lambda_2, \dots, \lambda_N$, and the corresponding eigenvectors are $\alpha_1, \alpha_2, \dots, \alpha_N$. Finally, the reduced number p is determined by

the cumulative contribution rate of the eigenvalue for variance r_{CCR} (Equation (3)), which utilizes $r_{CCR} \geq 99\%$:

$$x_i^{j*} = x_i^j - \frac{x_1^j + x_2^j + \dots + x_M^j}{M}; \quad (1)$$

$$C = \frac{1}{M} X^* X^{*T}; \quad (2)$$

$$r_{CCR} = \frac{\sum_{i=1}^p \lambda_i}{\sum_{j=1}^N \lambda_j} \times 100. \quad (3)$$

3.2. Random Forest

The random forest method comes from the decision tree and bagging methods. The decision tree learns a model from the given training dataset to classify new samples. The algorithm needs two sets of data: the training data used to construct the decision mechanism and the test data used to verify the constructed decision tree. The process of the decision tree learning algorithm (Algorithm 1) is presented below.

Algorithm 1. Decision Tree

Input: Training set $D = \{(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)\}$;
Attribute set $A = \{a_1, a_2, \dots, a_d\}$

Process: Function Tree Generate (D, A)

1. Generate the node
2. If all samples in D belong to the same category C , then

 3. Mark node as leaf node of class C
 4. End if
 5. If $A = \emptyset$ or the samples in D have the same value in A , then
 6. Mark node as leaf node, its category is marked as the class with the largest number of samples in D ; return
 7. End if
 8. Choose the optimal partition properties a_*
 9. For every value a_*^v of a_* , do
 10. Generate a branch for node; let D_v represent the sample subset of D evaluated at a_* to the a_*^v
 11. If D_v is empty, then
 12. Mark branch node as leaf node, its category is marked as the class with the largest number of samples in D ; return
 13. Else
 14. Take TreeGenerate($D_v, A \setminus \{a_*\}$) as branch node
 15. End if
 16. End for

Output: A decision tree with root node

On the basis of the bagging integration decision tree, the random forest further introduces random attribute selection in the training process of the decision tree.

4. Quantitative Analysis Method for Gas mixture

4.1. Support Vector Regression

Support vector regression is an important application branch of support vector machine. The basic idea is to find a regression plane to which all the data of a set are closest.

Consider training samples $D = \{(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)\}$, $y_i \in \mathbb{R}$ (m is the number of the samples), which aims to learn a regression model shaped like Equation (4), so that $f(x)$ is as close as possible to y (the absolute value), and ω and b are the model parameters to be determined:

$$f(x) = \omega^T x + b. \quad (4)$$

Suppose we can tolerate maximum deviation ε between $f(x)$ and y , namely, only when the difference between $f(x)$ and the absolute value y is larger than ε is the loss calculated. This is equivalent to making $f(x)$ the center, built with a width of 2ε intervals, and if the training samples are in this interval, the prediction will be right. We can obtain a loss function $g(n)$ with Equation (5) (N is the number of samples, y_n is the true value, and t_n is the predicted value):

$$g(n) = \frac{1}{2} \sum_{n=1}^N \{y_n - t_n\}^2 + \frac{1}{2} \|\omega\|^2. \quad (5)$$

The optimization problem can be re-expressed by introducing relaxation variables ε . For each data point x_n , the condition which makes the prediction point locate in the interval band is Equation (6), and the points above and below the interval satisfy Equation (7), where $y(x_n)$ is the true value, and ζ_n and $\hat{\zeta}_n$ are the positive and negative values of t_n beyond the interval 2ε :

$$y_n - \varepsilon \leq t_n \leq y_n + \varepsilon, \quad (6)$$

$$t_n \leq y(x_n) + \varepsilon + \zeta_n \text{ and } t_n \geq y(x_n) - \varepsilon - \hat{\zeta}_n. \quad (7)$$

The optimization problem of support vector regression can be written as Equation (8):

$$\begin{aligned} \min_{\omega, b, \zeta_n, \hat{\zeta}_n} \quad & C \sum_{n=1}^N (\zeta_n + \hat{\zeta}_n) + \frac{1}{2} \|\omega\|^2 \\ \text{s.t.} \quad & t_n \leq y(x_n) + \varepsilon + \zeta_n \\ & t_n \geq y(x_n) - \varepsilon - \hat{\zeta}_n \\ & \zeta_n \geq 0, \hat{\zeta}_n \geq 0, n = 1, \dots, N. \end{aligned} \quad (8)$$

