Analytica Chimica Acta 1119 (2020) 41–51

Contents lists available at ScienceDirect

Analytica Chimica Acta

journal homepage: www.elsevier.com/locate/aca

Understanding the learning mechanism of convolutional neural networks in spectral analysis

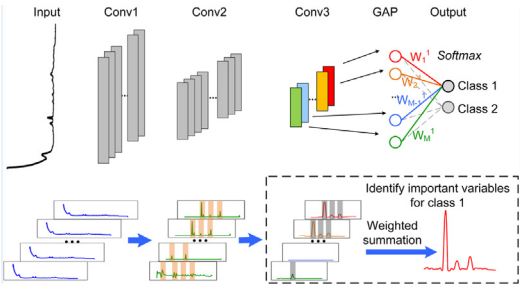
Xiaolei Zhang [a, b, 1], Jinfan Xu [a, b, 1], Jie Yang [a, b], Li Chen [c], Haibo Zhou [d], Xiangjiang Liu [a, b], Haifeng Li [c], Tao Lin [a, b, *], Yibin Ying [a, b, e]

[a] College of Biosystems Engineering and Food Science, Zhejiang University, Hangzhou, Zhejiang, 310058, China
[b] Key Laboratory of on Site Processing Equipment for Agricultural Products, Ministry of Agriculture and Rural Affairs, China
[c] School of Geosciences and Info-Physics, Central South University, South Lushan Road, Changsha, 410000, China
[d] Institute of Pharmaceutical Analysis and Guangdong Province Key Laboratory of Pharmacodynamic Constituents of Traditional Chinese Medicine & New Drug Research, College of Pharmacy, Jinan University, Guangzhou 510632, China
[e] Faculty of Agricultural and Food Science, Zhejiang A&F University, Hangzhou, Zhejiang, 311300, China

HIGHLIGHTS

- Identify active variables learned by a CNN model for spectral analysis.
- Illustrate the feature representations via different convolutional layers.
- Evaluate the model reliability with Monte-Carlo cross-validation.

GRAPHICAL ABSTRACT



ARTICLE INFO

Article history:
Received 11 November 2019
Received in revised form
27 February 2020
Accepted 29 March 2020
Available online 8 April 2020

Keywords:
Feature visualization
Interpretation
Deep learning
Reliability
Class activation mapping

ABSTRACT

Deep learning approaches, especially convolutional neural network (CNN) models, have achieved excellent performances in vibrational spectral analysis. The critical drawback of the CNN approach is the lack of interpretation, and it is regarded as a black box. Interpreting the learning mechanism of chemometric models is critical for intuitive understanding and further application. In this study, an interpretable CNN model with a global average pooling layer is presented for Raman and mid-infrared spectral data analysis. A class activation mapping (CAM)-based approach is leveraged to visualize the active variables in the whole spectrum. The visualization of active variables shows a discriminative pattern in which the most contributed variables peaked around theoretical chemical characteristic bands. The visualization of the feature maps by three convolutional layers demonstrates the data transformation pipeline and how the CNN model hierarchically extracts informative spectral features. The first layer acts as a Savitzky-Golay filter and learns spectral shape characteristics, while the second layer learns enhanced patterns from typical spectral peaks on a few correlated variables. The third layer shows stable activations on critical spectral peaks. A partial least squares - linear discriminant analysis (PLS-LDA) model is presented for comparison on classification accuracy and model interpretation. The CNN model yields mean classification accuracies of 99.01 and 100% for *E. coli* and meat datasets on the test set, while the PLS-LDA models obtain accuracies of 98.83 and 100%. Both the CNN and PLS-LDA

* Corresponding author. College of Biosystems Engineering and Food Science, Zhejiang University, Hangzhou, Zhejiang, 310058, China.
E-mail address: lintao1@zju.edu.cn (T. Lin).
[1] X.Z. and J.X. contributed equally to this work.

https://doi.org/10.1016/j.aca.2020.03.055

models demonstrate stable patterns on active variables while CNN models are more stable than PLS-LDA models on classification performances for various dataset partitions with Monte-Carlo cross-validation.

1. Introduction

The development of the deep learning approach creates great opportunities in the field of chemometrics. Convolutional neural network (CNN) approach is a state-of-the-art technique with remarkable performance on infrared and Raman spectral analysis [1–5]. Existing CNN models with different architectures provide improved performance by reducing the need for preprocessing and variable selection [3,6]. The end-to-end CNN approach with local connections and shared weights extracts local features efficiently from a full spectrum. Given the multilayer operations and complex nonlinear connections, neural network-based models have been criticized as a black box without much interpretation [7,8]. Interpretability is a critical criterion for further practical application of a CNN model, which is as important as providing high performance in vibrational spectroscopic data analysis [2]. A proper explanation approach, therefore, is desired to fully understand the internal representation of deep CNN models with stacked nonlinear operations.

Compared with traditional spectral analysis approaches, there still lack mature methods for explaining deep neural networks. A trade-off exists between predictive accuracy and interpretability [9,10]. Linear models, such as multilinear regression (MLR), principal component regression (PCR), or partial least squares (PLS), can be easily interpreted by their coefficients but sometimes achieve lower accuracies. Nonlinear models with complex structures are usually hard to quantify the relevance between variables and the result. Variable selection is another approach to provide a simple option for model interpretation by selecting a subset of most relevant variables, or removing uninformative variables [11]. Many variable selection algorithms, such as interval partial least squares (iPLS) [12], genetic algorithms [13], and competitive adaptive reweighted sampling (CARS) have been proposed to select important variables. It is, however, criticized for the low reproducibility, the risk of overfitting, and the loss of informative variables [14,15]. Explaining the learning mechanism of a CNN model, therefore, requires an advanced visualization approach to present feature representations in the model.

Recent studies have conducted preliminary investigations on interpreting deep learning models by discovering the most active variables of the whole spectrum [2,4,16]. A previous study identified the important variable regions by retraining the last convolutional layer and selecting features with positive coefficients [2]. Other studies visualized the five most active filters in the first and second convolutional layers [16], or computed the gradient of the output with respect to the input spectrum to obtain the regression coefficients for important variable visualization [4]. Another study indicated the importance of the variables by replacing a few variables in the spectrum and assessing the variance for the prediction [17]. The above methods were devoted to finding the critical variables while lacking an explanation of how a CNN model learns spectral features layer-by-layer. An explanation of inside the learning mechanism by a deep neural network therefore is desired for a comprehensive understanding of why the model produces a certain output. A previous study illustrated what a CNN model learns by presenting the feature maps of convolutional layers [18]. This work explains the preprocessing effect of convolutional layers but lacks an illustration of the important variables for spectral analysis. The class activation mapping (CAM) approach can localize discriminative image regions in image analysis by the weighted sum of the feature maps in the last convolutional layer [19]. It provides remarkable localization ability on image analysis for popular AlexNet [20], VGGNet [21], and GoogLeNet [21] models, and can be further used in spectral analysis. In this study, we provide an approach to identify each variable's influence on the final classification and illustrate how a CNN model works on spectral data by visualizing the feature maps in convolutional layers.

The objectives of this study are: (1) develop an explanation approach for CNN models to visualize the active variables in a whole spectrum based on the CAM method, (2) interpret the internal feature representations and data transformation pipeline of a CNN model by visualizing feature maps in the convolutional layers, and (3) evaluate the model reliability by Monte-Carlo cross-validation under five dataset partitions.

2. Materials and methods

In this section, we first introduce the convolutional neural network (CNN) model with the global average pooling (GAP) layer employed in this study (section 2.1). Next, we describe how to identify critical variables with a CAM-based approach in section 2.2. The feature representations within convolutional layers are provided in section 2.3, followed by experimental design in section 2.4.

2.1. Model architecture

A one-dimensional CNN model is developed for spectral analysis. The model adopts three stacked convolutional layers (labeled as Conv1, Conv2, and Conv3), one global average pooling layer (labeled as GAP, Fig. A1), and one output layer (Fig. 1). CNN models with two or three convolutional layers are commonly used according to previous studies [3,5,22]. The first convolutional layer extracts relevant data patterns from noisy raw spectra. The subsequent layers learn the more complex abstractions of these patterns [10]. With an increase in convolutional layers, the model improves representation capabilities with an increased number of abstractions of these patterns. The number of filters, therefore, increases in deeper convolutional layers. In this study, the number of filters in the three convolutional layers is 16, 32, and 64. A certain number of filters in the convolutional layer enables sufficient representation capabilities for the model. Filter size and kernel strides in each layer are optimized according to input variable numbers, spectral resolutions, and the number of convolutional layers (Table A1).

