Spectrochimica Acta Part A: Molecular and Biomolecular Spectroscopy 337 (2025) 126068

Contents lists available at ScienceDirect

Spectrochimica Acta Part A:
Molecular and Biomolecular Spectroscopy

journal homepage: www.journals.elsevier.com/spectrochimica-acta-part-a-molecular-and-biomolecular-spectroscopy

Deep tutti-frutti II: Explainability of CNN architectures for fruit dry matter predictions☆

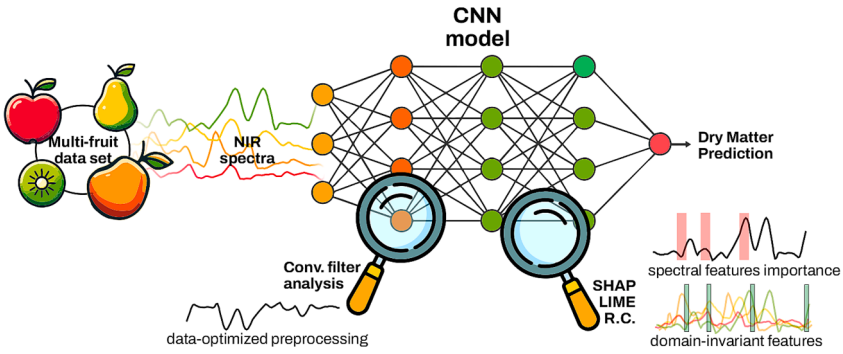
Dário Passos [a,b,c,*]

[a] CEOT – Center for Electronics, Optoelectronics and Telecommunications, Universidade do Algarve, Campus de Gambelas, 8005-189 Faro, Portugal
[b] Universidade do Algarve, Faculdade de Ciências e Tecnologia, Departamento de Física, Campus de Gambelas, 8005-189, Faro, Portugal
[c] CISCA – Algarve Cyber-Physical Systems Research Center, Universidade do Algarve, Campus de Gambelas, 8005-189 Faro, Portugal

HIGHLIGHTS

- CNN architectures for global models of dry matter predictions in multiple fruits.
- Explainability methods applied to CNNs allow us to identify relevant spectral bands.
- Depending on the CNN architecture, convolution filters learn different types of preprocessings.
- CNNs applied to a multi-fruit data set tend to rely on domain invariant features.

GRAPHICAL ABSTRACT



ARTICLE INFO

Keywords:
Fruit internal quality
NIR spectroscopy
Convolutional neural networks
Chemometrics
ML explainability

ABSTRACT

One of the criticisms that deep chemometric models usually face is their lack of explainability. In this work, three different explainability methods (Regression Coefficients, LIME and SHAP) are applied to different convolutional neural network (CNN) architectures, previously optimized for the task of multifruit dry matter content prediction based on NIR spectra. Additionally, a convolutional filter characterization is also performed to help clarify the type of modelling performed by the convolutional layers. The analysis allowed to extract information about the wavelength bands relevant to the models' performance (feature importance) and to understand how different convolutional layer topologies transform the spectra leading to three types of modelling: data driven preprocessing, dimensionality reduction and hierarchical feature extraction. Feature importance analysis indicates that the relevant spectral bands used by the different CNN architectures for prediction of dry matter is basically the same. They are the same as the bands relevant to PLS and these bands can be attributed to specific known vibrational groups. Moreover, in the context of the multifruit prediction task, the analysis also points out that CNNs tend to identify and use spectral features that are informative across different fruit spectra, much like domain-invariant features identified by di-CovSel variable selection.

☆ This article is part of a special issue entitled: 'SensorFINT' published in Spectrochimica Acta Part A: Molecular and Biomolecular Spectroscopy.

* Address: CEOT – Center for Electronics, Optoelectronics and Telecommunications, Universidade do Algarve, Campus de Gambelas, 8005-189 Faro, Portugal.
E-mail address: dmpassos@ualg.pt.

https://doi.org/10.1016/j.saa.2025.126068
Received 29 October 2024; Received in revised form 5 March 2025; Accepted 17 March 2025
Available online 19 March 2025