4.2. PSO

Particle swarm optimization seeks the optimal solution through cooperation and information sharing among individuals in the group. It simulates the swarm behavior of insects, herds, birds, and fish, which search for food in a cooperative way, with each member of the group constantly changing its search patterns by learning from its own experience and that of other members. The whole process of the algorithm is as follows:

- Step 1. Initialize a group of particles with the group size n , set their original velocity and location, and set the maximum number of iterations at the same time;
- Step 2. Define the fitness function to evaluate the fitness of each particle;
- Step 3. Find the optimal solution for each particle (individual extremum), from which a global value is found, which is called the global optimal solution;
- Step 4. Update the velocity and position of the particle by Equations (9) and (10), where V_{id} and X_{id} are the d dimensional velocity and position of particle i , P_{id} and P_{gd} are the d dimensional optimal position searched by particle i and the global optimal position of the whole group, ω is the inertia factor, C_1 and C_2 are the learning factor, and $\text{random}(0, 1)$ is a random number between (0, 1):

$$V_{id} = \omega V_{id} + C_1 \text{random}(0, 1)(P_{id} - X_{id}) + C_2 \text{random}(0, 1)(P_{gd} - X_{id}), \quad (9)$$

$$X_{id} = X_{id} + V_{id}. \quad (10)$$

Step 5. The algorithm will be terminated when the number of iterations reaches the setting; otherwise, it will return to step 2 to continue execution.

4.3. SVR Optimized by the PSO Algorithm

The performance of SVR depends on the appropriate choice of hyperparameters C and γ . The penalty coefficient C reflects the degree of the algorithm's penalty on the sample data exceeding the ε pipelines, and its value affects the complexity and stability of the model. If C is too small, the penalty for the sample data exceeding ε pipelines is small and the training error becomes larger. If C is too large, the learning accuracy will be improved correspondingly, but the generalization ability of the model will be worse. γ reflects the degree of correlation between the support vectors. If it is very small, the connection between the support vectors is relatively loose, learning machines are relatively complex, and promotion ability cannot be guaranteed; on the other hand, if it is too large, the influence between support vectors will be too strong, and the regression model will have difficulty achieving sufficient accuracy.

Particle-swarm-optimized SVR was used here to select the optimal combination of C and γ , which can solve the problem of hyperparameter selection and improve the prediction accuracy. The algorithm flow of particle-swarm-optimized SVR is shown in Figure 2.

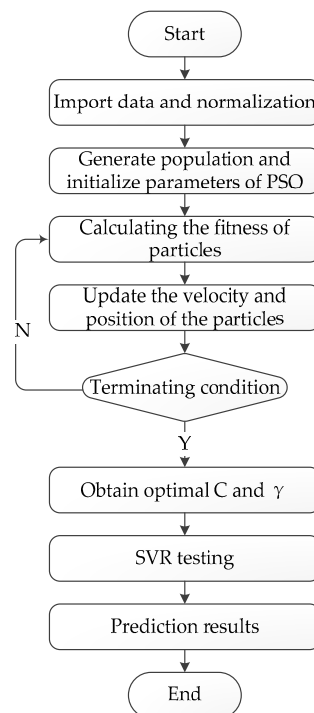


Figure 2. Particle swarm-optimized support vector regression (SVR) algorithm flow chart.

The algorithm steps were as follows:

- Step 1.* Import the original data, divide it into training data and test data, and normalize these;
- Step 2.* Initialize the parameters of PSO, including population n , particle velocity v , and position x , and iteration number;
- Step 3.* Calculate the fitness value of the particle. The current fitness value of the particle is compared with the fitness value of the historically optimal position. If it is better, it will be regarded as the current optimal position. Compared with the global optimal position fitness value of each particle, if it is better, it will be considered the current global optimal position;

- Step 4.* Update the velocity and position of the particle by Equations (9) and (10);
- Step 5.* Determine whether termination conditions are met. If they are satisfied, the optimal C value and γ value are output and assigned to SVR. Otherwise, return to step 3;
- Step 6.* Test the optimal model of SVR and obtain the prediction results.

