

Statistical classification of soft solder alloys by laser-induced breakdown spectroscopy: review of methods

R. Zdunek
rafal.zdunek@pwr.edu.pl

Faculty of Electronics, Wrocław University of Technology, Wrocław, Poland

M. Nowak

Faculty of Electronics, Wrocław University of Technology, Wrocław, Poland

E. Pliński

Faculty of Electronics, Wrocław University of Technology, Wrocław, Poland

This paper reviews machine-learning methods that are nowadays the most frequently used for the supervised classification of spectral signals in laser-induced breakdown spectroscopy (LIBS). We analyze and compare various statistical classification methods, such as linear discriminant analysis (LDA), quadratic discriminant analysis (QDA), partial least-squares discriminant analysis (PLS-DA), soft independent modeling of class analogy (SIMCA), support vector machine (SVM), naive Bayes method, probabilistic neural networks (PNN), and K-nearest neighbor (KNN) method. The theoretical considerations are supported with experiments conducted for real soft-solder-alloy spectra obtained using LIBS. We consider two decision problems: binary and multiclass classification. The former is used to distinguish overheated soft solders from their normal versions. The latter aims to assign a testing sample to a given group of materials. The measurements are obtained for several laser-energy values, projection masks, and numbers of laser shots. Using cross-validation, we evaluate the above classification methods in terms of their usefulness in solving both classification problems.

[DOI: <http://dx.doi.org/10.2971/jeos.2016.160061>]

Keywords: Laser-induced breakdown spectroscopy, soft solder alloys, supervised classification, principal component analysis, machine learning methods

1 INTRODUCTION

The most popular method of connecting electronic components on printed circuit boards (PCBs) is soft soldering. In this process, metallic material (solder) heated to the melting point (usually lower than 450°C) covers the connected elements. After the solder solidifies, an inseparable connection is obtained. Soft soldering can be performed in various ways. In manual assembly and repair of electronic parts, soldering irons (pencils, stations) are used. In an automated production process, more advanced soldering techniques are used, i.e., reflow and wave as well as laser soldering¹. Regardless of the method used, high-quality joints are obtained by using a suitable metal solder alloy, ensuring very good wettability of the alloy by using flux, and setting the correct melting temperature.

When the soldering parameters are not set correctly, the quality of the solder can decrease considerably. In particular, using a temperature that is too high may reduce the effectiveness of the flux. Overheating most often results in the formation of new layers of intermetallic compounds that weaken the solder, thereby increasing the risk of damage to an electrical circuit. This phenomenon occurs more frequently when lead-free solders are used [1]. When overheating occurs, a sol-

der surface is immediately covered with a layer of oxides and becomes desiccated.

To assess the quality of solders on a PCB, several inspection techniques can be used, including visual inspection, automated optical inspection [2, 3], analysis X-ray inspection (referred to as 2D or 3D X-ray computed tomography) [4]–[7], acoustic microscopy [8]–[10], and inspection by infrared laser systems [11, 12]. These methods provide a variety of opportunities for determining the quality of the solder by detecting cracks, air-filled voids in the solder, insufficient wetting, oversoldering, and bridging. However, they cannot be used to analyze the chemical composition, assign the solder to a given group (according to the EN: ISO 9453:2014 standard²), or determine whether the solder has been overheated or dried out. Such information is needed for assessing the quality of solder alloys. It is important for both solder and equipment manufacturers, who refer to technical documentation and standards, especially the RoHS and Waste Electrical and Electronic Equipment (WEEE)³.

Laser-induced breakdown spectroscopy (LIBS) [13]–[15], combined with statistical classification methods, is used in this pa-

¹IPC-7530: Guidelines for Temperature Profiling for Mass Soldering Processes (Reflow and Wave), Association Connecting Electronics Industries, May 2001

²ISO 9453:2014: Soft solder alloys - Chemical compositions and forms, International Organization for Standardization, April 2014

³Directive 2012/19/EU: Waste Electrical and Electronic Equipment, European Union, July 2012

per to tackle this problem. The foundations of the LIBS technique have been laid in the early 1960s through many studies [16]–[19]. A comprehensive review of this technique and its applications can be found in several recently published review papers [20]–[25] and books [26, 27]. LIBS is an atomic emission spectroscopy technique that can be used for chemical material analysis. It uses a short laser pulse to generate a high-temperature microplasma on the surface of a sample. The plasma is formed by the laser ablation of a very small amount (picograms to nanograms) of material, and it contains free electrons, excited atoms, and ions. Beginning a few microseconds after the end of the laser pulse, the plasma emits a continuous spectrum (continuum) in the range of 200–1000 nm. However, information about the structure of the analyzed material cannot be obtained directly from this spectrum. Thus, after another period of microseconds, the temperature of the plasma is lowered, and the discrete structure of the spectrum, which is essential for analysis and identification of plasma products, starts to emerge [28]–[30]. The parameters of the discrete spectrum, such as the wavelength, intensity, and shape, uniquely characterize the analyzed material.

Solder alloys used in soft soldering can be classified as leaded or lead-free (RoHS- and RoHS2-compliant⁴). Their chemical composition is specified by the EN ISO 9453:2014 standard. LIBS emission spectra should uniquely determine the chemical composition. However, when solder alloys differ only in the proportions of the same chemical elements, their spectra may be significantly correlated. The largest differences can be observed between leaded and lead-free solder alloys, where the detection of one or more emission lines of lead is quite simple. Many industrial devices use such an analytical approach, in addition to determining the percentage of lead in the alloy, on the basis of the respective calibration curves. If the type and quality of solder is expected to be assessed, and the alloys to be analyzed contain the same elements, it is advisable to use more advanced methods to analyze the observed spectra.

In many areas of research, LIBS-based data have been recently analyzed and classified using various statistical machine learning methods [31, 32]. Principal component analysis (PCA) is probably the most frequently used method for processing LIBS data. Examples include biomedical and environmental applications [33], phone manufacturer identification [34], and inspection of concrete aggregates recycled from demolished buildings [35]. In geology, Gottfried *et al.* [36] used PCA and partial least-squares discriminant analysis (PLS-DA) to classify carbonate, fluorite, and silicate geological materials. In a later study, Kim *et al.* [37] employed these methods for the rapid detection of heavy metals and oils in soil. Next, Zhu *et al.* [38] applied PLS-DA and support vector machine (SVM) to analyze LIBS data for sedimentary rocks. PLS-based computational tools have also been used to determine the composition of geological samples from Mars [39], for ash determination in coal [40], and to establish the fuel–air equivalence ratio [41]. Another approach to the analysis of LIBS data was presented by El Haddad *et al.* [42], who performed the on-site quantitative analysis of lead in real soil samples by using a series of artificial neural networks (ANNs). Other methods, such as lin-

ear discriminant analysis (LDA) and soft independent modeling by class analogy (SIMCA) [43], gave satisfactory results in the classification of soil and geomaterial samples. Senesi [44] provided a comprehensive review of the applications of LIBS in the classification of geomaterials with a focus on minerals and rocks. In archeology, PCA, PLS-DA [45], and ANNs [46] have been used to classify ceramics efficiently. Vítková *et al.* [47] applied LDA to analyze brick samples. In medicine, Kanawade *et al.* [48] used a similar computational technique to discriminate tissues during laser surgery. SIMCA, PLS-DA, SVM, classification and regression tree, and binary logistic regression are applied for the classification of human bones in [49]. Godoi *et al.* [50] tackled the problem of identifying toxic elements in toys using SIMCA, PLS-DA, and the K-nearest neighbor (KNN) method. Cisewski *et al.* [51] used SVM for the classification of a suspect powder to detect *Bacillus anthracis* spores. SVM was also successfully used by Liang *et al.* [52] to classify steel materials. In industrial applications, SVM, KNN, and the naive Bayes (NB) method are applied for the automatic sorting of aluminum alloys [53]. The analysis of variance, which is closely related to LDA, is used in [54] for the depth-profile analysis of galvanized steel sheets. Comprehensive reviews of statistical tools used for the identification and classification of LIBS data can be found in [24, 25].

In this paper, we discuss the application of LIBS technology to the classification and identification of soft solder alloys. We consider two decision problems. The first is concerned with binary classification that aims to discriminate overheated soft solders from their normal versions using LIBS spectra. The other uses multiclass classification to identify a group of materials to which a given sample belongs. Various statistical machine-learning methods are studied with respect to both classification problems. We review the most popular methods, including PLS-DA, SIMCA, LDA, QDA, SVM, KNN, NB, and one version of ANN. As mentioned above, all these methods have already been used in LIBS technology in various applications. In the experimental section, we discuss their effectiveness in solving these two classification problems.

The remainder of this paper is organized as follows: Section 2 presents the experimental setup and a short description of the analyzed solder materials. The statistical tools are described in Section 3. The classification results are presented in Section 4. Finally, the conclusions are drawn in Section 5.

2 EXPERIMENTAL STUDY

The testing instrumental configuration, which is shown in Figure 1, consists of a KrF excimer laser system, an LIBS spectrometer (which is connected to the optical head for observing plasma light by a fiber optic cable and a reflective collimator with a plano-convex lens), and a computer running the SpectraSuite software (Ocean Optics, USA).

2.1 KrF (248 nm) excimer laser system

The laser system used in the experiments consists of the CNC Optec Promaster with the excimer KrF (248 nm) ATL

⁴Restriction of Hazardous Substances Directive, European Union, <http://www.rohs.eu>

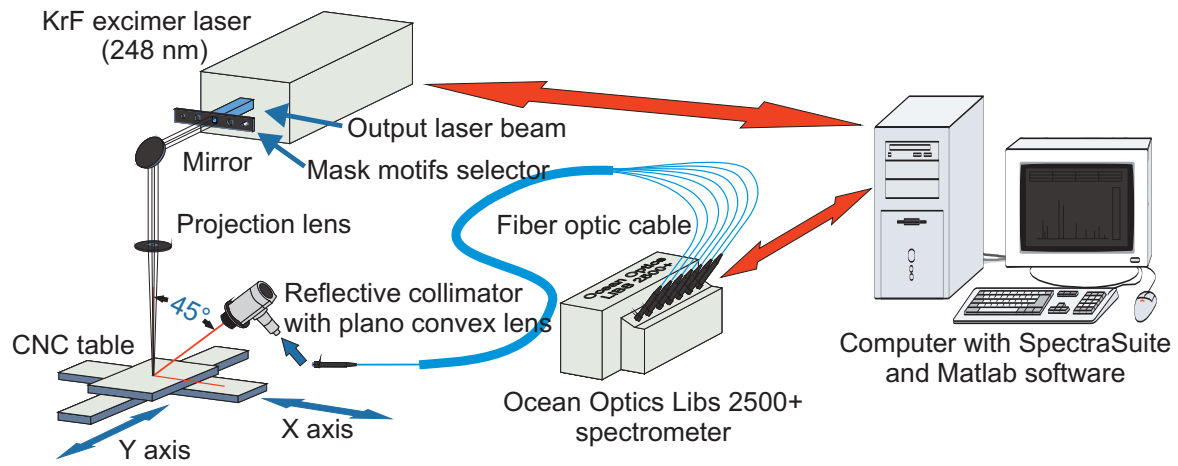


FIG. 1 LIBS experimental setup.

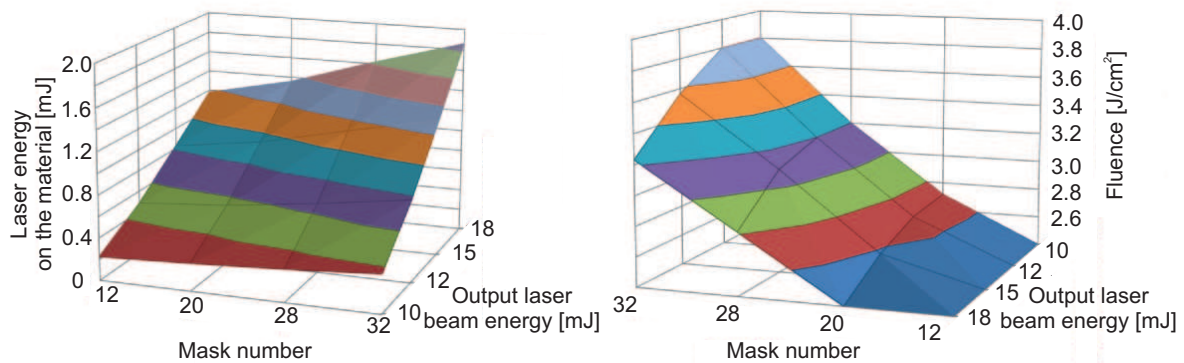


FIG. 2 (a) Real energy and (b) fluence values on the material, depending on the output laser energy and mask.

Lasertechnik ATLEX-300-SI-248 laser. The basic parameters of the output laser beam are as follows:

- energy: 4.5–22 mJ
- size: $6 \times 4 \text{ mm}^2$
- duration: 5–6 ns
- average power: $<6 \text{ W}$
- pulse repetition rate: 1–300 Hz

The laser source is combined with the optical system and a selector for choosing from among 32 masks with various motifs (circle, bar-shaped, and square apertures) and sizes. The optical system ensures the demagnification of the mask's size on the material at a rate of $\sim 10.45\times$. The laser beam in the workspace has a maximum energy of 2.03 mJ (measured using a Thorlabs energy meter with the ES111C pyroelectric energy sensor), and its size ranges from 24 to 240 μm , regardless of the shape of the projection mask. In this research, we used four square masks with sizes of 98×98 , 144×144 , 191×191 , and $240 \times 240 \mu\text{m}^2$ and four output beam energies: 10, 12, 15, and 18 mJ. The real energy and fluence values of the material are shown in Figure 2. Five shots are taken at one location on the sample. Precise movement of the material is obtained by using the computerized numerical control table in the Optec system.

2.2 LIBS device

The plasma light emission is observed using a spectrometer (Ocean Optics Libs 2500+) in the bandwidth range of

295–635 nm (three of seven channels) with a spectral resolution of 0.1 nm (FWHM). The trigger output of the laser is used to trigger the detection system. The integration time of the CCD array amounts to 1 ms, with a gate delay of 4 μs from the beginning of a laser pulse (a minimal value resulting from delays in the CCD arrays and a delay in the laser trigger). Direct acquisition of a plasma plume is performed by the optical head, which includes a reflective collimator (Thorlabs RC12SMA-P01) with a plano-convex lens (LA-4306-ML; $f = 40 \text{ mm}$) and a seven-channel sampling probe (BUN-7, Ocean Optics). It was placed at an angle of 45° with respect to the direction normal to the sample surface. The spectra from three channels are recorded by the SpectraSuite software and concatenated into one. Then, the continuum component (background) is removed by applying a denoising method based on the same concept as in [55] but fully automated.

2.3 Materials

The experiments are conducted using five soft solder alloys produced by the Cynel-Unipress⁵. Two of them are lead–tin alloys, and the rest, instead of lead, have more tin and different proportions of silver and copper. The symbols and the chemical compositions, specified by the manufacturer of the *Spektromaxx* spectrometer⁶, are listed in Table 1.