1. Introduction

Convolutional Neural Networks (CNNs) have been identified as one of the most promising deep learning algorithms for the prediction of fruit internal quality parameters, e.g. dry matter content, soluble solid content, acidity, protein content, etc., [1,2]. Despite their state-of the-art (SOTA) results in regression and classification tasks, the use of these algorithms by the general chemometrics community faces some distrust due to the "black box" image (lack of explainability) that deep neural network models project [1,3,4]. Unlike classical Partial Least Squares (PLS) [5] or Principal Component Analysis (PCA), whose internal decision process can be traced by looking at loadings, scores, Variable Importance in the Projection (VIP scores), etc., CNNs' often complex structures makes this task a bit more complex. Nonetheless, for the chemometrician working on/with these prediction models in the field of food quality control, some of the information that is most relevant is to identify if the spectral bands that the model relies on makes sense from a physicochemical point of view. This has been the most common scenario for the use of VIP scores [6] when SW-NIR spectral data (~ 780–1100 nm) is used to build PLS chemometric models aimed at prediction of internal quality parameters of fruit. Extracting information from Vis-NIR or SW-NIR spectra is a complicated problem because in this spectral range there are many combination bands and 2nd and 3rd overtones superpositions combined with the fact that biological tissues have a very complex chemical matrix. Given this complicated scenario, often there are unexpected/unexplainable mismatches between expected theoretical functional vibrational bands and the VIP highlighted relevant bands [7–9], i.e., even using a linear model like PLS does not guarantee that the most important spectral bands for prediction align with the intuitive theoretical expectation.

In the context of Machine Learning (ML), explainability and interpretability serve distinct yet complementary purposes [10,11]. Explainability refers to the methods and techniques that make the functioning of a model understandable to humans, often by highlighting the contribution of input features to the model's predictions (e.g. PLS VIP scores). On the other hand, interpretability involves understanding the internal mechanisms of the model, such as the behaviour and role of the convolutional filters in a CNN or how PLS scores elucidate the internal workings and transformations within the model. This type of post-train model analysis aims at enhancing the comprehensibility and trustworthiness of the models by addressing both what the model learns and how it processes the input data. This information is also paramount for these models to comply with the emerging regulatory environment for AI (Artificial Intelligence) around food safety/control in Europe. Even though explainability techniques can be used to gain additional information about deep learning models' decision processes in chemometrics, they are seldom applied by most authors because understandably, their focus is on the analysis of the predicted results themselves and model's viability rather than on the inner workings of the model.

In terms of CNNs for chemometric tasks, a few works explored the concepts of model explainability and interpretability. Acquarelli et al [12] used stability feature selection to identify relevant spectral bands for their spectral classification CNN models applied to 7 NIR data sets (beers, wines, tablets, coffees, olive oils, fruit juices and meats). Bjerrum et al [13] used the convolutional activations to study the role of the convolutional filters learned by their CNNs (aimed at pharmaceutical tablets classification) and tried to understand the type of automatic spectral transformations that the model learned during training. Cui & Fearn [14] proposed a method called "Regression Coefficients" (see Section 2.1) to study the impact of L2 and Dropout regularizations in their CNN models developed for quantify protein and ash content in two wheat flour data sets, and protein content in a third wheat seed data set. They also used this method to compare features importance between PLS and a CNN model. The method of computing the "Regression Coefficients" was also applied by Mishra & Passos [15] to their mango dry matter predicting CNN to highlight the different component contributions of their proposed augmented input vector (concatenation of spectral preprocessing methods).

Since neural networks are differentiable, several analysis techniques have been developed to study DL models based on the dynamics of internal gradient propagation. One of such techniques, Class Activation Mapping (CAM, [16]) was used by Zhang et al [17] to visualize spectral features' importance in their classification CNN applied to mid-infrared spectra of meat and Raman spectra of bacteria. Zhao et al [18] and Passos & Mishra [19] used a variant of the previous technique called gradient-weighted class activation mapping (Grad-CAM, [20]) to perform band selection and highlight features' importance for the problems of classification of hyperspectral aerial images of vegetation and wheat kernels, respectively. In Martins et al [21], the authors applied local interpretable model-agnostic explanations (LIME) to study which wavelength bands contributed the most for the orange SSC prediction of their "SpectraNet–53" deep learning model. In Schiemer et al [22], the authors proposed a data augmentation method to improve predictions of small data sets (proteins concentrations, monoclonal antibody concentrations and corn) and use Grad-CAM and Shapley additive explanations (SHAP) to study their (CNNs and PLS) models' response to their synthetic data. These are examples of the scarce application of explainability methods in deep learning chemometric modelling.