The model leverages a unique architecture that replaces the fully connected layer with a GAP layer. The utilization of the GAP layer, a structural regularizer, reduces the number of parameters, therefore, prevents overfitting [23]. It was applied to the milestone GoogLeNet and ResNet models to achieve excellent performances [24,25]. The GAP layer provides a downsampling operation, which takes the average value of the feature maps resulting from the previous convolutional layer [23]. The output of the GAP layer (F_i) is calculated in Eq. (1).

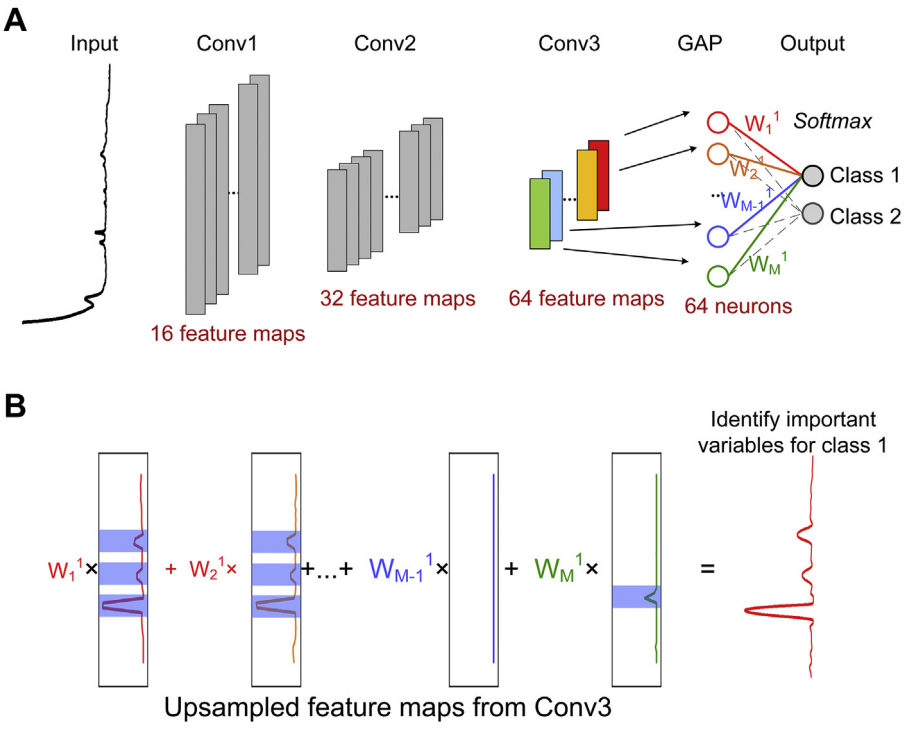


Fig. 1. CNN model architecture (A) and class activation mapping approach (B). The rectangles represent feature maps after convolutional operations, while the circles represent neurons in the fully connected layer. The W_i^c represents the ith weight for class c in the fully connected layer. The M equals to 64 in this study, which represents the total number of neurons in the GAP layer. The solid line represents the weight related to class 1, while the dotted line represents the weight related to class 2.

$$F_i = \frac{1}{N}\sum_{x=1}^{N} f_i(x) \tag{1}$$

where $f_i(x)$ is the value at location x in the ith feature map of Conv3 and N is the length of the feature map.

The output in the GAP layer F_i is multiplied by a corresponding weight w_i^c to obtain a summation output S_c before fed into a *softmax* classifier (Eq. (2)).

$$S_c = \sum_{i=1}^{M} F_i \times w_i^c \tag{2}$$

where M represents the total number of feature maps in layer Conv3. w_i^c represents the ith weight of a specific class c. C is the number of categories in the spectrum.

The S_c might be positive or negative. A *softmax* classifier is used for normalizing the S_c to the interval (0,1) with a summation of 1. The output values, therefore, represent the probability of the spectra over predicted classes (Eq. (3)).

$$\sigma(S_c) = \frac{e^{S_c}}{\sum_{j=1}^{C} e^{S_j}} \text{ for } c = 1, \ldots, C \text{ and } \boldsymbol{S} = (S_1, \ldots, S_c) \in \mathbb{R}^C \tag{3}$$

The objective function for this CNN model is the cross-entropy error loss with *L2* regularization, as shown in Eq. (4). The *L2* regularization is adopted for avoiding overfitting.

$$loss = -\frac{1}{N}\sum_{n=1}^{N}\left[y_n log\hat{y}_n + (1-y_n)\log\left(1-\hat{y}_n\right)\right] + \lambda||w||^2 \tag{4}$$

2.2. Identify active variables

The importance of each variable is computed by weighted feature maps of the last convolutional layer (Fig. 1). As illustrated in section 2.1, the class of a spectrum is determined by the value of S_c, which is the sum of the products by the output in the GAP layer (F_i) and the corresponding weights (w_i). To be specific, the output of the GAP layer is an average value of the feature map in layer Conv3. The S_c can therefore be rewritten as the product summation of the weights and feature maps in Conv3 (Eq. (5)). Besides, the feature maps in layer Conv3 can be back-transformed to the input spectral space by an upsampling approach named spline interpolation with order 2. Different quadratic curves are leveraged to fit every two adjacent points with the same slop at the interior data points to acquire a curve to fit all the datapoints. An upsampled vector P_i is thus obtained by the curve, which has same dimension with the raw spectrum and indicates the activation at each variable for a whole spectrum. Various feature maps in Conv3, however, focus on diverse regions of a spectrum, resulting from different filters. A product summation of the upsampled P_i with the corresponding weights thus represents the importance of each variable (Fig. 1). The importance of variable x is defined as $M_c(x)$ for a spectrum of class c, which can be calculated in Eq. (6). A higher value indicates the increased importance of the corresponding variable. As the data patterns extracted by the third convolutional layer are a combination of features from the previous two layers, the active variables therefore are obtained based on the whole model.

$$S_c = \sum_{i=1}^{M} F_i \times w_i^c = \frac{1}{N}\sum_{x=1}^{N} f_i(x) \times \sum_{i=1}^{M} w_i^c = \frac{1}{N}\sum_{x=1}^{N}\sum_{i=1}^{M} w_i^c \times f_i(x) \tag{5}$$

$$M_c(x) = \sum_{i=1}^{M} w_i^c \times P_i(x) \quad (6)$$

The class-specific pattern of active variables is generated by all of the correctly classified spectra. The variable importance is calculated by the feature maps in Conv3 and weights in the output layer. The weights are fixed values learned by the model, while the feature maps are results of the input spectrum and convolutional kernels. A pre-trained model has fixed weights and convolutional kernels. The difference for different samples therefore comes from the change in input spectrum. As individual differences exist, the variable importance ($M_c(x)$) is normalized to a range of 0–1 by min-max scaling before visualization. The min-max scaling approach does not change the relative peaks of the CAM, which provides a fair comparison for different spectra regardless the individual differences. The differences still exist even after normalization for different samples. An average value of M_c for different spectra with standard deviation error thus presents a class-specific pattern of important variables. The top ten percent of the most active variables for each sample are selected to illustrate important variable regions. Relative frequency distribution of sample numbers is leveraged to present the percentage of samples for each critical variable.