The test alloys are divided into two groups. One group contains the reference alloys, which are equivalent to solder made

⁵Phage Cynel-Unipress Co. Ltd.: solders, July 2015

⁶SPECTRO Analytical Instruments GmbH, July 2015

Name	Flux/No. core/%	Sn%	Pb%	Ag%	Cu%
S-Sn60Pb40	SW26/3/2.5	59.9	40.02	0.0003	0.0014
S-Pb70Sn27Ag	SW26/3/1.8	26.93	69.8	3.01	0.038
S-Sn99Cu1	F-SW32/3/3	98.85	0.020	0.0004	0.98
SAC 305 (Sn96.5Ag3Cu0.5)	PRO/1/1.5	96.53	0.014	2.97	0.46
S-Sn97Cu3	–	96.88	0.029	0.028	3.03

TABLE 1 Parameters of the solder alloys used in the tests, according to EN ISO 9453:2014, Flux [DIN 8511].

properly but covered with an oxide layer as a result of aging (Figure 3, top), and the other consists of overheated samples (Figure 3, bottom).

A hot-air gun with the air temperature set to 450°C is used to overheat the samples. The alloys are subjected to high temperature until the symptoms of overheating (described in Section 1) are observed. Consequently, the surfaces of the alloys change owing to the evaporation of the flux, their colors become dull, and surface tension is observed.

Ten testing alloys (five in each group) are selected. The spectral data are collected in 50 series of five laser shots in one location on each sample. Taking into account the settings described in Section 2.1, we use four types of masks and four energy values for the ten types of samples, which gives a total of 40,000 test shots (8000 series). Each series of measurements is performed at a pulse repetition rate equal to 1 Hz. Examples of emission spectra obtained with our measurement system at 12 mJ and using mask 12 are shown in Figure 4.

3 STATISTICAL ANALYSIS

The proposed methodology is based on the supervised classification of LIBS data, assuming that training data can be easily obtained. This approach is useful for the identification or discrimination of soft solder alloys, especially in decision problems, where a given testing sample must be classified with respect to a certain group of training samples. For example, one should decide whether the analyzed sample is overheated. In this case, we have a binary decision problem, which is easily solved using the standard SVM.

A more difficult decision problem occurs when we have more classes, for example, if we need to determine the group of materials to which a testing sample belongs. We assume that we have a dictionary or database of LIBS spectra of soldering materials that can be found on PCBs.

Let the observed LIBS spectrum be represented by the vector $\mathbf{x} \in \mathbb{R}^I$. The number I determines the spectral resolution, and it is not necessarily the number of subbands observed in one channel. Multichannel registrations can be concatenated. Hence, it might be a large number. In supervised classification, we need to have the training samples, i.e., the LIBS spectra of the most relevant materials to be analyzed. Let $\mathcal{D} = \{(\mathbf{x}_i^{(r)}, y_i^{(r)}), t = 1, \dots, T\}$ be the set of T training samples. Each $\mathbf{x}_i^{(r)}$ contains the LIBS spectrum of the known

material (solder alloy) that belongs to the group (class) indicated by $y_i^{(r)}$. We assume we have C groups of materials.

The aim of training is to find a classification rule or classifier \mathcal{F} such that $\mathcal{F}(\mathbf{x}_i^{(r)}) \rightarrow y_i^{(r)}$ for $t = 1, \dots, T$. The mapping can be obtained using many classification methods. In what follows, we attempt to find the most efficient classifier for a given classification problem.

The efficiency of the training is evaluated in the testing process: $\mathcal{F}(\mathbf{x}^{(t)}) \rightarrow y^{(t)}$, where $\mathbf{x}^{(t)}$ is the testing sample, and $y^{(t)}$ is the index of the class returned by the trained classifier \mathcal{F} . The quality of classification can be easily evaluated, e.g., by using the n -fold cross-validation (CV) technique [31, 32].

3.1 Principal component analysis

Let the training vectors $\{\mathbf{x}_i^{(r)}\}$ be regarded as realizations of a multivariate stochastic process $\mathcal{X} = \{\mathbf{x}_t : t = 1, \dots, T\}$. They form an inhomogeneous cloud of points in the space \mathbb{R}^I . The heterogeneity is justified by the spiky nature of LIBS spectra (see, e.g., Figure 4); only a few variables (emission lines) in each random vector \mathbf{x}_t are highly active. The low-activity variables generate the background, which is partially removed in the preprocessing stage. If the observed spectra have high resolution and the number of analyzed materials (classes) is much lower than the number of spectral points I , the variance of the random variables in \mathbf{x}_t can be quite diverse. In this case, we can easily find such orthogonal directions in the high-dimensional cloud of data points along which the variance is maximal. Such directions in \mathbb{R}^I will be referred to as feature vectors. Hence, the data points in \mathbb{R}^I can be modeled by a low-dimensional geometric object, which motivates the use of PCA [56, 57].

The random variables in \mathbf{x}_t are assumed to be correlated by the covariance matrix

$$\mathbf{C}_X^{(t)} = \mathcal{E}(\tilde{\mathbf{x}}_t \tilde{\mathbf{x}}_t^T) \in \mathbb{R}^{I \times I}, \quad (1)$$

where $\tilde{\mathbf{x}}_t = \mathbf{x}_t - \mathcal{E}(\mathbf{x}_t)$ is the vector of centralized random variables, and $\mathcal{E}(\cdot)$ is the expectation operator. Assuming that the stochastic process is ergodic, we can approximate the covariance matrix $\mathbf{C}_X^{(t)}$ by its empirical version $\hat{\mathbf{C}}_X$, which is symmetric and positive semidefinite. Thus, by using eigenvalue decomposition, we have $\hat{\mathbf{C}}_X = \mathbf{V} \mathbf{\Lambda} \mathbf{V}^T$, where $\mathbf{V}^T \mathbf{V} = \mathbf{I}$. The eigenvectors of $\hat{\mathbf{C}}_X$, which are expressed by the columns of $\mathbf{V} = [\mathbf{v}_1, \dots, \mathbf{v}_I] \in \mathbb{R}^{I \times I}$, determine mutually orthogonal feature vectors. The entries of the real diagonal matrix $\mathbf{\Lambda} = \text{diag}\{\lambda_i\} \in \mathbb{R}^{I \times I}$ are the corresponding eigenvalues. Let

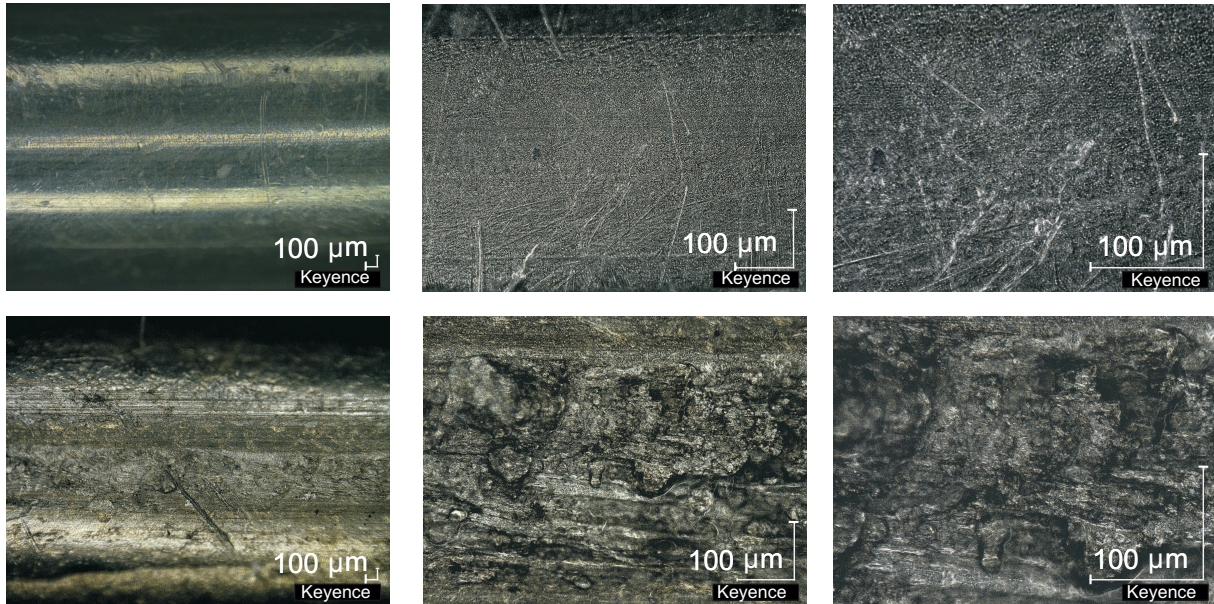


FIG. 3 Surface structure of the S-Sn97Cu3 alloy: (top) natural; (bottom) overheated; magnification rates: (left) 100; (middle) 500; (right) 1000.

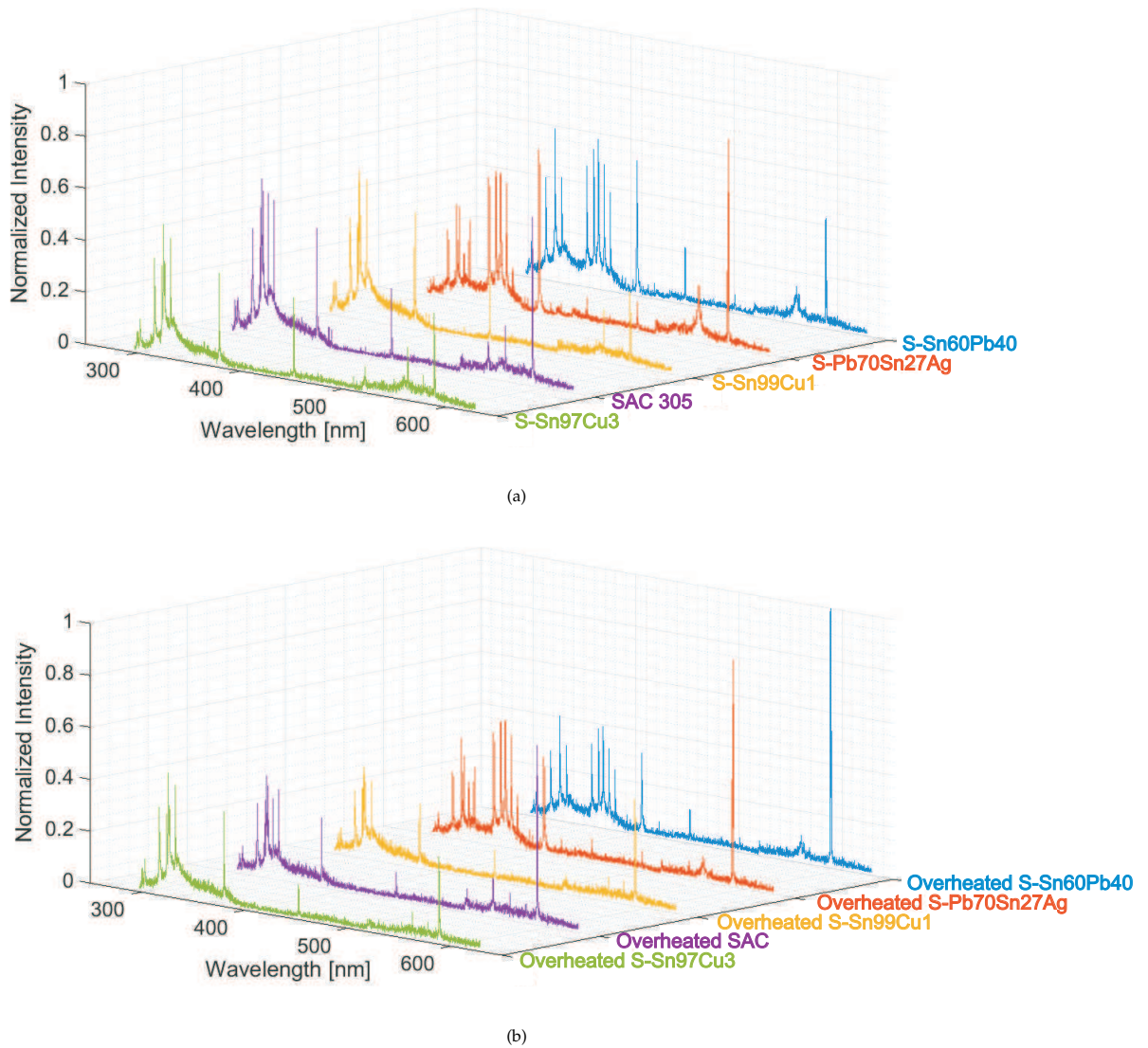


FIG. 4 LIBS spectra of soft solder alloy: (a) normal and (b) overheated. Spectra are recorded at an energy of 12 mJ using mask 12 (98 × 98 μm²) on the third shot.

$V_J = [v_1, \dots, v_J] \in \mathbb{R}^{I \times J}$ be a submatrix created from the first J column vectors in V_J span the basis for the following orthog-
 J eigenvectors that correspond to the largest eigenvalues. The

onal linear mapping:

$$z_i^{(r)} = V_j^T \tilde{x}_i^{(r)}, \quad (2)$$

where $\tilde{x}_i^{(r)}$ are realizations of \tilde{x}_t . The row vectors of $Z^{(r)} = [z_1^{(r)}, \dots, z_T^{(r)}] \in \mathbb{R}^{J \times T}$ determine the principal components (PCs). They are mutually uncorrelated, and because the eigenvalues are sorted in decreasing order, we have $\text{Var}\{z_1^{(r)}\} \geq \text{Var}\{z_2^{(r)}\} \geq \dots \geq \text{Var}\{z_j^{(r)}\}$, where $z_j^{(r)}$ is the j -row vector of $Z^{(r)}$.

Because $J \ll I$ and $J < T$, there is no need to calculate all the eigenvectors of \hat{C}_X . To calculate only a few dominant eigenvectors, we can use the stabilized version of the Lanczos iterations [58], which is implemented in MATLAB in the `eigs` function.

The number J can be roughly estimated by observing the behavior of the eigenvalues $\{\lambda_i\}$. The ratio of the variance explained by J PCs to the total variance is given by

$$\xi = \frac{\sum_{i=1}^J \lambda_i}{\sum_{i=1}^I \lambda_i} 100\%. \quad (3)$$

Hence, J should be as small as possible but, on the other hand, it should also be selected to maximize ξ . The problem of determining the optimal number of PCs has been widely discussed in the literature, e.g., [59]–[61]. In our approach, we set $J = 30$, for which $\xi \cong 90\%$. Thus, this choice considerably reduces the dimensionality while retaining nearly 90% of the explained variance.

3.2 K-nearest neighbor

The k -nearest neighbor (KNN) method [31, 32] is a fundamental method for classification and regression. Given the unlabeled testing sample $x^{(test)}$ and the set \mathcal{D} containing labeled training samples, the aim of KNN is to find k samples from the set \mathcal{D} that are the most similar to $x^{(test)}$ according to some metric. The predicted class of the sample $x^{(test)}$ is determined by majority voting.

The number k can be regarded in terms of penalty or regularization. For $k = 1$, the method is the simplest, and it is recommendable when the number of training samples is large and unperturbed with outliers. If $k = T$, KNN predicts the class of majority voting, which leads to strong oversmoothing. When some outliers are expected to occur, a few nearest neighbors should be used (often $k < 10$). Our observations show that the LIBS data obtained for many soft solders are not considerably perturbed with spiky outliers. We also check experimentally that for our measurements, the best classification accuracy is obtained for $k = 1$. For this case, the decision rule is given by $y^{(t)} = \arg \min_{1 \leq t \leq T} D(x^{(test)} || x_t^{(r)})$, where $D(x^{(test)} || x_t^{(r)})$ is the dissimilarity measure between both arguments. The Euclidean distance is the most frequently used, and it is optimal for samples normally distributed in classes. The LIBS spectra have a spiky nature, and the classes can sometimes differ in the magnitudes of only few emission lines. Hence, the Euclidean distance does not seem to be optimal for this application. In our tests, we also used the cosine measure, which

expresses the similarity in terms of the angle between the unit length vectors. In contrast to the former, the cosine measure is normalized and may be more suitable for comparing nonnegative data, such as LIBS spectra.