5. Experiments and Results

5.1. Dataset

The dataset used was based on the UCI (University of California Irvine) dataset [25], which consists of the responses of methane, ethylene, air, and their mixtures in arrays of 16 sensors (TGS2600, TGS2602, TGS2610, and TGS2620; four units of each type) with a continuous measurement time of 10,486 s. The gas-sensing material of this type of gas sensor is a metal oxide which is adsorbed on the surface of the metal oxide when it is heated to a certain high temperature in the air. When a reducing gas occurs, the surface concentration of the negatively charged oxygen decreases, causing the resistance of the sensor to decrease. Some parameters of these four types of sensors are presented in Table 2.

Table 2. The characteristics of the four types of sensors.

Sensors	Sensitivity (Rate of Change for R_S)	Stability	Detection Range (ppm)
TGS2602	0.08~0.5	Long-term stability	0~10
TGS2600	0.3~0.6	Long-term stability	0~10
TGS2610	0.5~0.62	Long-term stability	500~10,000
TGS2620	0.3~0.5	Long-term stability	50~5000

In order to facilitate observation, the sensor responses and concentration values were normalized, as shown in Figure 3. The four channels from top to bottom were TGS2602, TGS2600, TGS2610, and TGS2620, as well as the concentration of two gases. As shown in Figure 3, TGS2602 responded significantly to changes in ethylene concentration, but the response curve was not very obvious. TGS2600 and TGS2620 responded to changes in methane and ethylene, TGS2610 responded significantly to changes in methane concentration, and the four sensors had slow responses to rapidly changing gases.

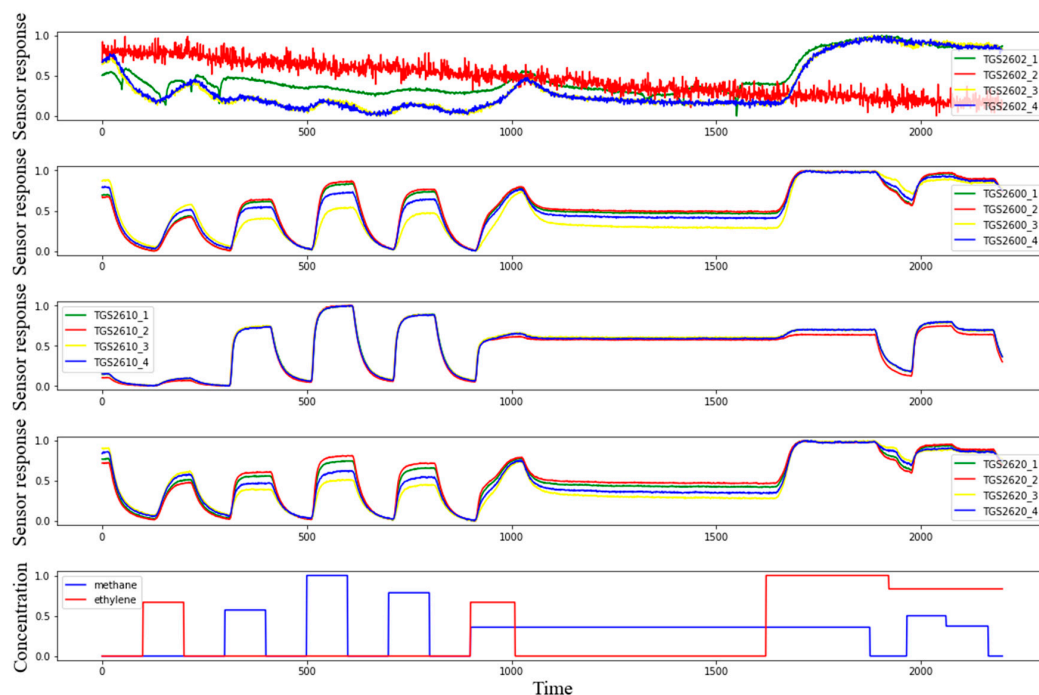


Figure 3. The sensor responses and gas concentrations.