2.3. Feature representations within convolutional layers

The feature maps by three convolutional layers are visualized to present how a CNN model extracts intricate features hierarchically and illustrate the data transformation through the model. In the first convolutional layer, the input spectrum is transformed into 16 feature maps by a sparse connection with 16 filters. The 16 feature maps correspond 16 kinds of patterns learned by the filters. In layer Conv2, 32 filters are adopted for convolution operation with the 16 feature maps generated in layer Conv1. After the second and third convolutional layer, 32 feature maps are obtained in layer Conv2 while there are 64 feature maps in layer Conv3. In these three convolutional layers, a rectified linear unit (ReLU) activation function is leveraged after convolution (Eq. (7)), which results in sparse representations by setting a negative value to zero. The CNN model therefore enhances the informative variable regions and weakens the uninformative variables step by step with the increased number of convolutional layers. The visualization of the feature maps illustrates how a CNN model works on an input spectrum for feature extraction. In this study, the feature maps are generated by the average spectrum of each class, including *E. coli* 498, *E. coli* 1116, chicken, turkey and pork.

$$f(x) = max(0, x) \quad (7)$$

2.4. Experimental design

In this study, a CAM-based approach is provided to measure the importance of the variables in a whole spectrum for a CNN model. The active variables learned by the CNN model are further compared with theoretical spectral peaks for evaluating the reliability of the model. Feature maps by three convolutional layers are presented to illustrate the layer-by-layer feature extraction mechanisms. Furthermore, the model sensitivity and reliability are evaluated by Monte-Carlo cross-validation with five random dataset partitions. Considering random factors in the optimization procedure, each CNN model is run for ten times with the same hyperparameters.

Both classification performance and interpretability of the CNN model are compared with the currently popular modeling approach, the partial least squares - linear discriminant analysis (PLS-LDA) [26] model. PLS-LDA model has been widely used in chemometrics for its accessible interpretation to present important variables in spectral analysis. In this study, PLS-LDA models are built using the same dataset partitions as CNN models for a fair comparison. The number of latent variables (LVs) is optimized from 2 to 20, according to the performance of the training and validation set. The best number of LVs is selected by the lowest summation of RMSEC and RMSEV. A variable importance in projection (VIP) approach is adopted for measuring the accumulative influence of each variable on the model [27]. The VIP score for the *j*th variable is defined in Eq. (8) suggested by a previous study [27]. The spectral variable with VIP score larger than 1 is the most relevant to the model, and variables with VIP scores lower than 0.5 are considered as irrelevant variables [28],

$$V_j = \sqrt{p \frac{\sum_{a=1}^{A} \left[q_a^2 t_a' t_a \left(\frac{w_{aj}}{||w_a||^2} \right) \right]}{\sum_a^A (q_a^2 t_a' t_a)}} \quad (8)$$

where the p is the total number of variables, A represents the total number of components, q_a and t_a are the y loading and x score vectors of ath component, respectively. w_{aj} is loading weights, which demonstrates the importance of the jth variable in each ath component.

The proposed explanation approach is tested by two datasets, including one mid-infrared spectral dataset (meat) and one Raman spectral dataset (*Escherichia coli*, *E. coli*). To evaluate the model performance, the dataset is randomly divided into training, validation, and test sets by a ratio of 7:1:2 with evenly distributed classes. The training, validation, and test samples for the *E. coli* dataset are 597, 85, and 171, while the meat dataset are 84, 12, and 24, respectively (Table 1). The training set is used for training and updating the model parameters, while the validation set is used to optimize the hyperparameters. For the PLS-LDA model, the baseline correction should be conducted for the *E. coli* data as an offset exists. The first derivative of the Savitzky-Golay (SG) filter is leveraged as a detrending method for the *E. coli* dataset. The meat dataset is less noisy and no preprocessing approach is used for the PLS-LDA model. Detrending methods are not adopted for CNN models as convolutional layers act as smoothing and detrending [2,4]. The min-max scaling is leveraged to suppress the effect of outliers by scaling the spectral data to 0–1 with small standard deviations, which ensure that all variables are equally represented in magnitude. Meat spectral data are min-max scaled by different spectra (Eq. (9), while the *E. coli* spectral data are normalized by different spectra and variables due to a high baseline shift (Eq. (10).

$$z_{ij} = \frac{x_{ij} - min(x_j)}{max(x_j) - min(x_j)} \quad (9)$$

$$z_{ij} = \frac{x_{ij} - min(x)}{max(x) - min(x)} \quad (10)$$

where z_{ij} represents the value for the jth variable of the ith spectrum after scaling, and the x_{ij} is the raw value of the jth variable for the ith spectrum.

The first dataset is the *E. coli* dataset, which includes 853 samples with *E. coli* 498 and *E. coli* 1116 (Fig. A2). The spectra cover a region of 50–2000 cm^{-1}. The typical Raman shifts (fingerprint information) from the cell wall are at 624, 652, 735, 955, 1330, and

Table 1
Datasets description.

Datasets	Total samples	Training samples	Validation samples	Test samples	Category number	Feature number
E. coli (Raman)	853	597	85	171	2	1733
Meat (MIR)	120	84	12	24	3	448

Table 2
Classification accuracies for the CNN model and PLS-LDA model. The average classification accuracies with standard deviation errors for two datasets with ten repeated runs are presented for the CNN model.

Datasets	Training set	Validation set	Test set
E. coli (Raman)			
CNN (mean ± std)	99.48% ± 0.09%	98.82% ± 0.62%	99.01% ± 0.54%
PLS-LDA	99.50%	97.65%	98.83%
Meat (MIR)			
CNN (mean ± std)	100% ± 0	100% ± 0	100% ± 0
PLS-LDA	100%	100%	100%

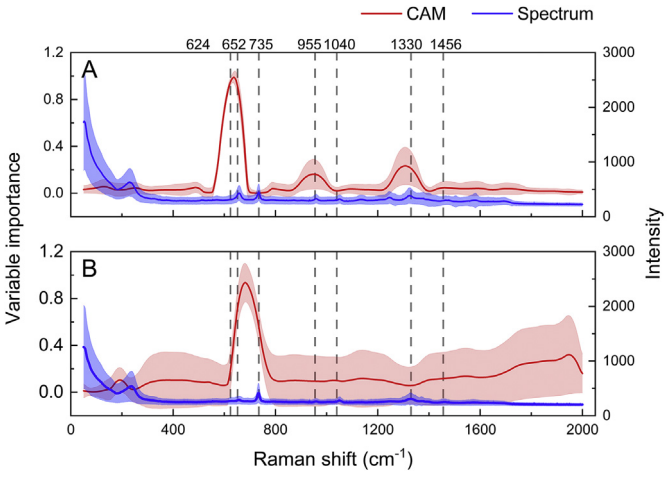


Fig. 2. Variable importance and raw spectra for the *E. coli* dataset of (A) *E. coli* 498, and (B) *E. coli* 1116. The red line represents the average variable importance, while the blue line is the mean spectrum. The shaded areas represent the standard deviation error. (For interpretation of the references to colour in this figure legend, the reader is referred to the Web version of this article.)

1456 cm^{-1}, according to a previous study [29]. The strongest Raman shifts, 735 and 1330 cm^{-1} are typically from adenine or adenosine monophosphate, while 624, 652 and 955 cm^{-1} are from aromatic ring skeletal, COO- bend, and C—N stretch, respectively [30]. The 1456 cm^{-1} is not clearly seen in this dataset. Another Raman shift 1040 cm^{-1} comes from the C—C ring breathing [30]. Few numbers of spectra show intensity peaks at about 1500 and 1600 cm^{-1}, which may be caused by random noises and contain no chemical information.

The second dataset is the meat dataset, which includes 120 samples of chicken, turkey, and pork spectra collected on a Spectra-Tech instrument [31] (Fig. A2). The spectra range from 1000 to 1800 cm^{-1} (mid-infrared) with 448 variables. The major feature at approximately 1650 cm^{-1} comes from water (O—H stretch) with a significant overlap from protein (amide I) [31]. The second-largest peak arises at approximately 1550 cm^{-1}, which is from amide II absorption of protein [32]. Another small peak can be observed at approximately 1740 cm^{-1} due to fat (C═O ester) content [31]. The peak at 1740 cm^{-1} is relatively more intense for pork meat among the three kinds of meats, which can been seen in Fig. A2. The dataset is open access at https://csr.quadram.ac.uk/%20example-datasets-for-download/.