KNN can be directly applied to observed LIBS spectra, but in this case, it might be inefficient, especially because the number of spectral subbands (I) is very large. In our experiments, we analyze both cases, i.e. when KNN is applied directly to the high-dimensional LIBS data as well as to the PCs given by Eq. (2).

3.3 Linear discriminant analysis

PCA assumes the convexity and linear separability of classes, but the information on their labels is neglected. In supervised learning, class-specific linear models usually work better than linear dimensionality reduction alone. This motivates us to use LDA [31, 32], which is based on Fisher’s linear discriminant, for the multiclass classification problem.

In PCA, we attempt to orthogonally diagonalize the empirical covariance matrix \hat{C}_X by maximizing the Rayleigh quotient:

$$v_j^{(PCA)} = \arg \max_{v_j} \frac{v_j^T \hat{C}_X v_j}{v_j^T v_j}, \quad \text{s.t. } v_j^T v_j = 1, \quad (4)$$

for $j = 1, \dots, J$.

In LDA, the generalized Rayleigh quotient is maximized:

$$v_j^{(LDA)} = \arg \max_{v_j} \frac{v_j^T \hat{C}_B v_j}{v_j^T \hat{C}_W v_j}, \quad (5)$$

for $j = 1, \dots, C - 1$. The matrix \hat{C}_B represents the empirical covariance matrix of the class means. It is expressed as

$$\hat{C}_B = \sum_{c=1}^C N_c (\bar{x}_c^{(r)} - \bar{x}^{(r)}) (\bar{x}_c^{(r)} - \bar{x}^{(r)})^T, \quad (6)$$

where $\bar{x}_c^{(r)}$ is the sample mean of the c^{th} class, $\bar{x}^{(r)}$ is the total empirical mean, and N_c is the number of training samples in the c^{th} class. The matrix \hat{C}_B represents between-class scattering or the mean distance between the centroids of classes. Obviously, this quantity should be maximized. The matrix \hat{C}_W in Eq. (5) expresses the within-class scattering, which can be modeled as follows:

$$\hat{C}_W = \sum_{c=1}^C \sum_{k \in \mathcal{N}_c} (x_k^{(r)} - \bar{x}_c^{(r)}) (x_k^{(r)} - \bar{x}_c^{(r)})^T, \quad (7)$$

where \mathcal{N}_c is the set of indices of the training samples that belong to the c^{th} class.

The problem in Eq. (5) can be rewritten in an equivalent form that involves the Fisher criterion:

$$V^{(LDA)} = \arg \max_V \frac{\det(V^T \hat{C}_B V)}{\det(V^T \hat{C}_W V)}, \quad (8)$$

where $V = [v_1, \dots, v_{C-1}]$. The nominator in Eq. (8) represents the variance of the class means, and its denominator refers

to the variance of individual classes. Hence, LDA attempts to find the projection that maximizes the variance of the class means and minimizes the variance of individual classes.

It is well known that any solution to the maximization problem in Eq. (5) satisfies the generalized eigenvalue equation

$$\hat{C}_B v_j^{(LDA)} = \lambda_j \hat{C}_W v_j^{(LDA)}, \quad \text{for } j = 1, \dots, C-1, \quad (9)$$

where λ_j is the generalized eigenvalue that corresponds to the j^{th} generalized eigenvector $v_j^{(LDA)}$. Assuming that \hat{C}_W is nonsingular, the equation reduces to the standard eigenvalue equation $\hat{C}_W^{-1} \hat{C}_B v_j^{(LDA)} = \lambda_j v_j^{(LDA)}$. Because the matrix $\hat{C}_W^{-1} \hat{C}_B$ is not symmetric, the eigenvectors $\{v_j^{(LDA)}\}$ are not mutually orthogonal.

Note that $\text{rank}(\hat{C}_B) = C - 1$, but $\text{rank}(\hat{C}_W) = \min_c(N_c - C)$, which is usually much smaller than the length of $x_k^{(r)}$. Thus, the matrix \hat{C}_W is singular in our application, if LDA is applied directly to observed LIBS spectra. To tackle the singularity problem, several stability techniques can be applied, including various forms of regularization. In our approach, we combined LDA with PCA; i.e., LDA is applied to the low-dimensional samples that are obtained using Eq. (2).

After PCA is used, the matrix

$$V^{(LDA)} = [v_1^{(LDA)}, \dots, v_{C-1}^{(LDA)}] \in \mathbb{R}^{J \times (C-1)}$$

contains the basis for the following linear projection:

$$y_t^{(LDA)} = (V^{(LDA)})^T z_t^{(r)} \in \mathbb{R}^{C-1}, \quad (10)$$

where $z_t^{(r)}$ is given by Eq. (2).

The set $\{y_t^{(LDA)}\}$ contains the low-dimensional training vectors, where the classes should be linearly separable. Note that for any testing sample $x^{(test)} \in \mathbb{R}^I$, we need to apply a similar projection:

$$y_{(test)}^{(LDA)} = (V^{(LDA)})^T V_J^T x^{(test)} \in \mathbb{R}^{C-1}, \quad (11)$$

where V_J^T is obtained by PCA.

Then, the decision on the class to which the sample $x^{(test)}$ belongs can be taken using the KNN classifier with the Euclidean or Mahalanobis distance.

LDA assumes that the within-class scattering is modeled by one matrix, given in Eq. (7). However, this assumption does not have to be satisfied generally. Modeling it separately by one covariance matrix for each class leads to the quadratic discriminant [31]. This approach is used in the QDA classifier.

3.4 Partial least-squares discriminant analysis

The partial least squares (PLS) method is used for modeling a statistical relationship between two sets of observed variables. Originally, it was designed for solving regression problems in the social sciences [62], but recent studies demonstrate that it

has become increasingly popular in the classification of LIBS spectra [24, 25, 36, 44], [63]–[68].

The PLS regression aims to determine orthogonal latent variables that best explain the set of observed variables and simultaneously predict the output variables. In classification, the latent variables should maximize the covariance between the training variables and the output variables associated with the indices of classes.

Let $X = [x_1^{(r)}, \dots, x_T^{(r)}] \in \mathbb{R}^{I \times T}$ be the matrix of training LIBS spectra, and $Y = [y_1^{(r)}, \dots, y_T^{(r)}] \in \mathbb{R}^{C \times T}$ contain the samples of output variables that can be defined in many ways. If Y is a vector of indices of classes, then we have PLS1. In general, the output variable can be statistically dependent. In our multi-class classification tests, $Y = [y_{ct}]$ is a binary matrix with the following entries: $y_{ct} = 1$ if $c = y_t^{(r)}$, and $y_{ct} = 0$ otherwise, where $c = 1, \dots, C$. The index of the class to which the t^{th} training sample belongs is denoted by $y_t^{(r)}$. The observed variables should be centralized, as in PCA. Hence, $\tilde{X} = [\tilde{x}_t]$ and $\tilde{Y} = [\tilde{y}_t]$, where $\tilde{x}_t = x_t - \mathcal{E}(x_t)$ and $\tilde{y}_t = y_t - \mathcal{E}(y_t)$. If $C > 2$, the PLS regression used for classification is referred to as the PLS-DA (PLS discriminant analysis).

The fundamental model for the PLS regression has the form of bilinear equations:

$$\tilde{X} = PT^T + E, \quad (12)$$

$$\tilde{Y} = QU^T + F. \quad (13)$$

Similarly to PCA, $P \in \mathbb{R}^{I \times J}$ and $Q \in \mathbb{R}^{C \times J}$ are referred to as the loading matrices, but they are not orthogonal. The matrix $T \in \mathbb{R}^{T \times J}$ contains J latent vectors or X-scores. The Y-scores are represented by the matrix $U \in \mathbb{R}^{T \times J}$. If $J \ll I$, PLS can be regarded as a dimensionality reduction technique. The residual errors are modeled by the matrices $E \in \mathbb{R}^{I \times T}$ and $F \in \mathbb{R}^{C \times T}$.

There are many computational strategies for estimating the matrices T , P , U , and Q in the models in Eqs. (12) and (13). In the nonlinear estimation by iterative partial least-squares (NIPALS) algorithm [62], the column vectors of these matrices are estimated recursively from the deflated matrices:

$$\tilde{X}_{(j)} = \tilde{X}_{(j-1)} - p_j t_j^T, \quad (14)$$

$$\tilde{Y}_{(j)} = \tilde{Y}_{(j-1)} - q_j u_j^T, \quad (15)$$

where $j = 1, \dots, J$, and $\{p_j, t_j, q_j, u_j\}$ are the j^{th} columns of the matrices P , T , Q , and U , respectively. Initially, $\tilde{X}_{(0)} = \tilde{X}$ and $\tilde{Y}_{(0)} = \tilde{Y}$. The X- and Y-scores are assumed to belong to the corresponding spaces of the observed and predicted output variables. Thus,

$$t_j = \tilde{X}_{(j-1)}^T w_j \quad \text{s.t.} \quad t_j^T t_j = 1, \quad (16)$$

$$u_j = \tilde{Y}_{(j-1)}^T c_j \quad \text{s.t.} \quad u_j^T u_j = 1, \quad (17)$$

where $w_j \in \mathbb{R}^I$ and $c_j \in \mathbb{R}^C$ are the weight vectors. To predict Y from X , the X-scores should be maximally correlated with the Y-scores, which leads to the constrained optimization

problem:

$$\begin{aligned}
 (w_j, c_j) &= \arg \max_{\substack{w \in \mathbb{R}^I, \|w\|_2 = 1 \\ c \in \mathbb{R}^C, \|c\|_2 = 1}} \text{cov}^2(t, u) \\
 &= \arg \max_{\substack{w \in \mathbb{R}^I \\ c \in \mathbb{R}^C}} \frac{(w^T \mathcal{E}(\tilde{X}_{(j-1)} \tilde{Y}_{(j-1)}^T) c)^2}{(w^T w)(c^T c)} \\
 &= \arg \max_{w \in \mathbb{R}^I} (w^T w)^{-1} \max_{c \in \mathbb{R}^C} \frac{c^T C_{YX}^{(j-1)} w (C_{YX}^{(j-1)} w)^T c}{c^T c} \\
 &= \arg \max_{w \in \mathbb{R}^I} \frac{w^T C_{XY}^{(j-1)} C_{YX}^{(j-1)} w}{w^T w}, \tag{18}
 \end{aligned}$$

where $C_{XY}^{(j-1)} = \mathcal{E}(\tilde{X}_{(j-1)} \tilde{Y}_{(j-1)}^T) \in \mathbb{R}^{I \times C}$ is the covariance matrix between the variables in $X_{(j-1)}$ and $Y_{(j-1)}$, $C_{YX}^{(j-1)} = (C_{XY}^{(j-1)})^T$, and $c_j = C_{YX}^{(j-1)} w_j$. Assuming $C_{XY}^{(j-1)}$ is a full-rank matrix, formula (18) shows that w_j is the eigenvector of the symmetric and positive-definite matrix $C_{XY}^{(j-1)} C_{YX}^{(j-1)}$, associated with the leading eigenvalue. The largest singular value of $C_{XY}^{(j-1)}$ determines the covariance between t_j and u_j . The loading matrix P in Eq. (12) can be estimated by formulating the ordinary least-squares (LS) problem, which minimizes the residual error E in the Euclidean metrics. Considering Eq. (16), the j^{th} column vector of P is given by

$$p_j = (w_j^T \tilde{X}_{(j-1)} \tilde{X}_{(j-1)}^T w_j)^{-1} \tilde{X}_{(j-1)} \tilde{X}_{(j-1)}^T w_j. \tag{19}$$

The rank-one estimate of $X_{(j-1)}$ associated with the first latent variable is given by

$$\hat{X}_j = p_j t_j^T. \tag{20}$$

In the first iterative step, \hat{X}_j is also the rank-one estimate of X . For the Y-variables, we have $u_j = \tilde{Y}_{(j-1)}^T c_j$,

$$q_j = (c_j^T \tilde{Y}_{(j-1)} \tilde{Y}_{(j-1)}^T c_j)^{-1} \tilde{Y}_{(j-1)} \tilde{Y}_{(j-1)}^T c_j,$$

and $\hat{Y}_j = q_j u_j^T$.

In many versions of the NIPALS, the weight vectors $\{w_j\}$ are not directly estimated by the eigenvalue decomposition of the covariance matrix $C_{XY}^{(j-1)} C_{YX}^{(j-1)}$; rather, the concept of the power method is applied. The weight vectors are computed with the following iterative rules:

$$w_j = \frac{\tilde{X}_{(j-1)} u_j}{u_j^T u_j}, \quad \text{Normalization: } \|w_j\|_2 \rightarrow 1, \tag{21}$$

$$c_j = \frac{\tilde{Y}_{(j-1)} t_j}{t_j^T t_j}, \quad \text{Normalization: } \|c_j\|_2 \rightarrow 1. \tag{22}$$

The vectors t_j and u_j are updated according to Eqs. (16) and (17). Initially, u_1 can be chosen as one column of \tilde{Y}^T . Note that $w_j \propto \tilde{X}_{(j-1)} u_j \propto \tilde{X}_{(j-1)} \tilde{Y}_{(j-1)}^T c_j \propto \tilde{X}_{(j-1)} \tilde{Y}_{(j-1)}^T \tilde{Y}_{(j-1)} t_j \propto \tilde{X}_{(j-1)} \tilde{Y}_{(j-1)}^T \tilde{Y}_{(j-1)} \tilde{X}_{(j-1)}^T w_j$. Hence, w_j is an eigenvector of $C_{XY}^{(j-1)} C_{YX}^{(j-1)}$. The column vectors in T satisfy the orthogonality condition, i.e., $\forall i \neq j : t_i^T t_j = 0$.