5.2. PCA Feature Extraction

The data matrix of PCA was 10,476 rows and 16 columns. The input matrix was artificially scaled so that the mean was 0 and the variance was 1. The covariance matrix of the normalized data was calculated to obtain a matrix of 16 rows and 16 columns. The 16 eigenvalues and contribution rates are shown in Table 3. When there were four principal components, the cumulative contribution rate reached 99.69% (more than 99%), which could almost represent all the information. From the fifth principal component, the cumulative contribution rate increased with a smaller step and gradually approached zero. It can be confirmed that the dataset decreased from the original 16 dimensions to four dimensions. In the literature [7], the volatile gas characteristics of paraffin samples were also analyzed using PCA. Using the first three principal components, it can be seen that the paraffin samples were clearly divided into four grades, and the eigenvalue contribution rate of these principal components was 93.34%, but the first five principal components were finally extracted to form a new feature dataset.

Table 3. The eigenvalues and contribution rate of PCA.

Principal Components	Eigenvalues	Contribution Rate/%	Cumulative Contribution Rate/%
PC1	10.134	63.33	63.33
PC2	4.204	26.27	89.61
PC3	1.277	7.98	97.59
PC4	0.336	2.10	99.69
PC5	0.025	0.15	99.85
PC6	0.016	0.10	99.95
PC7	0.003	0.02	99.97
PC8	0.002	0.01	99.98
PC9	0.002	0.01	99.99
...
PC16	0.000	0.00	100.00

5.3. Qualitative Identification for Gas Mixture

The feature vector sets of training data were used to model the random forest, and then the feature vector sets of the test data were qualitatively identified by the model. In order to confirm the relevance of the random forest algorithm, we compared this algorithm with two other algorithms: LR and SVM. We chose these algorithms in their basic form. In our comparisons, we used the default parameters of each algorithm, as cited below:

LR: penalty: 'l2', C: '1', solver: 'lbfgs', multiclass: 'multinomial';

SVM: 'kernel: 'linear', decision_function_shape: 'ovo'.

Figure 4 shows the confusion matrices for three classifiers, which were able to separate the four classes. In [31], they also used a confusion matrix to visually compare two classifiers. The sum of all values in the matrix is the total amount of data for classification. The values on the diagonal are the correctly identified data of each category, while the values off the diagonal are the misidentified data of each category. Comparing all of the data on the diagonal lines of the three figures (Figure 4a–c), we can see that the value on the diagonal of the RF is the largest, indicating that RF had the highest probability of correctly identifying each class compared with LR and SVM. The values on the diagonal of LR and SVM were similar, indicating that the classification effect of LR and SVM was similar. From Figure 4, we can see that the RF confusion matrix had the highest values on the diagonal and the lowest values off the diagonal. We calculated the average recognition rate η for each classifier using Equation (11) (x_{ii} is the value on the diagonal, $i \in 1, 2, 3, 4$; x is the total classification data). We found that η of RF was the highest (97%), and the average recognition rate of LR and SVM was 85%, which is 12% less than RF:

$$\eta = \frac{x_{11} + x_{22} + x_{33} + x_{44}}{x} \times 100\% \quad (11)$$

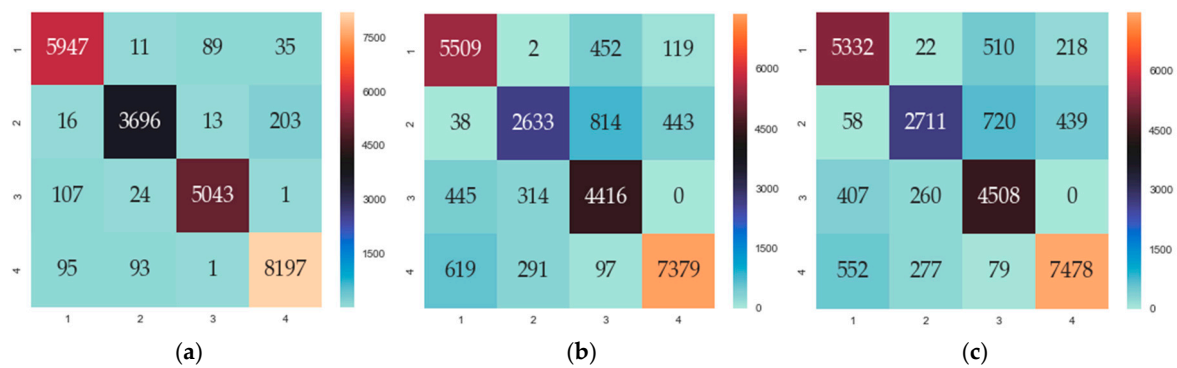


Figure 4. The confusion matrix of the three algorithms (1: single methane; 2: single ethylene; 3: air; 4: mixture). (a) The confusion matrix of the random forest (RF) classifier. (b) The confusion matrix of the logistic regression (LR) classifier. (c) The confusion matrix of the support vector machine (SVM) classifier.