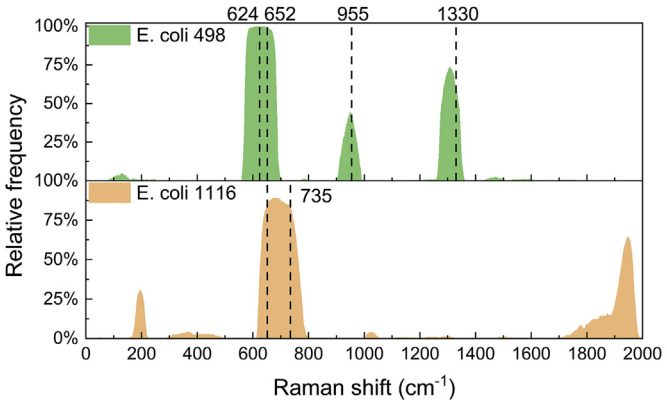


Fig. 3. The frequency distribution of the top ten percent of important variables for all correctly classified spectra for *E. coli* dataset.

All training and visualization processes were implemented on Python libraries Keras and Scikit-learn. The computations were performed on a Linux workstation (Ubuntu 14.04 LTS) with an Nvidia Geforce GTX 1080 Ti graphics card with 11 GB of RAM.

3. Results and discussion

3.1. Classification accuracy

The CNN and PLS-LDA approaches achieve high classification accuracies on both the *E. coli* and meat datasets (Table 2). For the *E. coli* dataset, the CNN model obtains mean classification accuracies of 99.48%, 98.82%, and 99.01% for ten repetitions on the training, validation, and test sets, respectively. The PLS-LDA model obtains a comparable classification accuracy of 98.83% on the test set for the *E. coli* dataset. The average classification accuracies among ten repetitions of the CNN model are higher than those of PLS-LDA, although there are small changes in the CNN model due to sophistication of the model or randomness from GPU computing. The CNN model obtains stable performance on the meat dataset among ten repetitions with 100% classification accuracy on training, validation, and test sets. The PLS-LDA model also obtains reasonably well on the meat dataset with 100% classification accuracies. Both CNN and PLS-LDA models obtain relatively high classification accuracies in the given datasets. A direct interpretation is critically needed to ensure excellent performance by the model is reliable.

3.2. Visualization of active variables

The most important variables selected by CNN and PLS-LDA models are consistent with the theoretical chemical characteristic bands for the two datasets (Figs. 2, 4 and 6). The active variables demonstrate a discriminative pattern at the category level for the CNN model (Figs. 2 and 4), while PLS-LDA models have the same important variables for different spectra (Fig. 6). This is because the variable importance for the CNN model is determined by the input spectrum and the model's weight while the VIPs are calculated by

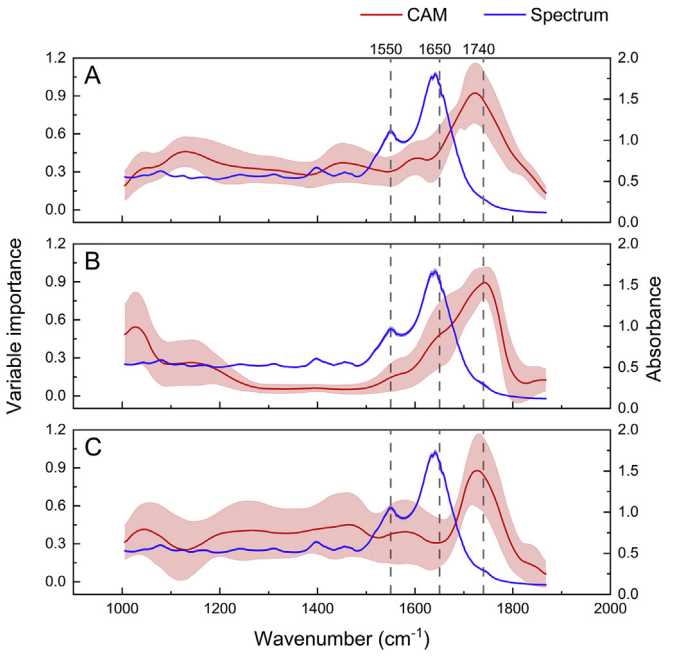


Fig. 4. Variable importance and raw spectra for the meat dataset of (A) chicken, (B) turkey, and (C) pork. The red line represents the average variable importance, while the blue line is the mean spectrum. The shaded areas represent the standard deviation error. (For interpretation of the references to colour in this figure legend, the reader is referred to the Web version of this article.)

the loadings and scores for the PLS-LDA model. The average line of the variable importance represents a general pattern for most of the spectral data, even if small variations exist for different samples (Figs. 2 and 4).

The active spectral regions selected by the CNN model for *E. coli* dataset are around 652 cm^{-1}, which corresponds to COO- bending from the cell wall (Fig. 2). To be specific, the discriminative region of active variables for *E. coli* 498 arise between 624 (aromatic ring skeletal) and 652 cm^{-1} while *E. coli* 1116 has active variables that arise between 652 and 735 cm^{-1} (adenine or adenosine monophosphate). For the top ten percent of the active variables, 99% samples of *E. coli* 498 yield the active variables between 624 and 635 cm^{-1}, while 82% samples of *E. coli* 1116 are between 652 and 735 cm^{-1} (Fig. 3). For *E. coli* 498, the variables approximately 955 cm^{-1} (C–N stretch) and 1330 cm^{-1} (adenine or adenosine monophosphate) also have high importance in the final classification decision. For *E. coli* 1116, a small number of samples yield high important variables from 1700 to 2000 cm^{-1}, which is hard to explain by chemical information (Fig. 3). The most contributed variables for misclassified spectra usually do not lie in chemical Raman shifts and demonstrate unique patterns for different spectra (Fig. A3). Besides, the Raman spectra have a large vertical offset due to the baseline effect. The CNN model, however, can detect the important variables through raw data without baseline correction.

The active variables for the meat dataset arise at approximately 1740 cm^{-1} (Fig. 4), which is the absorbance peak due to fat content (C=O ester). The pork meat spectra have higher absorbance peaks than chicken and turkey due to the higher fat content [31,33]. The fat content for pork meat is 3.82% followed by chicken and turkey meat with a fat content of 1.54 and 0.92%, according to a previous study [33]. The CNN model can learn the difference in spectra and show the highest activations in variables related to fat content. For the top ten percent of active variables, the chicken meat yields 90% samples with active variables between 1730 and 1750 cm^{-1}, while the turkey and pork meat have 78% samples with active variables

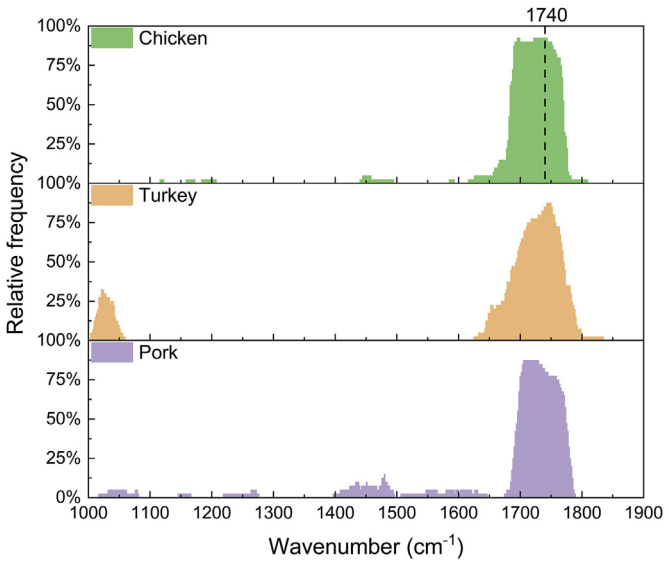


Fig. 5. The frequency distribution of the top ten percent of important variables for all correctly classified spectra for meat dataset.