In our tests, we used a modified version of NIPALS called the statistically inspired modification of PLS (SIMPLS) [69]. This

method is computationally more efficient and easier to interpret. In this approach, the covariance matrix $C_{XY}^{(j)}$ is updated recursively, but it is not calculated from the deflated matrices. Thus, $\forall j : \tilde{X}_{(j)} = \tilde{X}, \tilde{Y}_{(j)} = \tilde{Y}$, and $C_{XY}^{(0)} = \tilde{X} \tilde{Y}^T$. The latent vectors $\{t_j\}$ are assumed to be orthogonal:

$$\forall i \neq j : t_i^T t_j = w_i^T \tilde{X} \tilde{X}^T w_j = 0. \tag{23}$$

From Eqs. (23) and (19), we have $\forall i \neq j : p_i^T w_j = 0$, which means that the current w_j should be orthogonal to all previous X-loading vectors $\{p_1, \dots, p_i\}$ for $i < j$. Next, the X-loadings are projected onto the base $\{v_1, \dots, v_i\}$, created with the Gram-Schmidt orthogonalization. It means that $\forall i < j : v_i^T v_j = 0$. In such a base, the covariance matrix $C_{XY}^{(j-1)}$ is deflated by the following rule:

$$C_{XY}^{(j)} = C_{XY}^{(j-1)} - V_j (V_j^T C_{XY}^{(j-1)}), \tag{24}$$

where $V_j = [v_1, \dots, v_j]$. Finally, the Y-scores are orthogonalized with respect to the X-scores:

$$\forall j : u_j \leftarrow u_j - (t_i^T u_j) t_i \quad \text{for } i = 1, \dots, i-1. \tag{25}$$

The relationship between the output and input variables can be described by the multivariate regression model:

$$\tilde{Y} = \tilde{B} \begin{bmatrix} \mathbf{1}_T^T \\ \tilde{X} \end{bmatrix}, \tag{26}$$

where $\tilde{B} = [\hat{b}, B] \in \mathbb{R}^{C \times (I+1)}$ is the matrix of regression coefficients, and $\mathbf{1}_T = [1, \dots, 1] \in \mathbb{R}^T$ is a vector of all ones. Inserting the equations (12)–(13) to the model (26), we have $B = Q \tilde{W}^T \in \mathbb{R}^{C \times I}$, where $\tilde{W} = \left[\frac{w_1}{\|t_1\|_2}, \dots, \frac{w_I}{\|t_I\|_2} \right] \in \mathbb{R}^{I \times I}$. The vector $\hat{b} = T^{-1}(Y \mathbf{1}_T - B X \mathbf{1}_T) \in \mathbb{R}^C$ expresses the intercept term.

In the testing stage, the response for the testing sample $x^{(test)} \in \mathbb{R}^I$ can be readily calculated using the model (26). Thus,

$$\hat{y}^{(test)} = \tilde{B} \begin{bmatrix} 1 \\ \hat{x}^{(test)} \end{bmatrix}, \tag{27}$$

where $\hat{x}^{(test)} = x^{(test)} - \mathcal{E}(x^{(test)})$. Finally, the index of the testing sample is determined as $y^{(t)} = \max_c \{\hat{y}_c^{(test)}\}$.

One of the main advantages of the PLS regression is its high efficiency in working with a large set of input variables that can be partially dependent, whereas the number of observed samples may be relatively small. It is therefore particularly useful in the classification of LIBS spectra, where the number of spectral subbands is pretty large. Moreover, the emission spectrum of the analyzed material (solder alloy) usually contains a few emission lines that determine a spectral signature. Hence, some variables in any source signal (endmembr) might be correlated. The set of observed spectra is also not very large in practice. PLS also works well for collinear problems. It is also the case for the LIBS technology, where the difference between the LIBS spectra of various materials might be very small. Similarly to PCA, PLS extracts some hidden

factors (components) from the training LIBS spectra, but PLS extracts factors much more informative for classification. The factors not only capture the largest variance (as in PCA) but are the most correlated with the responses (output variables). The PLS regression is therefore much more robust in classifying LIBS spectra than the ordinary orthogonal regression with the PCs.

3.5 Soft independent modeling of class analogy

The soft independent modeling of class analogy (SIMCA) is a well-known supervised machine-learning method that was proposed for statistical pattern recognition by Wold in the 1970s [70, 71]. Since then, it has found many real-world applications, including the classification of LIBS spectra [43, 49, 50, 72].

It is based on the concept of PCA, but it does not search for global features holistically representing the whole set of training LIBS spectra. The global PCs do not necessary provide discriminant information. In SIMCA, PCA is applied separately to each class, which gives us relevant information on individual group structures. The testing sample is orthogonally projected onto the space spanned by PCs of each class, and the residual distances are calculated to evaluate the similarity of the testing sample to each class.

Let $\mathbf{X}_c^{(r)} = [\mathbf{x}_{t_c}^{(r)}] \in \mathbb{R}^{I \times T_c}$ contain T_c training samples that belong to the c^{th} class. By applying PCA to each $\mathbf{X}_c^{(r)}$, we obtain the matrix $\mathbf{V}_J^{(c)} \in \mathbb{R}^{I \times J}$ containing J feature vectors, and the PCs given by $\mathbf{z}_{t_c}^{(r)}$ according to the mapping (2). Selecting J PCs for the c^{th} class, the residual error between the training samples and the PC model is given by

$$\mathbf{E}^{(c)} = \tilde{\mathbf{X}}_c^{(r)} - \mathbf{V}_J^{(c)} \mathbf{Z}^{(c)}, \quad (28)$$

where $\tilde{\mathbf{X}}_c^{(r)} = [\tilde{\mathbf{x}}_{t_c}^{(r)}], \forall c : \tilde{\mathbf{x}}_{t_c}^{(r)} = \mathbf{x}_{t_c}^{(r)} - \mathcal{E}(\mathbf{x}_{t_c}^{(r)})$ and $\mathbf{Z}^{(c)} = [\mathbf{z}_{t_c}^{(r)}] \in \mathbb{R}^{T_c \times J}$. The mean distance between the samples assigned to the c^{th} class and the space spanned by their PCs can be expressed by the standard deviation of the residual error $\mathbf{E}^{(c)} = [e_{i,t_c}^{(c)}]$:

$$s_0 = \sqrt{\sum_{t_c=1}^{T_c} \sum_{i=1}^I \frac{(e_{i,t_c}^{(c)})^2}{(I-J)(T_c-J-1)}}. \quad (29)$$

As the error $e_{i,t_c}^{(c)}$ follows a normal distribution, the F -test is used to determine the critical distance at a given level of significance. Thus, $s_c = \sqrt{F_c s_0^2}$, where F_c is the F -value for $(I-J)$ and $(I-J)(T_c-J-1)$ degrees of freedom at the significance level α . This parameter determines a confidence region around each class, which can be interpreted as the threshold for the classification of a training sample as an outlier.

To classify the testing sample $\mathbf{x}^{(test)}$ to any group, it is first sequentially projected onto the spaces spanned by the PCs of each class. The projection is defined as follows:

$$\hat{\mathbf{x}}_c^{(test)} = \bar{\mathbf{x}}^{(c)} + \mathbf{V}_J^{(c)} (\mathbf{V}_J^{(c)})^T (\mathbf{x}^{(test)} - \bar{\mathbf{x}}^{(c)}), \quad (30)$$

where $\bar{\mathbf{x}}^{(c)}$ is the mean of the c^{th} class. The residual error $\mathbf{e}_c^{(test)} = \mathbf{x}^{(test)} - \hat{\mathbf{x}}_c^{(test)}$ is normally distributed with the standard deviation

$$s_c^{(test)} = \sqrt{\frac{(\mathbf{e}_c^{(test)})^T \mathbf{e}_c^{(test)}}{(I-J)}}. \quad (31)$$

If

$$s_c^{(test)} < s_c, \quad (32)$$

the sample $\mathbf{x}^{(test)}$ is considered to belong to the c^{th} class. Note that the condition (32) may be satisfied for multiple classes. Thus, SIMCA provides soft classification. If the hard classification is expected, then the sample $\mathbf{x}^{(test)}$ is assigned to the c^* -th class, if $c^* = \arg \min_{1 \leq c \leq C} F_c^{(test)}$, where $F_c^{(test)} = \left(\frac{s_c}{s_0}\right)^2$ is the F -value for $\mathbf{x}^{(test)}$.

3.6 Naive Bayes

The LIBS spectra that belong to the c^{th} class can be regarded as samples from the conditional probability distribution $p(\mathbf{x}|Y=c)$, where the discrete random variable $Y \in \{1, \dots, C\}$ takes the value c . Let us assume that we have prior knowledge on the distribution $p(Y)$, usually inferred directly from the training data. For the c^{th} class, it is given by the ratio

$$p(Y=c) = \frac{N_c}{T}, \quad (33)$$

where N_c is the number of training samples in the c^{th} class. By applying the Bayes rule, the probability of the class c , given the observation \mathbf{x} , can be represented by the posterior distribution:

$$p(Y=c|\mathbf{x}) = \frac{p(\mathbf{x}|Y=c)p(Y=c)}{\sum_{c=1}^C p(\mathbf{x}|Y=c)p(Y=c)}. \quad (34)$$

Neglecting the marginal distribution in Eq. (34), the Bayes classifier for the testing sample $\mathbf{x}^{(test)}$ is given by

$$\begin{aligned} \mathcal{F}(\mathbf{x}^{(test)}) &= \arg \max_{1 \leq c \leq C} p(Y=c|\mathbf{x}^{(test)}) \\ &= \arg \max_{1 \leq c \leq C} p(\mathbf{x}^{(test)}|Y=c)p(Y=c). \end{aligned} \quad (35)$$

The distribution $p(\mathbf{x}|Y=c)$ needs to be estimated; this can be done in many ways. In the naive Bayes (NB) classifier [31, 32], the random variables in $\mathbf{x} = [x_1, \dots, x_I]^T$ are assumed to be statistically independent, i.e., $p(\mathbf{x}|Y=c) = \prod_{i=1}^I p(x_i|Y=c)$. This assumption considerably simplifies the model and decreases computational cost but is often not fully satisfied in practice. As already mentioned, the emission lines of certain materials might be correlated, which violates the condition of independence. Nevertheless, NB often works well in many applications, especially for sparse features. LIBS spectra are nonnegative and have a spiky nature. Hence, the variables x_i should not be modeled by a Gaussian distribution. If the intensities of spectral lines were modeled by discrete values, then $p(x_i|Y=c)$ could be expressed by the multinomial distribution. However, such a model involves high computational cost. A good solution in practice is the use of a non-parametric

density estimation method, such as the kernel-smoothing density estimator:

$$p(x_i|Y = c) = N_c^{-1} \sum_{t_c=1}^{N_c} h_i^{(c)} K \left(\frac{x_i - x_{i,t_c}^{(r)}}{h_i^{(c)}} \right). \quad (36)$$

The parameters of the model (36) are estimated from the training samples $\{x_i^{(r)}\}$, where $x_{i,t_c}^{(r)}$ is the i^{th} entry of the training vector $x_{t_c}^{(r)}$ from the c^{th} class, $h_i^{(c)}$ is the smoothing parameter associated with the i^{th} variable, and $K(\cdot)$ is the kernel function. To enforce local smoothing, the kernel K is modeled with the Gaussian distribution. For simplicity, $h_i^{(c)} = h^{(c)}\sigma_i^{(c)}$, where $h^{(c)}$ is a constant in the c^{th} class and $\sigma_i^{(c)}$ is the empirical standard deviation of the i^{th} variable in the vectors $\{x_{t_c}^{(r)}\}$. To estimate $p(x|Y = c)$, the multidimensional kernels can also be used. In such a case, $p(x|Y = c)$ can be modeled by

$$p(x|Y = c) = N_c^{-1} \sum_{t_c=1}^{N_c} K_H \left(x - x_{t_c}^{(r)} \right). \quad (37)$$

The multidimensional kernel is given by

$$K_H(\xi) = \det(\mathbf{H})^{-1/2} K(\mathbf{H}^{-1/2}\xi), \quad (38)$$

where $\mathbf{H} \in \mathbb{R}^{I \times I}$ is a symmetric and positive-definite smoothing matrix and $K(\cdot)$ is a standard multivariate Gaussian distribution. Note that if \mathbf{H} is a diagonal matrix, the model (37) is equivalent to the product of Eq. (36).

NB also has some disadvantages that should not be neglected for our application. The probability density in Eq. (37) is estimated from training samples, and the estimate is better if the dimension x is smaller and more samples is used. However, the case in LIBS technology is usually the reverse. Moreover, on summing the same number of training samples in each class, the priors (33) are identical for each class, which is not informative. Hence, from the theoretical viewpoint, NB may not be optimal for classifying LIBS spectra.

3.7 Probabilistic neural network

The probabilistic neural network (PNN) is intrinsically related to the Bayes classifier, but it is implemented with the architecture of a feedforward multilayer neural network. It was proposed by Specht in 1990 [73] and is particularly useful for solving classification problems. PNN belongs to a family of artificial neural networks that are also applied for the classification of LIBS spectra [42, 63], [74]–[76].

Similarly to NB, PNN attempts to estimate the conditional distribution $p(x|Y)$ for each class. In the testing stage, PNN classifies $x^{(test)}$ according to the rule (35), given the prior $p(Y)$. However, the result of classification can be different from that with NB owing to a different implementation of the Bayes classifier.

PNN consists of four layers. The input layer contains I neurons: each of them receives one subband (the entry $x_i^{(test)}$) from the observed spectrum $x^{(test)}$. The input signals, after being centralized and normalized, are then given to the second

layer named the pattern layer. It consists of many hidden neurons grouped into C categories. In each category, there are as many neurons as the number of training vectors in this class. Each training vector is assigned to one neuron that has I input synapses receiving the signals from all subbands $\{x_i^{(test)}\}$. The neuron computes the Euclidean distance between the testing sample $x^{(test)}$ and the training sample $x_i^{(r)}$, and then the Gaussian radial basis function is used for activation. The output signals from all neurons in the pattern layer are then yielded to the third layer that performs the summation over each class. Hence, the summation layer contains C neurons. The outputs from these neurons are normalized to obtain estimates of the probability density function for each class. The hidden layers therefore play the role of the Gaussian kernel density estimator that was discussed in Subsection 3.6. The final (output) layer contains one output neuron, which compares the activations from the third layer, weights with the prior, and provides the index of the class to which the testing sample is assigned with the highest probability.

In the training stage, parameters such as standard deviations in the Gaussian activation functions are learned. They play a role similar to that played by the smoothing parameters in NB and can be estimated, e.g., using the cross-validation technique. Hence, one of the main advantages of PNN is fast training, which is much faster than in a backprojection network. Such a distributed architecture can also be readily parallelized but requires large memory resources.

3.8 Support vector machine

Many recently published studies [38], [51]–[53], [76]–[78] have shown that the SVM classifier [31, 32, 79] is also very efficient in a statistical analysis of LIBS spectra. The fundamental version of this classifier performs binary classification with linear separability of classes. It aims to find the hyperplane in the sample space that has the largest distance to the nearest training sample of any class.