5.4. Quantitative Analysis for Gas Mixture

Quantitative analysis for a gas mixture should be carried out after qualitative identification, in which the concentration estimation for a single gas and mixed gas by optimized SVR is carried out.

The number of particles and iterations were 81 and 10, respectively. The kernel function of SVR was selected as “rbf”, and 10-fold cross-validation was used in the training process. First, the training set was used for model training, and the best combination of C and γ was selected. Then, the test set was used for testing, and the concentration estimation results were obtained, with the determination coefficient R^2 (Equation (12), in which \hat{y}_i is the estimation value, \bar{y} is the average of the actual concentration, and y_i is the actual value) of the test samples as the evaluation criteria of the model to estimate ability. The value of R^2 is between 0 and 1, and the closer it is to 1, the better the regression model. The selected values of C and γ in different categories are shown in Table 4. The prediction effects of the four classes based on SVR improved by PSO model are shown in Figure 5. It can be seen from the fitting curve in Figure 5 that the gas concentration fitting effects for the four categories were very good, except for the deviation of some sample points. The errors between the predicted and actual values fluctuated around 0, which indicates that the fitting effects were very good. In [7], they also used this way to compare three feature extraction methods.

$$R^2 = \frac{\sum(\hat{y}_i - \bar{y})^2}{\sum(y_i - \bar{y})^2}. \quad (12)$$

Table 4. Parameters and concentration estimation results of different categories.

Categories	Single Gas		Mixed Gas	
Components	Methane	Ethylene	Methane	Ethylene
C	22,481	13,892	27,047	8546
γ	2.86	0.45	0.44	0.28
R^2	0.996	0.979	0.979	0.828

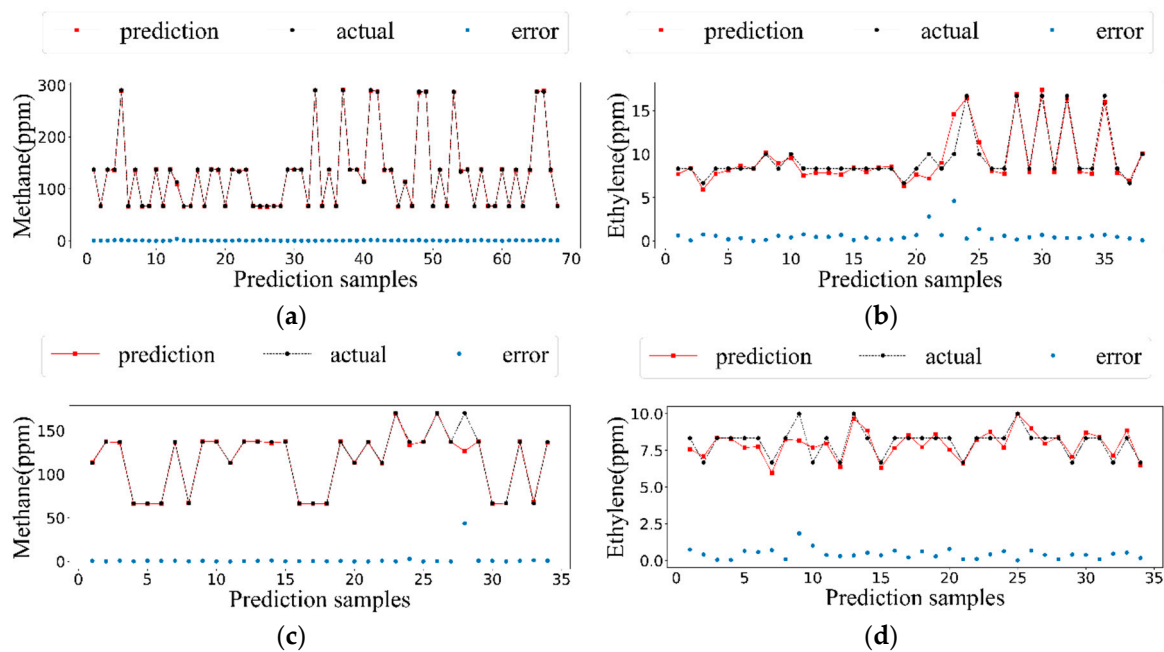


Figure 5. The predicted concentration along with the actual value. (a) The prediction and actual concentrations of single methane. (b) The prediction and actual concentrations of single ethylene. (c) The prediction and actual concentrations of methane in mixture. (d) The prediction and actual concentrations of ethylene in mixture.