In Passos & Mishra [23] (henceforth "**paper I**") the authors explored a joint Neural Architecture Search (NAS) and Hyperparameter Optimization (HPO) pipeline that produced several 'shallow' CNN architectures (each optimized under a different paradigm) for the task of dry matter (DM) prediction on a multifruit data set. That work was done under the hypothesis that if spectral signatures of DM are the common (i.e. approximately the same) among different fruits, a CNN trained on a data set with different fruits should be able to learn these common features. The optimized CNN architectures differ slightly in terms of number of convolutional layers, number of convolutional filters, dense layers, etc. but their prediction results were on the same close range. Based on this scenario, a couple of questions arise. Do different CNN architectures rely on (i.e. extract) information from different spectral bands or not? What drives the internal decision process of these models in terms of architectural features? Can the most relevant spectral bands for each architecture be identified and check if they make sense from a chemical point of view? How do these bands compare to the bands identified by linear models such as PLS?

In order to shine some light on the aforementioned questions, this work explores the use of machine learning explainability methods that are model agnostic (LIME, SHAP and Regression Coefficients) to identify spectral features importance for different CNN architectures. Convolutional filter analysis is also performed to help understand what type of spectral "manipulations" are driving the models' performance and what type of features extraction are being performed. The objective is to better understand how CNNs operate in a chemometric context and extract relevant information that can be used in future works towards crafting better CNN architectures for this type of problem. In section 2, the multifruit data set used is presented and described in detail. Following this, the theoretical background of the explainability methods (Regression Coefficients, LIME and SHAP) is presented and the convolutional filter analysis methodology is laid out. Section 3 presents a comparison between the feature importance extracted by the different methods for all the different CNN models, an attempt to interpret these features from a spectral point of view of individual fruit type and an interpretation of the role of the convolutional filters. Finally, in Sections 4 and 5 conclusions are drawn and takeaway messages are summarised.

2. Methodology

In **paper I**, the authors explored the possibility of creating a global model for predicting dry matter (DM) content for different fruits. The strategy was to use a data set composed of NIR spectra of different fruits

(and their DM content measurements) and craft a CNN architecture that could take advantage of the spectral variability introduced by different fruits under the hypothesis that this variability could improve model generalization. Diverse CNN architectures were generated by a NAS/HPO process that consisted in testing which combinations of neural network layers and hyperparameters (within certain constraints) minimized an objective function, which in this case was the Cross-Validation Root Mean Squared Error (RMSECV). Due to computational constraints, the search space used by the NAS/HPO was restricted considering only architectures up to 6 hidden layers (i.e. at maximum 3 convolutional and 3 dense layers). Although this optimization process converged toward shallow architectures, it is possible that deeper models outside the probed NAS search space can improve results further. Several approaches/strategies were used for defining the objective function and that resulted in slightly different architectures with different topologies and hyperparameters. These strategies consisted in using 5-fold cross-validation, cross-fruit validation, different model weights initialization and custom convolutional filters constraints (e.g. single filter or multi filter) during NAS/HPO. This resulted in 9 different CNN models aimed at either DM prediction (pure regression) or parallel DM prediction and fruit classification (regression and classification output). In this work we focus exclusively on studying model's behaviour associated with the regression task.

2.1. The data set

The multifruit data set[1] used in **paper I,** was compiled by P. Mishra and is comprised of NIR spectra, acquired with several F750 Produce Quality Meter (Felix Instruments, Camas, USA), of 4 different fruits: apple (n = 1405), kiwi (n = 548), mango (n = 725) and pear (n = 319) (from multiple cultivars, origins, and harvest years) and their corresponding dry matter content. The spectra used are the 2nd derivative preprocessing extracted directly from the F750 Produce Quality Meter and contain 105 features in the 735–1050 nm range. This spectral range was chosen to avoid information about the fruit peels (available in the lower wavelengths) and the higher noise regions above 1050 nm. The 2nd derivative ensures that amplitude and baseline biases are eliminated between different subsets of spectra.

The pear data, var. 'Conference', corresponds to batch 1 ($n_1 = 239$ harvested September 2019) and (batch 3, $n_3 = 80$ harvested September 2020) from Mishra & Woltering [24] acquired from a fruit distributor in The Netherlands. Apples (unknown cultivar) and kiwi data (two cultivars 'Gold' and 'Hayward') were extracted from legacy demo models provided by Felix Instruments. The mango data is composed by samples from different cultivars containing 270 'Kiett' and 270 'Kent' mangoes from Brazil (harvest year 2020) but acquired and processed in The Netherlands [25]. The remaining mango samples are retrieved from Felix Instruments legacy models (origin unknown). The individual fruit data sets were randomly partitioned into train (80 %) and test (20 %) sets and combined into the multifruit train and test sets (see Table 1 of **paper I**).