Let $x_i^{(r)} \in \mathbb{R}^I$ be the training sample, and $\forall t : y_t \in \{-1, 1\}$ be the indicator of the class to which the t^{th} sample is assigned. The aim is to find the hyperplane $\mathcal{H} = \{x : w^T x + b = 0\}$ that best separates both classes. The vector $w \in \mathbb{R}^I$ is normal to \mathcal{H} , and $\frac{b}{\|w\|_2}$ is the perpendicular distance from \mathcal{H} to the origin in \mathbb{R}^I . The data points located closest to \mathcal{H} are referred to as the support vectors (SVs). The best separating hyperplane should maximize the distance between the SVs in both classes. Thus, it should satisfy the conditions $w^T x_i^{(r)} + b \geq +1$ for $y_t = +1$ and $w^T x_i^{(r)} + b \leq -1$ for $y_t = -1$. This gives us the constraints

$$\forall t : y_t (w^T x_i^{(r)} + b) \geq 1. \quad (39)$$

The equality in Eq. (39) occurs only for the SVs. The margin between the classes, i.e., the shortest distance between the SVs from opposite classes, is equal to $\frac{2}{\|w\|_2}$. Obviously, the margin should be maximized, which leads to minimization of $\|w\|_2$. Regarding the constraints in Eq. (39), the task of finding the best separating hyperplane reduces to the quadratic programming (QP) problem, subject to the following inequality con-

straints:

$$\min_{\mathbf{w}, b} \frac{1}{2} \|\mathbf{w}\|_2^2, \quad \text{s.t.} \quad \forall t : y_t(\mathbf{w}^T \mathbf{x}_t^{(r)} + b) - 1 \geq 0. \quad (40)$$

The Lagrangian associated with the problem in Eq. (40) has the form

$$\mathcal{L}(\mathbf{w}, b) = \frac{1}{2} \|\mathbf{w}\|_2^2 - \sum_{t=1}^T \alpha_t \left(y_t(\mathbf{w}^T \mathbf{x}_t^{(r)} + b) - 1 \right), \quad (41)$$

where $\forall t : \alpha_t \geq 0$ is the Lagrangian multiplier. It has a stationary point when $\frac{\partial}{\partial \mathbf{w}} \mathcal{L}(\mathbf{w}, b) = \mathbf{0}$, which leads to

$$\mathbf{w} = \sum_{t=1}^T \alpha_t y_t \mathbf{x}_t^{(r)}, \quad (42)$$

and the condition $\frac{\partial}{\partial b} \mathcal{L}(\mathbf{w}, b) = 0$ gives us

$$\sum_{t=1}^T \alpha_t y_t = 0. \quad (43)$$

By inserting Eqs. (42) and (43) into Eq. (41) and performing straightforward computations, we obtain the dual QP problem:

$$\min_{\boldsymbol{\alpha}} \left(\frac{1}{2} \boldsymbol{\alpha}^T \mathbf{H} \boldsymbol{\alpha} - \mathbf{e}^T \boldsymbol{\alpha} \right), \quad \text{s.t.} \quad \mathbf{y}^T \boldsymbol{\alpha} = 0 \quad \text{and} \quad \boldsymbol{\alpha} \geq \mathbf{0}, \quad (44)$$

where $\mathbf{H} = [h_{st}] \in \mathbb{R}^{T \times T}$, $h_{st} = y_s y_t (\mathbf{x}_s^{(r)})^T \mathbf{x}_t^{(r)}$, $\boldsymbol{\alpha} = [\alpha_t] \in \mathbb{R}^T$, $\mathbf{y} = [y_t] \in \mathbb{R}^T$, and $\mathbf{e} = [1, \dots, 1] \in \mathbb{R}^T$. The QP problem in Eq. (44) is convex and can be easily solved by many solvers using, e.g., the active set or interior point algorithm. Having found the Lagrangian multipliers $\boldsymbol{\alpha}$, we obtain the optimal vector \mathbf{w}_{optim} from Eq. (42) and then calculate the optimal intercept b_{optim} from

$$b_{optim} = - \frac{\max_{t: y_t = -1} \mathbf{w}_{optim}^T \mathbf{x}_t^{(r)} + \min_{t: y_t = 1} \mathbf{w}_{optim}^T \mathbf{x}_t^{(r)}}{2}. \quad (45)$$

The parameters \mathbf{w}_{optim} and b_{optim} uniquely determine the best separating hyperplane \mathcal{H} .

Let $\mathbf{x}^{(test)} \in \mathbb{R}^I$ be the testing sample. The class of $\mathbf{x}^{(test)}$ can be determined from the decision rule

$$\mathcal{F}(\mathbf{x}^{(test)}) = \text{sign} \left(\mathbf{w}_{optim}^T \mathbf{x}^{(test)} + b_{optim} \right). \quad (46)$$

If the training samples are not perfectly separable, i.e., if outliers exist, the constraint in Eq. (39) can be relaxed to the form $\forall t : y_t(\mathbf{w}^T \mathbf{x}_t^{(r)} + b) \geq 1 - \xi_t$, where $\forall t : \xi_t \geq 0$ is the slack variable. Obviously, the outlier samples are assumed to be rare in the entire training set; hence, the vector $\boldsymbol{\xi} = [\xi_t]$ is sparse. In this case, the primal QP problem in Eq. (40) takes the form

$$\min_{\mathbf{w}, b} \frac{1}{2} \|\mathbf{w}\|_2^2 + C_{\bar{\xi}} \|\boldsymbol{\xi}\|_1, \quad \text{s.t.} \quad y_t(\mathbf{w}^T \mathbf{x}_t^{(r)} + b) - 1 + \xi_t \geq 0, \quad (47)$$

where $\boldsymbol{\xi} \geq \mathbf{0}$, and $C_{\bar{\xi}} \geq 0$ is the soft-margin penalty parameter. Surprisingly, the problem in (47) transforms to the very simple dual form

$$\min_{\boldsymbol{\alpha}} \left(\frac{1}{2} \boldsymbol{\alpha}^T \mathbf{H} \boldsymbol{\alpha} - \mathbf{e}^T \boldsymbol{\alpha} \right), \quad \text{s.t.} \quad \mathbf{y}^T \boldsymbol{\alpha} = 0 \quad \text{and} \quad C_{\bar{\xi}} \geq \alpha_t \geq 0, \quad (48)$$

which can also be solved using many well-known QP solvers.

When the classes are not linearly separable, nonlinear SVM [31, 32] can be applied. In this classifier, the training samples are nonlinearly mapped to a higher-dimensional space using the so-called kernel tricks. In classification, several kernels are commonly used, e.g., the Gaussian, polynomial, sigmoidal, and multilayer perceptron kernels. If SVM is applied to the output from PCA, nonlinear separation seems unnecessary. The PCs are linearly uncorrelated, and the clusters are convex. This motivates the use of linear SVM. We have also experimentally confirmed this assertion by using nonlinear singular value decomposition with various trained kernels. In each case, the box constraint $C_{\bar{\xi}}$ in the soft margin was estimated from the training set with the quasi-Newton method (the Broyden–Fletcher–Goldfarb–Shanno method). A similar optimization tool was used to estimate the variance in the Gaussian kernel. We also tested various degrees of the polynomial from one to four. In each case, the best results were obtained for the linear classifier or when the polynomial degree was set to one.

The standard linear SVM is a binary classifier. Thus, it can be directly applied to binary decision problems, e.g., to decide whether a given solder alloy is overheated. When a sample can be classified into more than two classes, we can use one multiclass SVM or a larger number of standard SVM classifiers. We selected the latter; i.e., we use as many classifiers as there are classes. Each classifier is trained to recognize one class against the rest. Then, in the testing process, a testing sample is verified separately by each classifier. Note that this methodology incurs higher computational cost than the use of one multiclass classifier, but it offers many additional advantages. For example, if a testing sample cannot be identified by any trained classifier, we can apply another classifier only to the unrecognized sample, or we can assign this sample to some unknown class. Similarly, if a testing sample is recognized by more than one classifier, we can also repeat the classification with another, more efficient, classifier. This approach is particularly useful in practice, when outliers or other perturbations occur.

4 CLASSIFICATION RESULTS

In this section, we compare the algorithms discussed in Section 3 in terms of their efficiency in classifying the LIBS spectra of soft solder alloys. In the experiments, we used the soft solder alloys discussed in Section 2.3 and the LIBS device described in Section 2.2. The classification tools were implemented in MATLAB 2012 and run on a computational server equipped with two CPUs [Intel Xeon(R) X5650, 2.66 GHz].

We analyzed two classification problems:

- **A:** 2 classes: the samples of the normal solder alloys (listed in Table 1) form one class, and their overheated versions belong to the other class,
- **B:** 10 classes: the labels 1–5 correspond to the solder alloys listed in Table 1, and the labels 6–10 refer to their respective overheated versions.

The quality of classification is evaluated using the misclassification rate (MCR) and the confusion matrix implemented in

MATLAB 2012. The MCR measure (as a percentage) is taken from the *Statistics Toolbox*, and it accounts for the proportion of misclassified samples. The confusion matrix is calculated by the confusion function in the *Neural Network Toolbox*, and then it is plotted in a Hinton diagram.

The classification results obtained with the tested algorithms are statistically compared using 100 repetitions of n -fold CV. For problem A, 20% of the samples are selected for training, and the rest for testing. In the other case, a five-fold CV is applied, i.e., 40 samples from each class are taken for training, and 10 are taken for testing.

Several measurement scenarios are tested. In each case, we set the following parameters of the excimer laser system: the mask size, energy, and number of laser shots in each location (see Section 2.1). We selected four masks, four energy values, and five shots, which gives us 80 measurement scenarios. In the following, we use the following notation: m , mask (size); e , energy (in mJ); s , shot. For example, the scenario labeled $m12e10s1$ uses mask 12, an energy of 10 mJ, and the first shot.

For solving both classification problems, we selected the following algorithms: KNN(E) (KNN with Euclidean metrics) and KNN(C) (KNN with the cosine similarity) (Section 3.2), LDA (Section 3.3), QDA (Section 3.3), PLS-DA (Section 3.4), SIMCA (Section 3.5), NB (Section 3.6), PNN (Section 3.7), and SVM (Section 3.8). For the KNN family, we set $k = 1$. In general, classification algorithms can be applied directly to high-dimensional LIBS data or low-dimensional PCs. We analyze both cases. The former is restricted only to the selected algorithms. When PCA is not used, LDA and QDA fail owing to the singularity of the covariance matrices (as mentioned in Section 3.3). The methods, such as NB and PNN, are also intractable owing to their computational complexity when applied to high-dimensional data. Hence, we could classify the high-dimensional LIBS spectra using only PLS-DA, SVM, and KNNs. The latter case is more flexible because low-dimensional and orthogonal data are easier to handle. The above algorithms, except for SIMCA, are combined with PCA, i.e., applied to the low-dimensional PCs (see Section 3.1). SIMCA is intrinsically related with PCA; hence, there is no need to apply it to PCs. Each tested algorithm is applied in each measurement scenario and run according to the CV rule mentioned above. The statistics of the MCR of the samples is presented in various forms: box plots and cumulative results in tables, bar charts, and confusion matrices. A box plot shows the median and 25th and 75th percentiles (marked by the edges of the box), extreme data points (indicated by whiskers), and outliers.

4.1 Problem A

All the above algorithms can be used for solving problem A. We consider two cases. First, the algorithms (except for SIMCA) are combined with PCA. Table 2 lists the number of measurement scenarios that satisfy various MCR thresholds (rows) for this case. It can also be interpreted as the cumulative MCR with respect to the number of measurement scenarios for each algorithm.

The results demonstrate that SIMCA, LDA, PLS-DA, and SVM significantly outperform the other methods. These algorithms make it possible to attain an accuracy of 100 % ($MCR = 0\%$) for many measurement scenarios. There are 62 scenarios for SIMCA, 39 for LDA and PLS-DA, and 21 for SVM. We have observed that SIMCA is more resistant to fluctuations in the laser energy and inexact setting of the mask. This observation also confirms the theoretical assumption that it is suitable for the data disturbed with outliers. Its performance diminishes with increasing energy for large masks, i.e., for mask 28 when the energy exceeds 18 mJ and for mask 32 when the energy exceeds 15 mJ. The condition $MCR \leq 3\%$ is satisfied for 98.75% of the samples (see Table 2). The statistics of the MCR for LDA and PLS-DA are comparable, i.e., the same number of measurement scenarios satisfying a given MCR threshold, and nearly identical accuracy for each scenario. This observation is surprising because the algorithms orthogonalize different covariance matrices. The highest accuracy ($MCR = 0\%$) in the entire energy range is observed for mask 12. The remaining scenarios give $MCR > 0\%$. The threshold $MCR \leq 3\%$ is satisfied for 96.25% of the cases. SVM can also yield the highest accuracy, but for not as many measurement scenarios. Like LDA and PLS-DA, this algorithm works best for mask 12 in the entire energy range. Increasing the mask number and energy also lowers the performance. It satisfies the condition $MCR \leq 3\%$ in 92.5% of the samples. Independent of the measurement scenario, all four algorithms – SIMCA, LDA, PLS-DA, and SVM – give results that satisfy the threshold $MCR \leq 5\%$ (see Table 2).

The KNN algorithms, despite their simplicity, do not exhibit the worst performance. Indeed, the LDA applies KNN to the feature vectors. When the Euclidean distance was used, we could obtain $MCR = 0.18\%$ for $m12e10s4$, but only five scenarios give $MCR \leq 1\%$. When the cosine similarity was used, nine scenarios satisfy this threshold, but for one case $m12e12s4$, we could obtain $MCR = 0.17\%$. Both KNN algorithms give $MCR \leq 3\%$ for at least 30% of the scenarios.

The performance of PNN in classifying the LIBS spectra is slightly worse than the performance of KNN algorithms. The lowest $MCR = 0.69\%$ is obtained for the scenario $m12e12s5$, and it was the only result below 1%. NB and QDA give MCR values one order worse than those of the other algorithms. The best scenario for NB and QDA is $m20e18s4$, for which $MCR = 3.13\%$ and $MCR = 3.66\%$, respectively. For $MCR \in [5, 10]\%$, NB is more efficient than QDA. The comparison of the results obtained with NB and PNN shows that the feedforward neural-network implementation of the Bayes classifier seems to be more efficient.

In summary, SIMCA gives the best accuracy in the binary classification of the LIBS spectra. Owing to its high resistance to outliers, it is not so sensitive to the choice of measurement scenario. It classifies all the samples correctly in approximately 30% more scenarios compared to LDA and PLS-DA. SVM can also be used for solving problem A, especially when implemented with one extra class. The other algorithms are not recommendable for classifying the LIBS spectra in problem A.

In the experiments, we also test the algorithms with the high-

Method	MCR							
	= 0	≤ 0.01	≤ 0.1	≤ 0.5	≤ 1	≤ 3	≤ 5	≤ 10
LDA	39 (48.75)	50 (62.5)	58 (72.5)	63 (78.75)	66 (82.5)	77 (96.25)	80 (100)	80 (100)
QDA	-	-	-	-	-	-	2 (2.5)	44 (55)
SVM	21 (26.25)	37 (46.25)	54 (67.5)	63 (78.75)	66 (82.5)	74 (92.5)	80 (100)	80 (100)
NB	-	-	-	-	-	-	8 (10)	67 (83.75)
KNN(E)	-	-	-	5 (6.25)	12 (15)	24 (30)	41 (51.25)	74 (92.5)
KNN(C)	-	-	-	9 (11.25)	14 (17.5)	27 (33.75)	47 (58.75)	75 (93.75)
PLS-DA	39 (48.75)	50 (62.5)	58 (72.5)	63 (78.75)	66 (82.5)	77 (96.25)	80 (100)	80 (100)
SIMCA	62 (77.5)	62 (77.5)	66 (82.5)	69 (86.25)	70 (87.5)	79 (98.75)	80 (100)	80 (100)
PNN	-	-	-	-	1 (1.25)	17 (21.25)	38 (47.5)	74 (92.5)

TABLE 2 Number of measurement scenarios (in parentheses, percentage) that satisfy a given MCR threshold for binary classification, where the algorithms are combined with PCA.

Method	MCR							
	= 0	≤ 0.01	≤ 0.1	≤ 0.5	≤ 1	≤ 3	≤ 5	≤ 10
SVM	59 (73.75)	61 (76.25)	64 (80)	70 (87.5)	70 (87.5)	74 (92.5)	80 (100)	80 (100)
KNN(E)	-	-	1 (1.25)	6 (7.5)	13 (16.25)	25 (31.25)	43 (53.75)	74 (92.5)
PLS-DA	70 (87.5)	70 (87.5)	70 (87.5)	70 (87.5)	71 (88.75)	80 (100)	80 (100)	80 (100)

TABLE 3 Number of measurement scenarios (in parentheses, percentage) that satisfy a given MCR threshold for binary classification without using PCA.