In order to prove the regression effect of SVR optimized by the PSO algorithm, Figure 6 shows the comparison between our proposed methodology and SVR. It can be seen from Figure 6 that the approach based on SVR optimized by the PSO algorithm provided a smaller prediction error than SVR, which proves that the regression effect does improve through our method.

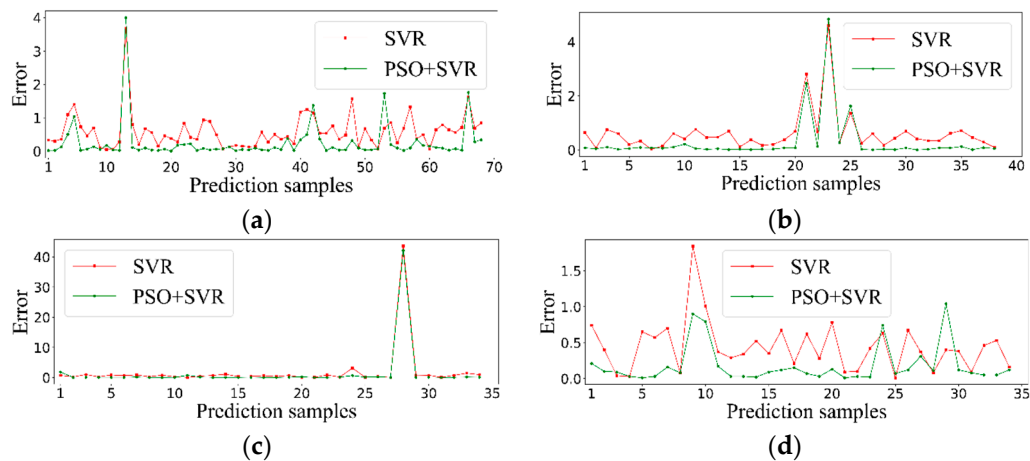


Figure 6. Prediction error provided by particle swarm optimization (PSO) + SVR (green) and SVR (red). (a) The prediction error of single methane. (b) The prediction error of single ethylene. (c) The prediction error of methane in mixture. (d) The prediction error of ethylene in mixture.

6. Conclusions

In this work, a novel qualitative and quantitative analysis strategy was proposed to provide accurate analysis of multicomponent gas mixtures. The proposed strategy combined PCA with random forest (PCA + RF) for identification. PCA can extract the principal components that contain most of the information and reduce the redundant factors. Random forest, as a classifier, was used to identify

the gas mixture. The methodology also used SVR optimized by PSO as a tool to quantify the gas component of a mixture.

The experimental results show that the best identification performance was obtained by PCA + RF compared with LR and SVM. Its recognition rate was 97%, a gain of 12% compared with LR and SVM. SVR optimized by PSO had a better regression result for every gas component than SVR, and at the same time, it solved the problem of selecting the hyperparameters of SVR.

Author Contributions: Funding acquisition, S.F.; resources, S.F.; project administration, S.F.; methodology, Z.L.; writing—original draft preparation, Z.L.; Visualization K.X.; Validation D.H.