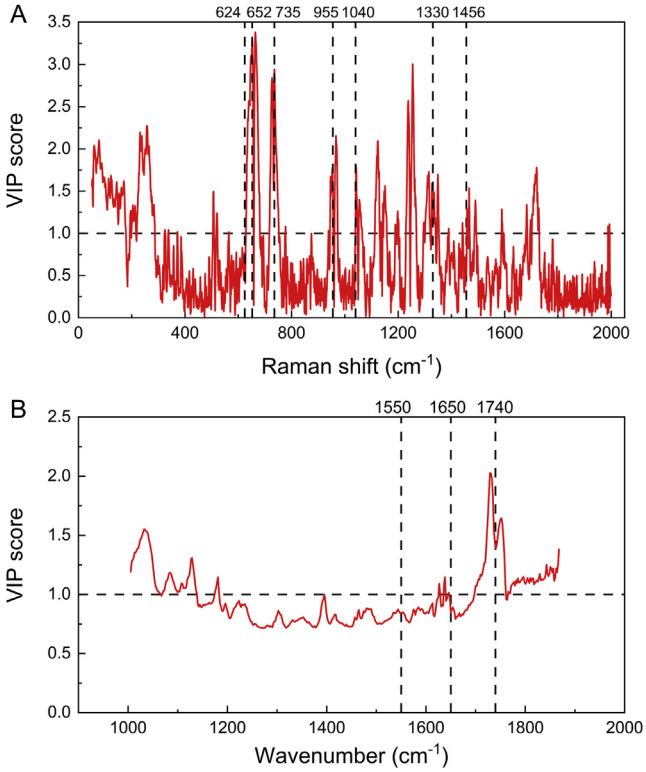


Fig. 6. The VIP score of the PLS-LDA model using (A) *E. coli* and (B) meat datasets.

between 1730 and 1750 cm^{-1} (Fig. 5). The model shows relatively lower activation at protein-related variables of 1550 and 1650 cm^{-1}. There exists a small difference in protein content for chicken, turkey, and pork, according to a previous study [33]. Additionally, the O–H stretch from water has a significant absorbance at 1650 cm^{-1}, which overlaps with the protein absorbance. Therefore, the model has low activation on variables of approximately 1650 cm^{-1}. For the turkey meat, the CNN model has a high average activation and standard deviation on variables near 1000 cm^{-1},

which does not contain any significant chemical absorbance peaks. The high standard deviation error indicates that only a few spectra have high activation on these variables, which is not a standard characteristic band of the turkey meat.

The most important variables selected by PLS-LDA models are consistent with the theoretical characteristic spectral bands as well as the active variables selected by CNN models (Fig. 6). For the *E. coli* dataset, the variables at around 652 and 955, and 1330 cm^{-1} have strong influences for the PLS -LDA model with VIPs lager than 1.5. These selected variables are consistent with the results of the CAM approach for *E. coli* 498 (Fig. 3). The selected important variables nearby 755 cm^{-1} are consistent with CAM results by *E. coli* 1116. Not all the variables with strong influence on the PLS-LDA model are related to chemical bond. For example, the variables with high VIPs at 250 or 1750 cm^{-1} are difficult to explain by the chemical information. For the meat dataset, the fat content related variables around 1740 cm^{-1} yield the largest influence on the PLS-LDA model (Fig. 6). Many variables between 1000 and 1150 cm^{-1} are selected as important variables with high VIP scores. This is comparable to the CAM results where high activations are found between 1100 and 1200 cm^{-1} for chicken meat and 1000 to 1100 cm^{-1} for turkey and pork meat (Fig. 4). The peaks at around 1400 cm^{-1} for the PLS-LDA model (Fig. 6) are consistent with high activation of the CAM for pork meat (Figs. 4 and 5).

3.3. Interpreting the feature representations of convolutional layers

The changes in the feature maps by different convolutional layers demonstrate how deep neural networks extract intricate features hierarchically. Feature maps of *E. coli* 498 are mainly discussed as presented in Fig. 7, while other classes can be found in the supplementary materials (Figs. A5-A8). The first convolutional layer performs convolution on single-channel input spectral data to produce 16 feature maps. The generated 16 feature maps, as the different input channels for the second convolutional layer, can produce 32 feature maps after the convolutional operation in this layer. The same is true for the third layer with 64 filters of 32 channels to generate 64 feature maps. The convolutional neural network acts as an information distillation pipeline to extract overall features, enhancing informative variables, and filtering out irrelevant information [34]. The feature maps genuinely correspond to the input spectra due to the spatial invariance of the CNN, with the spectral dimension decreasing gradually as the networks deepen. Visualization of feature maps gives insight into the data transformation via convolutional layers in the CNN model.

The first layer acts for spectral preprocessing and learns spectral shape characteristics. Feature maps by Conv1 are relatively smooth except for a high response in Raman shifts of 652 and 735 cm^{-1} For *E. coli* 498 (Fig. 7). Nearly half of the feature maps in the first layer have a similar shape to the original input spectrum and become less noisy than the input spectrum. The corresponding convolutional filters act as Savitzky-Golay filters to calculate the moving weighted average of the input spectra, smooth the spectral data, and remove artifacts [4,35]. The visualization of convolutional filters agrees with this finding (Fig. A4). Filters containing many non-zero elements and linear trends of intensity indicate similar effects of smoothing and derivative, which is consistent with the previous study [2]. A well-trained convolutional model can replace the traditional preprocessing treatment in the spectral analysis [4]. One-third of feature maps illustrate nearly zero activations on the whole spectrum, which are generated by rectified linear unit (ReLU) activations. The zero activations mean these filters are not sensitive to the input spectrum, which are promising for speedup and energy reduction [36,37]. The remaining feature maps have high activation on different Raman shift peaks.

The second convolutional layer allows neural networks to extract the enhanced informative Raman shift peaks on a few correlated variables with almost zero activations on the non-feature variables. The second convolutional layer recognizes the spectrum with significant variations for different filters. More than 70% of filters have high activations on variables between 600 and 700 cm^{-1}, which around theoretical Raman shift peaks 624 and 652 due to aromatic ring skeletal and COO- bend. Four filters (Number 4, 12, 29, and 31) are sensitive to the local variations of the Raman spectrum, which demonstrate fluctuation patterns over different variables. The 23rd, 26th, and 28th filters learn the overall trend of the spectrum with a similar trend to the input spectrum. The variables approximated to the characteristic bands are highly activated, that have great importance in the spectral classification. The uninformative variables that make a little contribution to the model decision are reduced to zero in the second layer.

The feature maps in the third convolutional layer demonstrate an increased stable activation on critical Raman shift peaks at approximately 624, 652, 955, and 1330 cm^{-1}. The feature maps are arranged by weights related to *E. coli* 498 in the fully connected layer (Fig. 7). The higher weights represent more critical contributions in the final classification. The feature maps with positive weights related to the *E. coli* 498 extract the specific features belonging to this class. Among the three highly activated Raman shift peaks, variables between 624 and 652 cm^{-1} (come from aromatic ring skeletal and COO- bend) make the enormous contributions in the classification. The feature maps with zero weight have no influence on the final classification as the general patterns for both classes. The feature maps with negative weights related to the *E. coli* 498 have a negative effect on the classification, which may distinguish features of the other category. These feature maps yield high activations on characteristic variables between 652 and 735 cm^{-1}, which are specific characteristic bands belonging to *E. coli* 1116. The shape of the features belongs to *E. coli* 1116 can be seen in Fig. A5 in the supplemental materials.

The *E. coli* dataset is presented in this section. The feature maps for the meat dataset are presented in Figs. A6-A8 in supplemental materials. A hierarchical feature representation pattern in convolutional layers is found for the meat dataset. The chicken, turkey, and pork meat show class specific feature maps in the third layer.

The CNN model classifies a spectrum based on the global average pooling results and the trained weights in the fully connected layer (Fig. 8). The mean spectrum of *E. coli* 498 is adopted as an example here for detailed illustration. The feature maps in the third convolutional layer as shown in Fig. 7 are transferred to a vector (Fig. 8A) by global average pooling. Weights related to class *E. coli* 498 and 1116 (Fig. 8B) multiply with the output of the GAP layer to generate the weighted values in Fig. 8C. The S_c in Fig. 8D is a summation of the weighted values for two classes, respectively. The S_1 related to *E. coli* 498 is 2.30, while the S_2 corresponding to *E. coli* 1116 is −2.37, as presented in Fig. 8D. The model can make a decision that the spectrum belongs to *E. coli* 498 after the *softmax* layer.