Method	MCR						
	≤ 0.05	≤ 0.1	≤ 0.5	≤ 1	≤ 3	≤ 5	≤ 10
LDA	-	1 (1.25)	17 (21.25)	46 (57.5)	75 (93.75)	79 (98.75)	80 (100)
QDA	-	-	-	-	11 (13.75)	49 (61.25)	80 (100)
SVM	-	-	2 (2.5)	10 (12.5)	58 (72.5)	75 (93.75)	80 (100)
NB	-	-	-	-	4 (5)	46 (57.5)	80 (100)
KNN(E)	-	-	-	-	-	5 (6.25)	28 (35)
KNN(C)	-	-	-	-	-	5 (6.25)	37 (46.25)
PLS-DA	-	-	9 (11.2)	27 (33.7)	69 (86.2)	77 (96.25)	80 (100)
SIMCA	2 (2.5)	4 (5)	32 (40)	52 (65)	77 (96.2)	79 (98.75)	80 (100)
PNN	-	-	-	-	-	1 (1.25)	21 (26.25)

TABLE 4 Number of measurement scenarios (in parentheses, percentage) that satisfy a given MCR threshold for problem B (10 classes) using the algorithms combined with PCA.

dimensional LIBS data (without using PCA). In this case, we selected only three algorithms: PLS-DA, SVM, and KNN(E). As already mentioned, SIMCA intrinsically projects the input data on the local PCs; therefore, it is not considered in this test. The results of such a classification are summarized in Table 3.

We observed that if SVM and PLS-DA are not used with PCA, the number of best measurement scenarios (for which $MCR = 0\%$) is substantially higher, i.e., by 38 and 31, respectively. A particularly good result was obtained with PLS-DA, for which 70 scenarios ensure the highest accuracy. SVM is more sensitive to outliers (even with the soft margin) but also gives much better results than with PCA. The accuracy could be even better if a kernel version of SVM is applied without PCA, but the learning of the kernel on such high-dimensional data is very time-consuming. With the use of KNN(E), the improvement in the number of scenarios is not very large (only 2 for $MCR \leq 5\%$). For the best scenario $m12e10s4$, MCR diminishes from 0.18 to 0.14. Thus, all the tested algorithms without PCA offer better accuracy of binary classification but obviously at higher computational cost. It is justified by the fact that 30 PCs explain only about 90% of the total variance. The

small difference in the intensity of spectral lines is included in the unexplained variance.

4.2 Problem B

For problem B, similar measurement scenarios are tested. First, we analyze the classification algorithms combined with PCA. Figure 5 illustrates the box plots of the MCR samples obtained with each algorithm for the best measurement scenario. The confusion matrices for the best measurement scenarios are presented in Figure 6 in the form of Hinton diagrams. The titles give the lowest MCR values.

For problem B, the number of measurement scenarios that satisfy a given MCR threshold are listed in Table 4.

The results show that the tested algorithms can be assigned to four groups on the basis of their performance, and they are ordered in the decreasing order of performance as follows: (1) SIMCA; (2) LDA, SVM, and PLS-DA; (3) QDA and NB; (4) the KNN family and PNN. The groups have different sensitivity to the spectral shape. When low energy and a small mask

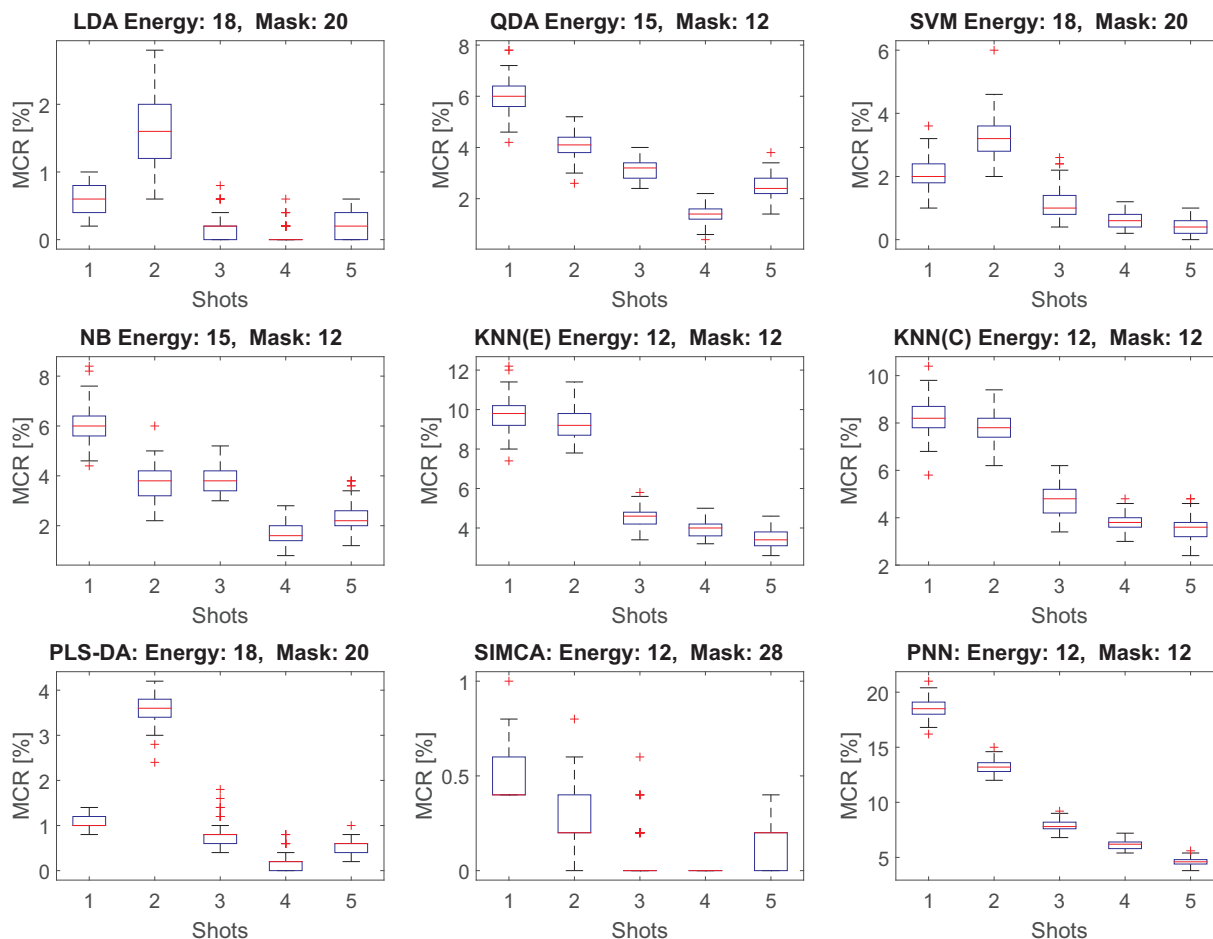


FIG. 5 MCR statistics (box plots) obtained by the classification algorithms combined with PCA and applied to problem B (10 classes) for their best measurement scenarios (given in the titles).

(particularly m12) are used, the spectrum is weakly noisy, and its emission lines have low intensity. Increasing the energy or mask size causes the intensity of emission lines and the background to increase, which may emphasize the spectrum in one or more observed channels. For all the algorithms, at least three shots should be used. The first two shots remove undesirable pollution or oxides.

The first algorithm, regarded as the individual group, is SIMCA. As in problem A, the SIMCA gives the best results for multi-label classification. It classifies one scenario (*m28e12s4*) with 100% accuracy, and $MCR \leq 0.1\%$ for 4 scenarios (see Table 4). LDA can reach this level of MCR with only one scenario (*m20e18s4*). It is the second best algorithm for solving problem B but noticeably worse than SIMCA. The condition $MCR \leq 0.5\%$ is satisfied by LDA in 17 scenarios, whereas SVM and PLS-DA satisfy it in 2 and 9 scenarios, respectively. The performance of SVM and PLS-DA is considerably worse than with LDA. PLS-DA also has some occasional problems in the classification of solder alloys with labels 8 and 9. This means that it has a lower sensitivity to the difference in concentration of Cu and Sn in the overheated solder alloys, despite one of them containing Ag. The emission lines of Ag may not contain meaningful information owing to the low concentration of this element. The problem does not occur for the solders with labels 8 and 10, as well as 9 and 10, which additionally shows that PLS-DA is the most sensitive to the difference in concentration of Cu in the overheated soft solders.

The performance of the third group is one order worse. QDA and NB allow us to obtain $MCR \leq 2\%$ in one case. NB gives slightly worse results for $MCR \leq 3\%$. In the range $MCR \in [5, 10]\%$, their performance is comparable. For both algorithms, we observed some difficulty in the classification of overheated lead-free solders that contain Sn and Cu. They are labeled 8, 9, and 10 (see Figure 6). The effect is even more noticeable when we compare solders with the labels 8 and 10, which have a very similar proportion of Sn and Cu, and there is no other relevant element. The solder labeled 9 contains the third significant element, Ag, but it is also often misclassified because of the similar proportions of Sn and Cu. Similar classification errors are observed for NB.

The fourth group is very sensitive to Ag. The KNN algorithms classify the samples with $MCR > 3.4\%$, and they cannot distinguish well the overheated solders with Ag from their healthy versions. This problem occurs for the solders *Pb70SnAg3*, labeled 2 and 7, and *Sn96.5Ag3Cu0.5*, labeled 4 and 9 (see Figure 6). The phenomenon is easier to observe for KNN(C). We also noticed some problems in the classification of the overheated leaded solders (labeled 6 and 7), despite the presence of Ag. The emission lines of Pb have considerably higher intensity than those of Ag. Hence, the latter are ignored, which results in the low classification accuracy. The KNN family is weakly sensitive to the difference between the solders in 8 and 10 (with similar proportions of Sn and Cu). However, with reference to NB and QDA, these algorithms

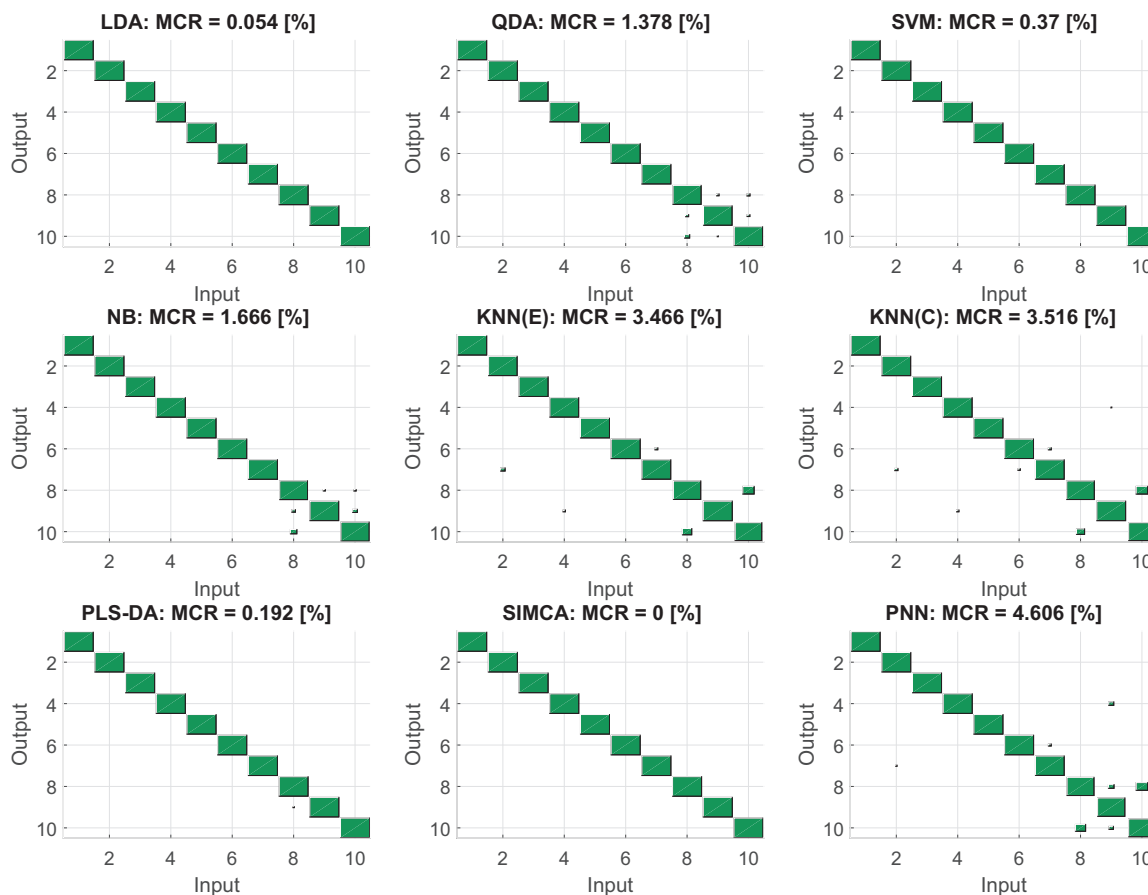


FIG. 6 Hinton diagrams of the confusion matrices for the best measurement scenario for each algorithm combined with PCA. The title of each panel gives the corresponding MCR value.

recognize the lead-free solders labeled 9 ($Sn96.5Ag3Cu0.5$) well. PNN, in spite of having similar problems as the KNN algorithms, cannot classify well the solders labeled 8, 9, and 10. Consequently, PNN shows the highest classification error ($MCR = 4.6\%$).

In summary, the condition $MCR \leq 3\%$ is satisfied by LDA, QDA, SVM, NB, PLS-DA, and SIMCA in at least one measurement scenario. All the algorithms are able to yield $MCR \leq 5\%$ for the selected parameter settings, but this condition is met by LDA, SVM, PLS-DA, and SIMCA in 98.75%, 93.75%, 96.25%, and 98.75% of all the cases, respectively.

For problem B, the changes resulting from the direct application of the classification methods to the high-dimensional LIBS data are more gentle than for problem A. There is significant correlation in the MCR statistics between the results obtained with and without the use of PCA – compare Figure 5 with Figure 7 for SVM, KNN(E), and PLS-DA. Similarly, the Hinton diagrams shown in Figures 6 and 8 appear very similar. However, the best MCR values are not the same.

Table 5 lists the results obtained for problem B without using PCA. For the multi-class classification, the difference in accuracy is less than that for problem A. The lowest MCR for SVM increases from 0.37% to 0.45%, and the best scenario changes from $m20e18s5$ to $m28e12s5$. The number of scenarios in the range $MCR \in [1,3]\%$ is comparable. For KNN(E), the minimal MCR slightly decreases from 3.47% to 3.42%. PLS-DA is rather advantageous; its lowest MCR diminishes from

0.192% for $m20e18s4$ to 0.026% for $m28e12s5$ (the same scenario as for SVM without PCA). The condition $MCR \leq 0.05\%$ is satisfied for two scenarios. The number of scenarios falling into the range $MCR \leq 0.5\%$ triples to 27. Thus, when considering the accuracy of classification, PLS-DA should not be applied to the PCs. Indeed, PLS-DA intrinsically performs model-dimensionality reduction on the observed data; hence, it need not be applied to already reduced PCs.