Funding: This research was funded by (Key research and development project from Hebei Province, China) grant number (19210404D); (Key research project of science and technology from Ministry of Education of Hebei Province, China) grant number (ZD2019010); (The Project of Industry-University Cooperative Education of Ministry of Education of China) grant number (201801335014). And the APC was funded by (19210404D).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Asal, M.; Nasirian, S. Acetone gas sensing features of zinc oxide/tin dioxide nanocomposite for diagnosis of diabetes. *Mater. Res. Express* **2019**, *9*, 095093. [[CrossRef](#)]
2. Pagonas, N.; Vautz, W.; Seifert, L.; Slodzinski, R.; Jankowski, J.; Zidek, W.; Westhoff, T.H. Volatile organic compounds in uremia. *PLoS ONE* **2012**, *9*, e46258. [[CrossRef](#)] [[PubMed](#)]
3. Gancarz, M.; Wawrzyniak, J.; Gawrysiak-Witulska, M.; Wiącek, D.; Nawrocka, A.; Rusinek, R. Electronic nose with polymer-composite sensors for monitoring fungal deterioration of stored rapeseed. *Int. Agrophys.* **2017**, *3*, 317–325. [[CrossRef](#)]
4. Rusinek, R.; Gancarz, M.; Krekora, M.; Nawrocka, A. A novel method for generation of a fingerprint using electronic nose on the example of rapeseed spoilage. *J. Food Sci.* **2019**, *1*, 51–58. [[CrossRef](#)] [[PubMed](#)]
5. Jiang, H.; Zhang, M.; Bhandari, B.; Adhikari, B. Application of electronic tongue for fresh foods quality evaluation: A review. *Food Rev. Int.* **2018**, *8*, 746–769. [[CrossRef](#)]
6. Qiu, S.; Wang, J. The prediction of food additives in the fruit juice based on electronic nose with chemometrics. *Food Chem.* **2017**, *230*, 208–214. [[CrossRef](#)] [[PubMed](#)]
7. Men, H.; Fu, S.; Yang, J.; Cheng, M.; Shi, Y.; Liu, J. Comparison of SVM, RF and ELM on an electronic nose for the intelligent evaluation of paraffin samples. *Sensors* **2018**, *18*, 285. [[CrossRef](#)] [[PubMed](#)]
8. Gorji-Chakespari, A.; Nikbakht, A.M.; Sefidkon, F.; Ghasemi-Varnamkhasti, M.; Brezmes, J.; Llobet, E. Performance Comparison of Fuzzy ARTMAP and LDA in Qualitative Classification of Iranian Rosa damascena Essential Oils by an Electronic Nose. *Sensors* **2016**, *16*, 636. [[CrossRef](#)]
9. Gu, Q.; R Michanowicz, D.; Jia, C. Developing a Modular Unmanned Aerial Vehicle (UAV) Platform for Air Pollution Profiling. *Sensors* **2018**, *18*, 4363. [[CrossRef](#)]
10. Morawska, L.; Thai, P.K.; Liu, X.; Asumadu-Sakyi, A.; Ayoko, G.; Bartonova, A.; Bedini, A.; Chai, F.; Christensen, B.; Dunbabin, M.; et al. Applications of low-cost sensing technologies for air quality monitoring and exposure assessment: How far have they gone? *Environ. Int.* **2018**, 286–299. [[CrossRef](#)]
11. Rai, A.C.; Kumar, P.; Pilla, F.; Skouloudis, A.N.; di Sabatino, S.; Ratti, C.; Yasar, A.; Rickerby, D. End-user perspective of low-cost sensors for outdoor air pollution monitoring. *Sci. Total Environ.* **2017**, 691–705. [[CrossRef](#)] [[PubMed](#)]
12. Al Barakeh, Z.; Breuil, P.; Redon, N.; Pijolat, C.; Locoge, N.; Viricelle, J.P. Development of a normalized multi-sensors system for low cost on-line atmospheric pollution detection. *Sens. Actuators B Chem.* **2017**, 1235–1243. [[CrossRef](#)]
13. Escobar, J.M.; Suescun, J.P.S.; Correa, M.A.; Metaute, D.O. Forecasting concentrations of air pollutants using support vector regression improved with particle swarm optimization: Case study in Aburrá Valley, Colombia. *Urban Clim.* **2019**. [[CrossRef](#)]
14. Na, J.; Jeon, K.; Lee, W.B. Toxic gas release modeling for real-time analysis using variational autoencoder with convolutional neural networks. *Chem. Eng. Sci.* **2018**, 68–78. [[CrossRef](#)]
15. Li, X.B.; Wang, D.S.; Lu, Q.C.; Peng, Z.R.; Wang, Z.Y. Investigating vertical distribution patterns of lower tropospheric PM_{2.5} using unmanned aerial vehicle measurements. *Atmos. Environ.* **2018**, 62–71. [[CrossRef](#)]