3.4. Model reliability

The CNN approach obtains more stable classification accuracies compared with the PLS-LDA model on Monte-Carlo cross-validation. The CNN models achieve classification accuracies from 98.42 to 99.01% on raw spectral data, while the PLS-LDA approach demonstrates comparable fluctuating accuracies from 96.49 to 99.42% on preprocessed data by first derivative SG filter (Fig. 9). The CNN models demonstrate high classification accuracies for different dataset partitions with the same hyperparameters. For the PLS-LDA model, the optimal number of latent variables changes in 2 or 3 for different dataset partitions (Table A2). The CNN models, therefore,

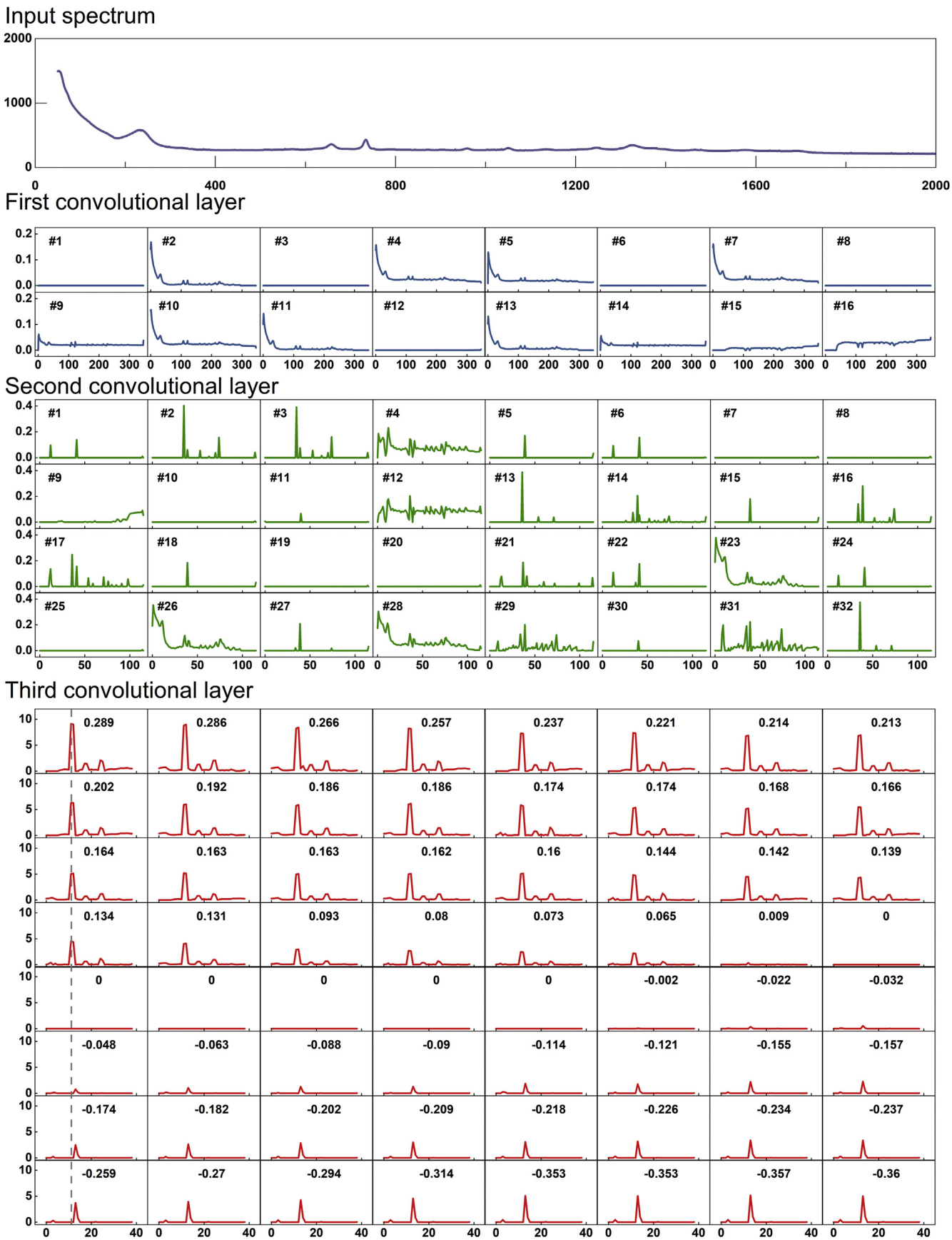


Fig. 7. Visualization of the average input spectrum and feature maps of three convolutional layers for *E. coli* 498. The numbers in the first and the second layers represent the order of channels. The numbers displayed on the third layer are the weights of the corresponding feature maps related to *E. coli* 498. Feature maps in layer Conv3 are sorted by these weights.

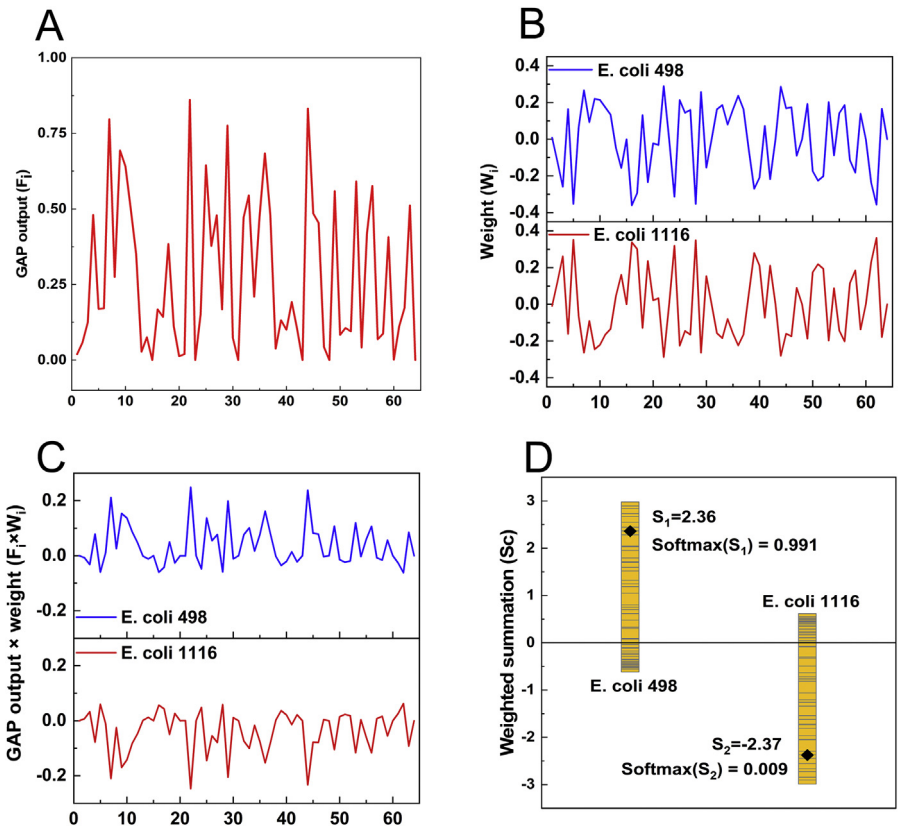


Fig. 8. The classification process for a given spectrum. (A) Output of the GAP layer. (B) Weights in the fully connected layer related to *E. coli* 498 and *E. coli* 1116, respectively. (C) Weighted value of the output in the GAP layer. (D) Weighted summations of the output in the GAP layer, Stacked bars represent weighed values while the diamond represents a summation of all these values. The x-axis in A, B, and C represents different neurons in the GAP layer.

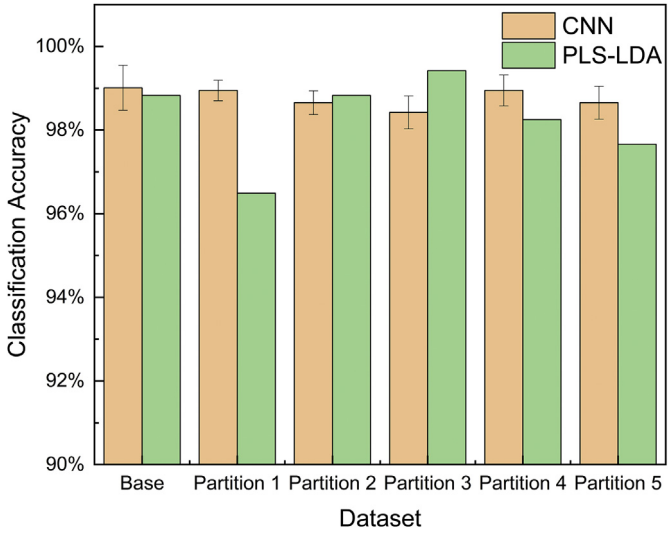


Fig. 9. Classification accuracies for CNN and PLS-LDA models by base scenario and five dataset partitions for the *E. coli* dataset. The average classification accuracies with standard deviation error bars of ten repeated runs are presented for the CNN model.

are more robust to different dataset distributions than PLS-LDA models.