Figure 9 illustrates the mean MCR obtained in testing each algorithm combined with PCA versus the number of folds in n -fold CV, where $n = 2, \dots, 10$. The best measurement scenario is selected for each algorithm according to Figure 5. By increasing the number of CV folds, we observe a decrease in the MCR with a decay rate depending on the algorithm. This behavior is explained by the fact that the training set has fewer samples for smaller n . The results show that the MCR does not change significantly for $n \geq 5$. Hence, the use of five-fold CV is justified by this fact.

A similar analysis is performed for problem A. We obtain nearly the same mean MCR for each CV fold with SIMCA, LDA, PLS-DA, and SVM. For the latter, we were able to obtain satisfactory results even when the training set contains 10% of all the samples.

The classification algorithms are also compared with respect to the mean runtime in MATLAB. This includes the averaged time required for training and testing, which increases linearly with increasing number of CV folds. For $n = 2, \dots, 10$,

Method	MCR						
	≤ 0.05	≤ 0.1	≤ 0.5	≤ 1	≤ 3	≤ 5	≤ 10
SVM	–	–	2 (2.5)	12 (15)	60 (75)	75 (93.75)	80 (100)
KNN(E)	–	–	–	–	–	5 (6.25)	27 (33.75)
PLS-DA	2 (2.5)	4 (5)	27 (33.75)	45 (56.25)	75 (93.75)	78 (97.75)	80 (100)

TABLE 5 Number of measurement scenarios (in parentheses, percentage) that satisfy a given MCR threshold for problem B (10 classes) without using PCA.

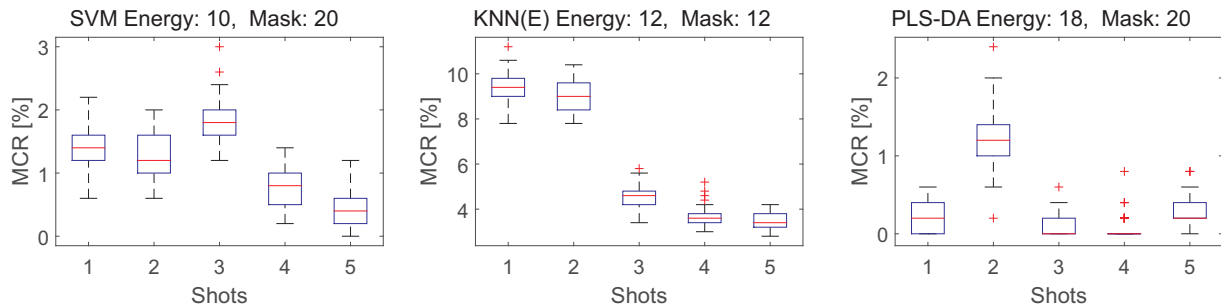


FIG. 7 MCR statistics (box plots) obtained by the classification algorithms applied to problem B (10 classes) without PCA for their best measurement scenarios (given in the titles).

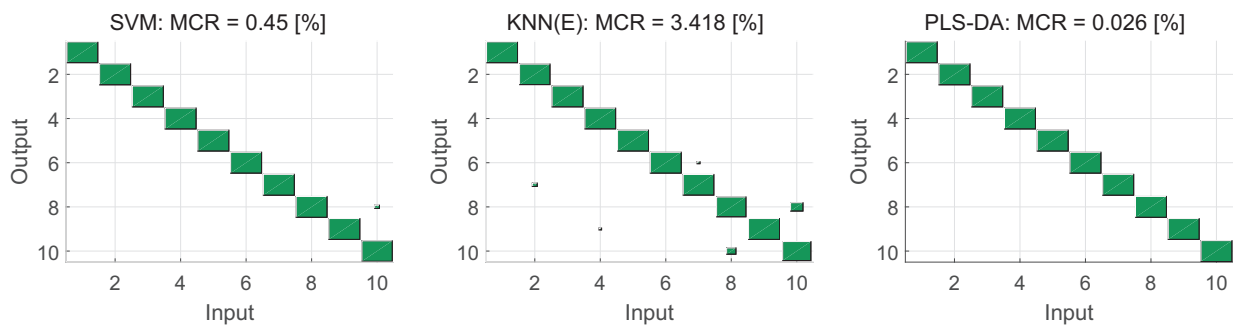


FIG. 8 Hinton diagrams of the confusion matrices for the best measurement scenario for each algorithm without PCA. The title of each panel gives the corresponding MCR value.

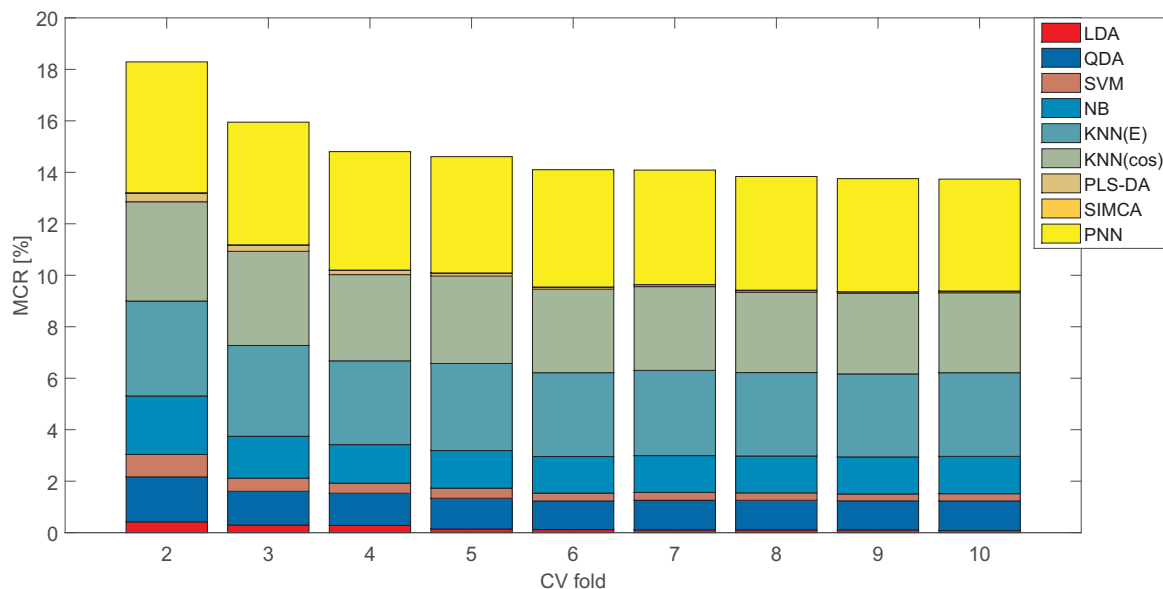


FIG. 9 Mean MCR of various classification algorithms combined with PCA in testing versus the number of CV folds.

QDA appeared to be the fastest, with a runtime in the range of 15 – 40 ms. LDA has a similar runtime of 22 – 42 ms. Among the fast algorithms, we can also include KNN(E), with a runtime of 28 – 59 ms, and KNN(C), with a runtime of 25 – 63 ms. NB and SVM are slower, with runtimes of 2.56 – 9.26 s and 3.71 – 10.97 s, respectively. SIMCA appeared to be the slowest with a runtime of 29.4 – 198.2 s.

Changing n in CV affects the MCR values and runtime. The ranges of the absolute changes and relative changes are listed in Table 6. Additionally, the last column presents the efficiency as the ratio of the relative changes (MCR to runtime).

The results demonstrate that LDA not only exhibits very good performance (the second best with respect to MCR) but also is

Algorithm	MCR [%]	RC-MCR	Time [ms]	RC-T	Efficiency
LDA	0.42-0.8	81	15-40	167	0.49
QDA	1.75-1.15	37	22-42	91	0.41
SVM	0.87-0.28	69	3709-10973	196	0.35
NB	2.27-1.45	37	2555-9256	262	0.14
KNN(E)	3.69-3.25	13	28-59	111	0.12
KNN(C)	3.86-3.11	19	25-63	152	0.13
PLS-DA	0.33-0.07	85	48-209	335	0.25
SIMCA	0.02-0	100	29400-198220 29400-125388*	574 326*	0.17 0.31*
PNN	5.08-4.35	14	335-1222	265	0.05

TABLE 6 Efficiency and performance changes (MCR, time) versus number of CV folds ($n = 2, \dots, 10$) for the tested algorithms combined with PCA. For example, the range 0.05 – 0.02 in the MCR means that the MCR values change from 0.05 to 0.02 as n changes from 2 to 10. RC-MCR is the relative change in the MCR values, and RC-T refers to the relative change in time. The efficiency is computed as the ratio of RC-MCR to RC-T.

the most efficient. A large relative change in the MCR value is also observed for SIMCA, PLS-DA, and SVM. SIMCA gives the best classification results but is not so efficient due to its computational time. This result is attributed to the fact that SIMCA applies PCA separately to each class of the training samples. For this algorithm, we noticed that the relative MCR for the folds $n = 2, \dots, 5$ is the same as for $n = 2, \dots, 10$. Hence, the case for $n = 2, \dots, 5$ is also included in Table 6 and denoted with an asterisk. The worst efficiency is observed for PNN. An increase in the RC-MCR by only 14% results in the very low level of its efficiency.

5 CONCLUSIONS

We studied computational tools for the statistical classification of solder alloys. Two classification problems were analyzed: (a) supervised separation of healthy solder samples from their overheated versions and (b) material identification. Several statistical classification tools, such as LDA, QDA, SVM, NB, KNN, PLS-DA, SIMCA, and PNN were discussed. Experiments based on LIBS observations showed that SIMCA outperforms the other algorithms for both classification problems. It yields the highest classification accuracy for the largest number of measurement scenarios. Unfortunately, it is the slowest algorithm with a runtime of dozens or even hundreds of seconds. For the first classification problem, algorithms such as LDA with PCA, SVM, PLS-DA, and SIMCA can classify the samples in many scenarios with an accuracy of 100%, where only 20% of samples are used for training. The performance of SVM, PLS-DA, and KNN(E) can be improved for a large number of the measurement scenarios if these algorithms are applied directly to high-dimensional LIBS data. i.e. without using PCA (especially SVM and PLS-DA). For the other problem, the classification error (MCR) of SIMCA does not exceed 1% for more than 60% of the samples. For one scenario, *m28e12s4*, we obtained $MCR = 0\%$. LDA and PLS-DA leads to slightly higher values of MCR and a lower number of scenarios, but their computational time is relatively short and changes from dozens of milliseconds (LDA) to hundreds of milliseconds (PLS-DA). With reference to SIMCA, they are significantly faster – by a factor of hundreds to thousands. PLS-DA should not be applied to PCs. If applied directly to high-dimensional data, its MCR is one order lower. SVM ex-

hibits slightly worse performance, and in the multiclass implementation, it is considerably slower than LDA and PLS-DA. Nevertheless, SVM has other clear advantages. In our approach, it classifies one class versus the rest. If a tested solder lies outside of any training group, it cannot be recognized by any SVM classifier. Hence, it can be assigned to an extra class. When LDA is used, it might be assigned to the most similar class (according to some metrics), which leads to incorrect interpretation. This case may occur in practice, e.g., if a laser shot reaches the substrate or the solder has a chemical composition different from that of the training samples. With the use of SVM, the testing samples that cannot be classified into any training group can be quite diverse. To find some similarity between them, we may try to cluster them using unsupervised machine-learning algorithms. For nonnegatively constrained samples such as LIBS spectra, we may use various models of nonnegative matrix factorization [80]. This approach will be analyzed in our future research.

6 ACKNOWLEDGEMENTS

The authors thank Prof. Krzysztof M. Abramski, Dr. Arkadiusz J. Antończak, and other colleagues from the Laser and Fiber Electronics Group at Wrocław University of Technology for providing the laboratory equipment necessary to conduct the research. We are also grateful to Cynel-Unipress Co. Ltd. for the preparation and measurement of the material samples/soft solder alloys.

References

- [1] K. Zeng, and T. King-Ning, "Six cases of reliability study of Pb-free solder joints in electronic packaging technology," *Mat. Sci. Eng. Rep.* **38**, 55-105 (2002).
- [2] C. W. Mak, N. V. Afzulpurkar, M. N. Dailey, and P. B. Saram, "A Bayesian approach to automated optical inspection for solder jet ball joint defects in the head gimbal assembly process," *IEEE T. Autom. Sci. Eng.* **11**, 1155-1162 (2014).
- [3] M. Koscielski, and J. Sitek, "Influence of the fluxes properties on quality and the microstructure of lead-free solder joints executed by selective soldering," *Solder. Surf. Mt. Tech.* **26**, 2-7 (2014).