16. Li, M.; Wang, W.L.; Wang, Z.Y.; Xue, Y. Prediction of PM 2.5 concentration based on the similarity in air quality monitoring network. *Build. Environ.* **2018**, 11–17. [[CrossRef](#)]
17. Liu, H.; Wu, H.; Lv, X.; Ren, Z.; Liu, M.; Li, Y.; Shi, H. An intelligent hybrid model for air pollutant concentrations forecasting: Case of Beijing in China. *Sustain. Cities Soc.* **2019**. [[CrossRef](#)]
18. Zhang, Y.; Wang, J.; Bian, X.; Huang, X.; Qi, L. A continuous gas leakage localization method based on an improved beamforming algorithm. *Measurement* **2017**, 143–151. [[CrossRef](#)]
19. Zeng, L.; Long, W.; Li, Y. A novel method for gas turbine condition monitoring based on KPCA and analysis of statistics T2 and SPE. *Processes* **2019**, 7, 124. [[CrossRef](#)]
20. Liu, L. Research on logistic regression algorithm of breast cancer diagnose data by machine learning. In Proceedings of the 2018 International Conference on Robots & Intelligent System (ICRIS), Changsha, China, 26–27 May 2018.
21. Zhang, J.; Zheng, C.H.; Xia, Y.; Wang, B.; Chen, P. Optimization enhanced genetic algorithm-support vector regression for the prediction of compound retention indices in gas chromatography. *Neurocomputing* **2017**, 183–190. [[CrossRef](#)]
22. Fonollosa, J.; Sheik, S.; Huerta, R.; Marco, S. Reservoir computing compensates slow response of chemosensor arrays exposed to fast varying gas concentrations in continuous monitoring. *Sens. Actuators B Chem.* **2015**, 618–629. [[CrossRef](#)]
23. Gancarz, M.; Nawrocka, A.; Rusinek, R. Identification of volatile organic compounds and their concentrations using a novel method analysis of MOS sensors signal. *J. Food Sci.* **2019**, 8, 2077–2085. [[CrossRef](#)] [[PubMed](#)]
24. Lentka, Ł.; Smulko, J.M.; Ionescu, R.; Granqvist, C.G.; Kish, L.B. Determination of gas mixture components using fluctuation enhanced sensing and the LS-SVM regression algorithm. *Metrol. Meas. Syst.* **2015**, 3, 341–350. [[CrossRef](#)]
25. Roy, P.S.; Ryu, C.; Dong, S.K.; Park, C.S. Development of a natural gas Methane Number prediction model. *Fuel* **2019**, 246, 204–211. [[CrossRef](#)]
26. Murguia, J.S.; Vergara, A.; Vargas-Olmos, C.; Wong, T.J.; Fonollosa, J.; Huerta, R. Two-dimensional wavelet transform feature extraction for porous silicon chemical sensors. *Anal. Chim. Acta* **2013**, 1–15. [[CrossRef](#)] [[PubMed](#)]
27. Pan, Y.; Chen, S.; Qiao, F.; Ukkusuri, S.V.; Tang, K. Estimation of real-driving emissions for buses fueled with liquefied natural gas based on gradient boosted regression trees. *Sci. Total Environ.* **2019**, 741–750. [[CrossRef](#)] [[PubMed](#)]
28. Zhang, T.; Song, S.; Li, S.; Ma, L.; Pan, S.; Han, L. Research on Gas concentration prediction models based on LSTM multidimensional time series. *Energies* **2019**, 12, 161. [[CrossRef](#)]
29. Wei, N.; Li, C.; Duan, J.; Liu, J.; Zeng, F. Daily Natural gas load forecasting based on a hybrid deep learning model. *Energies* **2019**, 12, 218. [[CrossRef](#)]
30. Liu, Y.; Nie, F.; Gao, Q.; Gao, X.; Han, J.; Shao, L. Flexible unsupervised feature extraction for image classification. *Neural Netw.* **2019**, 65–71. [[CrossRef](#)]
31. Fonollosa, J.; Rodríguez-Luján, I.; Trincavelli, M.; Vergara, A.; Huerta, R. Chemical discrimination in turbulent gas mixtures with MOX sensors validated by gas chromatography-mass spectrometry. *Sensors* **2014**, 14, 19336–19353. [[CrossRef](#)]