The active variables selected by CNN and PLS-LDA models are stable on the Monte-Carlo cross-validation with different data partitions (Fig. 10). For the CNN approach, the most activated variables are approximately 624, 652, 735, 955, and 1330 cm^{-1}, which are typical theoretical Raman shift bands. The highest activation variables at near 624, 652, and 735 cm^{-1} are identical for the six dataset partitions. Small changes arise at variables that contain no chemical information. For PLS-LDA models, the VIP approach also provides reliable results for 6-time data partitions with small fluctuations for wavelength variables. Variables with chemical information at approximately 652 and 735 cm^{-1}, however, always have increased activations. These variables are consistent with the active variables identified by CNN models. CNN models are reliable with stable accuracies and important variables under different dataset partitions, which is consistent with the performance of PLS-LDA models with the VIP approach. The results indicate the CAM approaches can provide a satisfying understanding of the critical activations learned by the convolutional neural network.

4. Conclusion and future work

The present study interprets the learning mechanism of convolutional neural networks on vibrational spectral analysis and compares the results with PLS-LDA approach. The convolutional neural networks as well as PLS-LDA models achieve high classification accuracies on both Raman and mid-infrared spectral datasets. The interpretation of both CNN and PLS-LDA approaches is based on the models with high performances. This study provides some insight in how convolutional networks convert the one-dimensional spectral data to its classification results by visualizing the active variables and intricate feature representations. The active variables exhibit a discriminative pattern at the category level with a high relation to theoretical chemical variables for CNN models. The multiple convolutional operations extract features

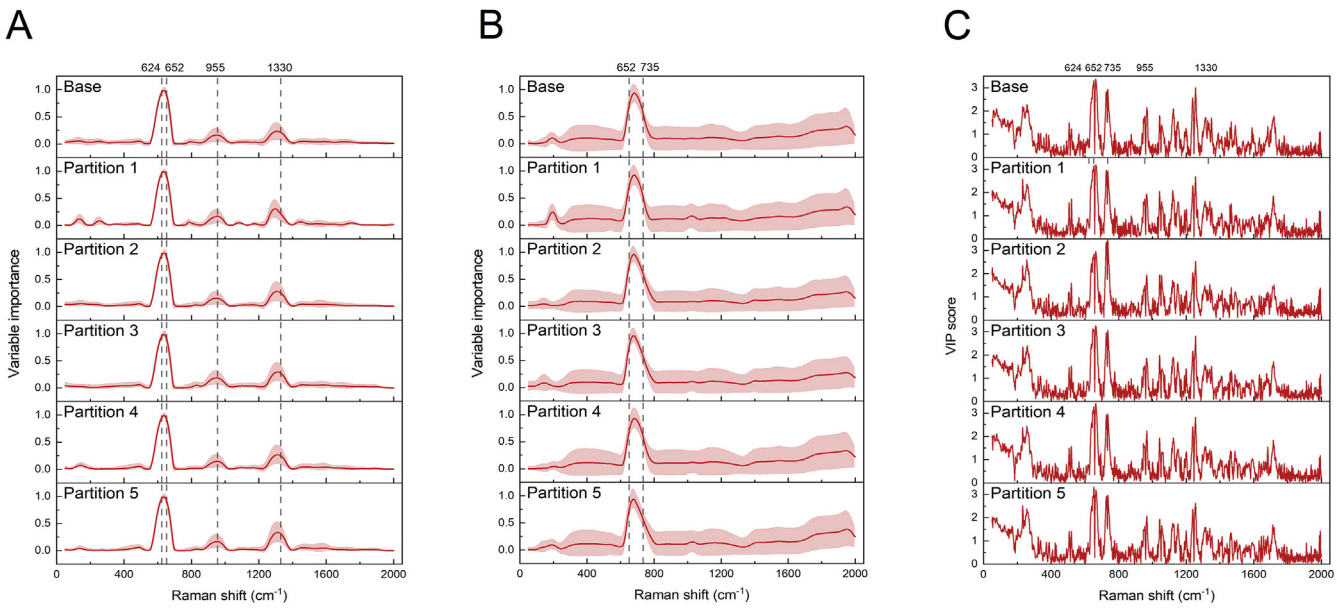


Fig. 10. Model sensitivity to different dataset partitions. Variable importance for (A) *E. coli* 498 and (B) *E. coli* 1116 of CNN models and (C) coefficients of PLS-LDA models for base scenario and five dataset partitions. (For interpretation of the references to colour in this figure legend, the reader is referred to the Web version of this article.)

hierarchically as a distillation pipeline. The first layer acts for spectral preprocessing and learns spectral shape characteristics. The second layer extracts enhanced typical spectral peaks on few correlated variables. The third layer demonstrates increased stable activation on major chemical peaks. The reliabilities of CNN and PLS-LDA models are tested on different dataset partitions with Monte-Carlo cross-validation. The selected active variables for both CNN and PLS-LDA models are stable for different dataset partitions.

The presented explainable approach can further be applied on various CNN models to interpret their behaviors on spectral analysis. Grad-CAM [38] is a potential approach to visualize active variables for more complex CNN models. Based on the comprehensive understanding of the CNN models, we can optimize the model structure and improve model robustness in the future studies for global models with different products.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

CRediT authorship contribution statement

Xiaolei Zhang: Writing - original draft, Formal analysis, Methodology, Writing - review & editing. **Jinfan Xu:** Writing - original draft, Formal analysis, Methodology, Writing - review & editing. **Jie Yang:** Methodology, Writing - review & editing. **Li Chen:** Investigation. **Haibo Zhou:** Investigation, Resources. **Xiangjiang Liu:** Investigation, Resources. **Haifeng Li:** Investigation, Resources. **Tao Lin:** Writing - review & editing, Project administration, Funding acquisition. **Yibin Ying:** Writing - review & editing, Supervision.

Acknowledgments

This work was partially funded by the National Natural Science Foundation of China under Grant Number F030601 and 31701316, and Zhejiang University.

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.aca.2020.03.055.