- [4] Y. Wang, M. Wang, and Z. Zhang, "Microfocus X-ray printed circuit board inspection system," *Optik* **125**, 4929–4931 (2014).
- [5] M. S. Laghari, and Q. A. Memon, "Identification of faulty BGA solder joints in X-ray images," *Int. J. Fut. Comp. Commun.* **4**, 122–125 (2015).
- [6] J. C. E. Mertens, J. J. Williams, and N. Chawla, "Development of a lab-scale, high-resolution, tube-generated X-ray computed-tomography system for three-dimensional (3D) materials characterization," *Mater. Charact.* **92**, 36–48 (2014).
- [7] M. Rauer, A. Volkert, T. Schreck, S. Härter, and M. Kaloudis, "Computed-tomography-based analysis of voids in SnBi₅₇Ag₁ solder joints and their influence on the reliability," *J. Fail. Anal. Prevent.* **14**, 272–281 (2014).
- [8] S. Bord, A. Clement, J. C. Lecomte, and J. C. Marmeggi, "Investigation of IC samples using X-ray computer tomography," *J. Phys. IV* **118**, 413–417 (2004).
- [9] L. Su, Z. Zha, X. Lu, T. Shi, and G. Liao, "Using BP network for ultrasonic inspection of flip chip solder joints," *Mech. Syst. Signal Pr.* **34**, 183–190 (2013).
- [10] F. Dugal, and M. Ciappa, "Study of thermal cycling and temperature aging on PbSnAg die attach solder joints for high power modules," *Microelectron. Reliab.* **54**, 1856–1861 (2014).
- [11] B. R. Livesay, *Electronic Materials Handbook Vol 1: Packaging*, (ASM International, Metals Park, 1994).
- [12] X. Lu, T. Shi, J. Han, G. Liao, L. Su, and S. Wang, "Defects inspection of the solder bumps using self reference technology in active thermography," *Infrared Phys. Techn.* **63**, 97–102 (2014).
- [13] A. Ciucci, V. Palleschi, S. Rastelli, R. Barbini, F. Colao, R. Fantoni, A. Palucci, S. Ribezzo, and H. J. L. Van der Steen, "Trace pollutants analysis in soil by a time-resolved laser-induced breakdown spectroscopy technique," *Appl. Phys. B* **63**, 185–190 (1996).
- [14] A. Ciucci, M. Corsi, V. Palleschi, S. Rastelli, A. Salvetti, and E. Tognoni, "New procedure for quantitative elemental analysis by laser-induced plasma spectroscopy," *Appl. Spectrosc.* **53**, 960–964 (1999).
- [15] E. H. Piepmeier, *Laser ablation for atomic spectroscopy* (John Wiley & Sons, New York, 1986).
- [16] G. C. Dacey, "Optical masers in science and technology," *Science* **135**, 71–74 (1962).
- [17] F. Brech, and L. Cross, "Optical micro-emission stimulated by a ruby MASER," *Appl. Spectrosc.* **16**, 59 (1962).
- [18] E. F. Runge, R. W. Minck, and F. R. Bryan, "Spectrochemical analysis using a pulsed laser source," *Spectrochim. Acta* **20**, 733–736 (1964).
- [19] L. Radziemski, and D. Cremers, "A brief history of laser-induced breakdown spectroscopy: from the concept of atoms to LIBS 2012," *Spectrochim. Acta B* **87**, 3–10 (2013).
- [20] C. Aragon, and J. A. Aguilera, "Characterization of laser induced plasmas by optical emission spectroscopy: a review of experiments and methods," *Spectrochim. Acta B* **63**, 893–916 (2008).
- [21] C. Pasquini, J. Cortez, L. M. C. Silva, and F. B. Gonzaga, "Laser induced breakdown spectroscopy," *J. Brazil. Chem. Soc.* **18**, 463–512 (2007).
- [22] D. W. Hahn, and N. Omenetto, "Laser-induced breakdown spectroscopy (LIBS), part I: review of basic diagnostics and plasma-particle interactions: still-challenging issues within the analytical plasma community," *Appl. Spectrosc.* **64**, 335A–366A (2010).
- [23] D. W. Hahn, and N. Omenetto, "Laser-induced breakdown spectroscopy (LIBS), part II: review of instrumental and methodological approaches to material analysis and applications to different fields," *Appl. Spectrosc.* **66**, 100A–119A (2012).
- [24] F. Anabitarte, A. Cobo, and J. M. Lopez-Higuera, "Laser-induced breakdown spectroscopy: fundamentals, applications, and challenges," *ISRN Spectrosc.* **2012**, 1–12 (2012).
- [25] J. El Haddad, L. Canioni, and B. Bousquet, "Good practices in LIBS analysis: review and advices," *Spectrochim. Acta B* **101**, 171–182 (2014).
- [26] D. A. Cremers, and L. J. Radziemski (eds.), *Handbook of laser-induced breakdown spectroscopy* (John Wiley & Sons, Chichester, 2013).
- [27] S. Musazzi, and U. Perini (eds.), *Laser-induced breakdown spectroscopy: theory and applications* (Springer, New York, 2014).
- [28] J. P. Singh, and S. N. Thakur, *Laser-induced breakdown spectroscopy* (Elsevier, Amsterdam, 2007).
- [29] D. A. Cremers, F. Y. Yueh, J. P. Singh, and H. Zhang, *Laser-induced breakdown spectroscopy, Elemental analysis*, Encyclopedia of Analytical Chemistry, 9780470027318.a0708.pub2 (2012).
- [30] A. W. Miziolek, V. Palleschi, and I. Schechter, *Laser-induced breakdown spectroscopy* (Cambridge University Press, Cambridge, 2006).
- [31] C. M. Bishop, *Pattern recognition and machine learning* (Springer, New York, 2007).
- [32] D. Barber, *Bayesian reasoning and machine learning* (Cambridge University Press, New York, 2012).
- [33] V. K. Unnikrishnan, K. S. Choudhari, S. D. Kulkarni, R. Nayak, V. B. Kartha, C. Santhosh, and B. M. Suri, "Biomedical and environmental applications of laser-induced breakdown spectroscopy," *Pramana* **82**, 397–401 (2014).
- [34] M. Á. Aguirre, M. Hidalgo, A. Canals, J. A. Nóbrega, and E. R. Pereira-Filho, "Analysis of waste electrical and electronic equipment (WEEE) using laser-induced breakdown spectroscopy (LIBS) and multivariate analysis," *Talanta* **117**, 419–424 (2013).
- [35] H. Xia, and M. C. M. Bakker, "Reliable classification of moving waste materials with LIBS in concrete recycling," *Talanta* **120**, 239–247 (2014).
- [36] J. L. Gottfried, R. S. Harmon, F. C. D. L. Jr., and A. W. Miziolek, "Multivariate analysis of laser-induced breakdown spectroscopy chemical signatures for geomaterial classification," *Spectrochim. Acta B* **64**, 1009–1019 (2009).
- [37] G. Kim, J. Kwak, K. R. Kim, H. Lee, K. W. Kim, H. Yang, and K. Park, "Rapid detection of soils contaminated with heavy metals and oils by laser-induced breakdown spectroscopy (LIBS)," *J. Hazard. Mater.* **263**, 754–760 (2013).
- [38] X. Zhu, T. Xu, Q. Lin, L. Liang, G. Niu, H. Lai, M. Xu, et al., "Advanced statistical analysis of laser-induced breakdown spectroscopy data to discriminate sedimentary rocks based on Czerny-Turner and Echelle spectrometers," *Spectrochimica Acta B* **93**, 8–13 (2014).
- [39] R. C. Wiens, S. Maurice, J. Lasue, O. Forni, R. B. Anderson, S. Clegg, S. Bender, et al., "Pre-flight calibration and initial data processing for the ChemCam laser-induced breakdown spectroscopy instrument on the Mars Science Laboratory rover," *Spectrochimica Acta B* **82**, 1–27 (2013).
- [40] Y. Shunchun, L. Jindong, D. Meirong, C. Kai, L. Junyan, and L. Jun, "Extracting coal ash content from laser-induced breakdown spectroscopy (LIBS) spectra by multivariate analysis," *Appl. Spectrosc.* **65**, 1197–1201 (2011).

- [41] M. M. Tripathi, K. K. Srinivasan, S. R. Krishnan, F. Y. Yueh, and J. P. Singh, "A comparison of multivariate LIBS and chemiluminescence-based local equivalence ratio measurements in premixed atmospheric methane-air flames," *Fuel* **106**, 318–326 (2013).
- [42] J. El Haddad, D. Bruyere, A. Ismaël, G. Gallou, V. Laperche, K. Michel, L. Canioni, et al., "Application of a series of artificial neural networks to on-site quantitative analysis of lead into real soil samples by laser-induced breakdown spectroscopy," *Spectrochim. Acta B* **97**, 57–64 (2014).
- [43] P. M. Mukhono, K. H. Angeyo, A. Dehayem-Kamadjeu, and K. A. Kaduki, "Laser-induced breakdown spectroscopy and characterization of environmental matrices utilizing multivariate chemometrics," *Spectrochim. Acta B* **87**, 81–85 (2013).
- [44] G. S. Senesi, "Laser-induced Breakdown Spectroscopy (LIBS) applied to terrestrial and extraterrestrial analogue geomaterials with emphasis to minerals and rocks," *Earth-Sci. Rev.* **139**, 231–267 (2014).
- [45] M. Sinmaz, B. G. Oztoprak, E. Akman, M. Gunes, E. Kacar, F. Tulek, and A. Demir, "Classification of archeologic ceramics analysed by Laser-induced Breakdown Spectroscopy (LIBS)," *Balk. Phys. Lett.* **22**, 24–35 (2014).
- [46] A. Ramil, A. J. López, and A. Yáñez, "Application of artificial neural networks for the rapid classification of archaeological ceramics by means of laser-induced breakdown spectroscopy (LIBS)," *Appl. Phys. A* **92**, 197–202 (2008).
- [47] G. Vítková, L. Prokeš, K. Novotný, P. Pořízka, J. Novotný, D. Všíanský, L. Čelko, et al., "Comparative study on fast classification of brick samples by combination of principal component analysis and linear discriminant analysis using stand-off and table-top laser-induced breakdown spectroscopy," *Spectrochim. Acta B* **101**, 191–199 (2014).
- [48] R. Kanawade, F. Mehari, C. Knipfer, M. Rohde, K. Tangermann-Gerk, M. Schmidt, and F. Stelzle, "Pilot study of laser-induced breakdown spectroscopy for tissue differentiation by monitoring the plume created during laser surgery—An approach on a feedback Laser control mechanism," *Spectrochim. Acta B* **87**, 175–181 (2013).
- [49] S. Moncayo, S. Manzoor, F. Navarro-Villoslada, and J. O. Caceres, "Evaluation of supervised chemometric methods for sample classification by laser-induced breakdown spectroscopy," *Chemometr. Intell. Lab.* **146**, 354–364 (2015).
- [50] Q. Godoi, F. O. Leme, L. C. Trevizan, E. R. Pereira Filho, I. A. Rufini, D. Santos, and F. J. Krug, "Laser-induced breakdown spectroscopy and chemometrics for classification of toys relying on toxic elements," *Spectrochim. Acta B* **66**, 138–143 (2011).
- [51] J. Cisewski, E. Snyder, J. Hannig, and L. Oudejans, "Support vector machine classification of suspect powders using laser-induced breakdown spectroscopy (LIBS) spectral data," *J. Chemometr.* **26**, 143–149 (2012).
- [52] L. Liang, T. Zhang, K. Wang, H. Tang, X. Yang, X. Zhu, Y. Duan, et al., "Classification of steel materials by laser-induced breakdown spectroscopy coupled with support vector machines," *Appl. Opt.* **53**, 544–552 (2014).
- [53] M. Grzegorzec, D. Schwerbel, D. Balthasar, and D. Paulus, "Automatic sorting of aluminium alloys based on spectroscopy measures," in *ÖAGM/AAPR Workshop 2011* (Graz, Austria, 2011).
- [54] T. Canel, P. Demir, E. Kacar, B. G. Oztoprak, E. Akman, M. Gunes, and A. Demir, "Optimization of parameters for depth resolution of galvanized steel by LIBS technique," *Opt. Laser Technol.* **54**, 257–264 (2013).
- [55] A. M. El Sherbini, and A. A. S. Al Aamer, "Development of a simple software program used for evaluation of plasma electron density in LIBS experiments via spectral line shape analysis," *J. Signal Inf. Process.* **3**, 502–515 (2012).
- [56] I. T. Jolliffe, *Principal component analysis* (second edition, Springer, New York, 2002).
- [57] J. D. Hybl, G. A. Lithgow, and S. G. Buckley, "Laser-induced breakdown spectroscopy detection and classification of biological aerosols," *Appl. Spectrosc.* **57**, 1207–1215 (2003).
- [58] G. H. Golub, and C. F. V. Loan, *Matrix computations* (The Johns Hopkins University Press, Baltimore, London, 1996).
- [59] S. Dray, "On the number of principal components: a test of dimensionality based on measurements of similarity between matrices," *Comput. Stat. Data An.* **52**, 2228–2237 (2008).
- [60] J. Josse, and F. Husson, "Selecting the number of components in principal component analysis using cross-validation approximations," *Comput. Stat. Data An.* **56**, 1869–1879 (2012).
- [61] M. O. Ulfarsson, and V. Solo, "Selecting the number of principal components with SURE," *IEEE Signal Proc. Lett.* **22**, 239–243 (2015).
- [62] H. Wold, "Estimation of principal components and related models by iterative least squares," in *Multivariate Analysis*, P. Krishnaiah, ed., 391–420 (Academic Press, New York, 1966).
- [63] J.-B. Sirven, B. Bousquet, L. Canioni, and L. Sarger, "Laser-induced breakdown spectroscopy of composite samples: comparison of advanced chemometrics methods," *Anal. Chem.* **78**, 1462–1469 (2006).
- [64] F.-Y. Yueh, H. Zheng, J. P. Singh, and S. Burgess, "Preliminary evaluation of laser-induced breakdown spectroscopy for tissue classification," *Spectrochim. Acta B* **64**, 1059–1067 (2009).
- [65] J. M. Andrade, G. Cristoforetti, S. Legnaioli, G. Lorenzetti, V. Palleschi, and A. A. Shaltout, "Classical univariate calibration and partial least squares for quantitative analysis of brass samples by laser-induced breakdown spectroscopy," *Spectrochim. Acta B* **65**, 658–663 (2010).
- [66] S. Yao, J. Lu, J. Li, K. Chen, J. Li, and M. Dong, "Multi-elemental analysis of fertilizer using laser-induced breakdown spectroscopy coupled with partial least squares regression," *J. Anal. Atom. Spectrom.* **25**, 1733–1738 (2010).
- [67] R. A. Putnam, Q. I. Mohaidat, A. Daabous, and S. J. Rehse, "A comparison of multivariate analysis techniques and variable selection strategies in a laser-induced breakdown spectroscopy bacterial classification," *Spectrochim. Acta B* **87**, 161–167 (2013).
- [68] B. Yunfeng, Z. Ying, Y. Jingwen, W. Zhongchen, and L. Ying, "Classification and discrimination of minerals using laser induced breakdown spectroscopy and Raman spectroscopy," *Plasma Sci. Technol.* **17**, 923 (2015).
- [69] S. de Jong, "SIMPLS: an alternative approach to partial least squares regression," *Chemometr. Intell. Lab.* **18**, 251–263 (1993).
- [70] S. Wold, "Pattern recognition by means of disjoint principal components models," *Pattern Recogn.* **8**, 127–139 (1976).
- [71] S. Wold, and M. Sjostrom, "SIMCA: a method for analyzing chemical data in terms of similarity and analogy," in *Chemometrics: theory and application*, B. R. Kowalski, ed., 243–281 (American Chemical Society, Washington, D. C., 1977).
- [72] S. M. Clegg, E. Sklute, M. D. Dyar, J. E. Barefield, and R. C. Wiens, "Multivariate analysis of remote laser-induced breakdown spectroscopy spectra using partial least squares, principal component analysis, and related techniques," *Spectrochim. Acta B* **64**, 79–88 (2009).

- [73] D. F. Specht, "Probabilistic neural networks," *Neural Networks* **3**, 109-118 (1990).
- [74] V. Motto-Ros, A. Koujelev, G. Osinski, and A. Dudelzak, "Quantitative multi-elemental laser-induced breakdown spectroscopy using artificial neural networks," *J. Eur. Opt. Soc.-Rapid* **3**, 08011 (2008).
- [75] J. O. Caceres, S. Moncayo, J. D. Rosales, F. J. M. de Villena, F. C. Alvira, and G. M. Bilmes, "Application of laser-induced breakdown spectroscopy (LIBS) and neural networks to olive oils analysis," *Appl. Spectrosc.* **67**, 1064-1072 (2013).
- [76] D. Pokrajac, A. Lazarevic, V. Kecman, A. Marcano, Y. Markushin, T. Vance, N. Reljin, et al., "Automatic classification of laser-induced breakdown spectroscopy (LIBS) data of protein biomarker solutions," *Appl. Spectrosc.* **68**, 1067-1075 (2014).
- [77] T. Zhang, S. Wu, J. Dong, J. Wei, K. Wang, H. Tang, X. Yang, et al., "Quantitative and classification analysis of slag samples by laser induced breakdown spectroscopy (LIBS) coupled with support vector machine (SVM) and partial least square (PLS) methods," *J. Anal. Atom. Spectrom.* **30**, 368-374 (2015).
- [78] L. Sheng, T. Zhang, G. Niu, K. Wang, H. Tang, Y. Duan, and H. Li, "Classification of iron ores by laser-induced breakdown spectroscopy (LIBS) combined with random forest (RF)," *J. Anal. Atom. Spectrom.* **30**, 453-458 (2015).
- [79] C. Cortes, and V. Vapnik, "Support-vector networks," *Machine Learning* **20**, 273-297 (1995).
- [80] A. Cichocki, R. Zdunek, A. H. Phan, and S.-I. Amari, *Nonnegative matrix and tensor factorizations: applications to exploratory multi-way data analysis and blind source separation* (John Wiley & Sons, Chichester, 2009).