References

[1] Y. Chen, Z. Wang, Quantitative analysis modeling of infrared spectroscopy based on ensemble convolutional neural networks, Chemometr. Intell. Lab. Syst. 181 (2018) 1–10, https://doi.org/10.1016/j.chemolab.2018.08.001.
[2] J. Acquarelli, T. van Laarhoven, J. Gerretzen, T.N. Tran, L.M.C. Buydens, E. Marchiori, Convolutional neural networks for vibrational spectroscopic data analysis, Anal. Chim. Acta 954 (2017) 22–31, https://doi.org/10.1016/j.aca.2016.12.010.
[3] X. Zhang, T. Lin, J. Xu, X. Luo, Y. Ying, DeepSpectra: an end-to-end deep learning approach for quantitative spectral analysis, Anal. Chim. Acta 1058 (2019) 48–57, https://doi.org/10.1016/j.aca.2019.01.002.
[4] C. Cui, T. Fearn, Modern practical convolutional neural networks for multivariate regression: applications to NIR calibration, Chemometr. Intell. Lab. Syst. 182 (2018) 9–20, https://doi.org/10.1016/j.chemolab.2018.07.008.
[5] S. Malek, F. Melgani, Y. Bazi, One-dimensional convolutional neural networks for spectroscopic signal regression, J. Chemom. 32 (2018) e2977, https://doi.org/10.1002/cem.2977.
[6] C. Ni, D. Wang, Y. Tao, Variable weighted convolutional neural network for the nitrogen content quantization of Masson pine seedling leaves with near-infrared spectroscopy, Spectrochim. Acta Part A Mol. Biomol. Spectrosc. 209 (2019) 32–39, https://doi.org/10.1016/j.saa.2018.10.028.
[7] W.J. Murdoch, C. Singh, K. Kumbier, R. Abbasi-Asl, B. Yu, Definitions, methods, and applications in interpretable machine learning, Proc. Natl. Acad. Sci. Unit. States Am. (2019) 201900654, https://doi.org/10.1073/pnas.1900654116.
[8] J. Yang, J. Xu, X. Zhang, C. Wu, T. Lin, Y. Ying, Deep learning for vibrational spectral analysis: recent progress and a practical guide, Anal. Chim. Acta 1081 (2019) 6–17, https://doi.org/10.1016/j.aca.2019.06.012.
[9] R. Guha, On the interpretation and interpretability of quantitative structure–activity relationship models, J. Comput. Aided Mol. Des. 22 (2008) 857–871, https://doi.org/10.1007/s10822-008-9240-5.
[10] A. Fernandez, F. Herrera, O. Cordon, M. Jose del Jesus, F. Marcelloni, Evolutionary fuzzy systems for explainable artificial intelligence: why, when, what for, and where to? IEEE Comput. Intell. Mag. 14 (2019) 69–81, https://doi.org/10.1109/MCI.2018.2881645.
[11] Y.-H. Yun, H.-D. Li, B.-C. Deng, D.-S. Cao, An overview of variable selection methods in multivariate analysis of near-infrared spectra, TrAC Trends Anal. Chem. (Reference Ed.) 113 (2019) 102–115, https://doi.org/10.1016/j.trac.2019.01.018.
[12] L. Nørgaard, A. Saudland, J. Wagner, J.P. Nielsen, L. Munck, S.B. Engelsen, Interval partial least-squares regression (iPLS): a comparative chemometric study with an example from near-infrared spectroscopy, Appl. Spectrosc. 54 (2000) 413–419, https://doi.org/10.1366/0003702001949500.
[13] M. Arakawa, Y. Yamashita, K. Funatsu, Genetic algorithm-based wavelength

selection method for spectral calibration, J. Chemom. 25 (2011) 10–19, https://doi.org/10.1002/cem.1339.
[14] Y.-H. Yun, J. Bin, D.-L. Liu, L. Xu, T.-L. Yan, D.-S. Cao, Q.-S. Xu, A hybrid variable selection strategy based on continuous shrinkage of variable space in multivariate calibration, Anal. Chim. Acta 1058 (2019) 58–69, https://doi.org/10.1016/j.aca.2019.01.022.
[15] J. Zhang, H. Yan, Y. Xiong, Q. Li, S. Min, An ensemble variable selection method for vibrational spectroscopic data analysis, RSC Adv. 9 (2019) 6708–6716, https://doi.org/10.1039/C8RA08754G.
[16] E.J. Bjerrum, M. Glahder, T. Skov, Data Augmentation of Spectral Data for Convolutional Neural Network (CNN) Based Deep Chemometrics, vols. 1–10, 2017. http://arxiv.org/abs/1710.01927.
[17] W. Ng, B. Minasny, M. Montazerolghaem, J. Padarian, R. Ferguson, S. Bailey, A.B. McBratney, Convolutional neural network for simultaneous prediction of several soil properties using visible/near-infrared, mid-infrared, and their combined spectra, Geoderma 352 (2019) 251–267, https://doi.org/10.1016/j.geoderma.2019.06.016.
[18] J. Dong, M. Hong, Y. Xu, X. Zheng, A practical convolutional neural network model for discriminating Raman spectra of human and animal blood, J. Chemom. (2019) 1–12, https://doi.org/10.1002/cem.3184.
[19] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, A. Torralba, Learning deep features for discriminative localization, in: 2016 IEEE Conf. Comput. Vis. Pattern Recognit., IEEE, 2016, pp. 2921–2929, https://doi.org/10.1109/CVPR.2016.319.
[20] A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet classification with deep convolutional neural networks, Adv. Neural Inf. Process. Syst. (2012) 1097–1105, https://doi.org/10.1016/j.protcy.2014.09.007.
[21] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, Int. Conf. Learn. Represent. (2015) 1–14, https://doi.org/10.1016/j.infsof.2008.09.005.
[22] L. Liu, M. Ji, M. Buchroithner, Transfer learning for soil spectroscopy based on convolutional neural networks and its application in soil clay content mapping using hyperspectral imagery, Sensors (2018) 18, https://doi.org/10.3390/s18093169.
[23] M. Lin, Q. Chen, S. Yan, Network in Network. https://doi.org/10.1109/ASRU.2015.7404828, 2013.
[24] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, in: 2015 IEEE Conf. Comput. Vis. Pattern Recognit., IEEE, 2015, pp. 1–9, https://doi.org/10.1109/CVPR.2015.7298594.
[25] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: 2016 IEEE Conf. Comput. Vis. Pattern Recognit., IEEE, 2016, pp. 770–778, https://doi.org/10.1109/CVPR.2016.90.
[26] E.K. Kemsley, Discriminant analysis of high-dimensional data: a comparison of principal components analysis and partial least squares data reduction methods, Chemometr. Intell. Lab. Syst. 33 (1996) 47–61, https://doi.org/10.1016/0169-7439(95)00090-9.
[27] T. Mehmood, K.H. Liland, L. Snipen, S. Sæbø, A review of variable selection methods in Partial Least Squares Regression, Chemometr. Intell. Lab. Syst. 118 (2012) 62–69, https://doi.org/10.1016/j.chemolab.2012.07.010.
[28] B. Galindo-Prieto, L. Eriksson, J. Trygg, Variable influence on projection (VIP) for orthogonal projections to latent structures (OPLS), J. Chemom. 28 (2014) 623–632, https://doi.org/10.1002/cem.2627.
[29] H. Zhou, D. Yang, N.P. Ivleva, N.E. Mircescu, R. Niessner, C. Haisch, SERS detection of bacteria in water by in situ coating with Ag nanoparticles, Anal. Chem. 86 (2014) 1525–1533, https://doi.org/10.1021/ac402935p.
[30] Y. Liu, H. Zhou, Z. Hu, G. Yu, D. Yang, J. Zhao, Label and label-free based surface-enhanced Raman scattering for pathogen bacteria detection: a review, Biosens. Bioelectron. 94 (2017) 131–140, https://doi.org/10.1016/j.bios.2017.02.032.
[31] O. Al-Jowder, E.K. Kemsley, R.H. Wilson, Mid-infrared spectroscopy and authenticity problems in selected meats: a feasibility study, Food Chem. 59 (1997) 195–201, https://doi.org/10.1016/S0308-8146(96)00289-0.
[32] H. Fabian, W. Mantele, in: J.M. Chalmers (Ed.), Infrared Spectroscopy of Proteins, Handb. Vib. Spectrosc., John Wiley & Sons, Ltd, Chichester, UK, 2006, pp. 1–42, https://doi.org/10.1002/0470027320.s8201.
[33] Y.H. Lan, J. Novakofski, R.H. McCUSKER, M.S. Brewer, T.R. Carr, F.K. McKEITH, Thermal gelation of pork, beef, fish, chicken and Turkey muscles as affected by heating rate and pH, J. Food Sci. 60 (1995) 936–940, https://doi.org/10.1111/j.1365-2621.1995.tb06265.x.
[34] F. Chollet, Deep Learning with Python, 2017.
[35] G.M. Savitzky A, Smoothing and differentiation of data by simplified least squares procedures, Anal. Chem. 36 (1964) 1627–1639.
[36] S. Han, J. Pool, J. Tran, W.J. Dally, Learning Both Weights and Connections for Efficient Neural Networks. http://arxiv.org/abs/1506.02626, 2015, 1-9.
[37] A. Parashar, M. Rhu, A. Mukkara, A. Puglielli, R. Venkatesan, B. Khailany, J. Emer, S.W. Keckler, W.J. Dally, SCNN: an Accelerator for Compressed-Sparse Convolutional Neural Networks, 2017. http://arxiv.org/abs/1708.04485.
[38] R.R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, D. Batra, Grad-Cam, Visual explanations from deep networks via gradient-based localization, in: 2017 IEEE Int. Conf. Comput. Vis., IEEE, 2017, pp. 618–626, https://doi.org/10.1109/ICCV.2017.74.