Dimensionality reduction techniques are useful for spectra data sets exploration. UMAP (Uniform Manifold Approximation and Projection) [26], is used here (see Fig. 1) to reduce the dimensionality of the spectra to a two-dimensional low order embedding for visualization purposes. UMAP considers each spectrum as a point in a 105-dimensional space (the number of spectral features) and computes sample similarity by constructing a k-nearest neighbour graph for each sample and fuzzy set theory approaches to represent these similarities. Dimensionality reduction is achieved by finding a similar fuzzy topological representation in 2-dimensional space that best preserves the structure of the high-dimensional representation. This is done by using stochastic gradient descent to minimize the cross-entropy between the high-dimensional and low-dimensional fuzzy topological representations, resulting in a 2D coordinates representation for each spectrum in the data set. Similar spectra, in the original 105-dimensional space, are positioned close together in this new 2D low-dimensional representation. In Fig. 1, this technique identifies two clusters of pears (blue) that correspond to the two harvest years and four clusters of mangoes (the two cultivars from Brazil and two others in the Felix Instruments legacy demo model data). For apples it identifies a single cluster indicating a uniform batch and for kiwis there are signs of individual structures within the main cluster which is a sign of multiple batches and/or cultivars, etc. Overall, this is a complex multi-domain (different fruit) data set that presents variability even at the sub-domain level (e.g. different batches or harvest seasons).

2.2. Model agnostic methods: regression coefficients, LIME and SHAP

In the terms of model interpretability, this work explores three different methods that can be used to interpret how spectral features contribute to the CNNs predictions. These are model-agnostic methods (i.e. applicable to any "black-box" model) with different degrees of sophistication and can be applied to both linear and non-linear models. All of them are built around the same principle, the addition of perturbations to the input (spectral) features and a subsequent analysis of the prediction response of the model. This means that these three methods are applied to individual samples in a post training phase, by running the models in inference mode. The coefficients returned by these methods can be readily interpreted in the same fashion as VIP scores for PLS, i.e., they are a measure of feature importance for the different CNN architectures studied.

The "Regression Coefficients" (RC) method proposed by Cui, Fearn (2018) is the simplest one and is a straightforward numerical method for computing local sensitivity, offering an intuitive understanding of how individual features affect model predictions. The method works akin to perturbation theory in Physics, where a small perturbation is added sequentially to each spectral feature and then the difference between predictions based on original and perturbed spectra is computed (basically a local numerical derivative). In practice, regression coefficients w_i are computed by recurrently applying

$$w_i = \frac{f(x_1, \cdots, x_i + \varepsilon, \cdots, x_n) - f(\boldsymbol{x})}{\varepsilon},$$

where ε is the perturbation (a very small number, e.g. 10^{-5}), $\boldsymbol{x} = \{x_1, x_2, \cdots, x_n\}$ is an input spectrum and $f()$ is the prediction function (or prediction model). This method was implemented locally as a python function in our analysis pipeline and is available online in the author's github repository.

The second method, LIME (Local Interpretable Model-agnostic Explanations, [27]) approximates the model locally with a linear surrogate model and uses perturbations and weighted sampling to explain individual predictions. LIME creates new samples in the vicinity of the target sample by slightly altering the spectral features of the input spectrum (in feature space). In the context of spectral features, this means modifying the spectral data to generate new examples that are close to the original spectrum in feature space. LIME then builds a surrogate interpretable model, typically a linear model such as linear regression or Ridge regression, over these perturbed samples. The perturbed samples are weighted using a similarity kernel function (exponential kernel function), that gives more weight to samples that are closer to the input sample in feature space. This surrogate model is then used to approximate the local behaviour of the complex model, in this case a CNN, allowing for the interpretation of the model's predictions for a specific input sample (spectrum). In this work, the LIME python package "lime 0.2.0.1" [28] was used.

The third method, SHAP (SHapley Additive exPlanations) [29], is the most complex method and is derived from cooperative game theory

[1] https://github.com/dario-passos/DeepLearning_for_VIS-NIR_Spectra.

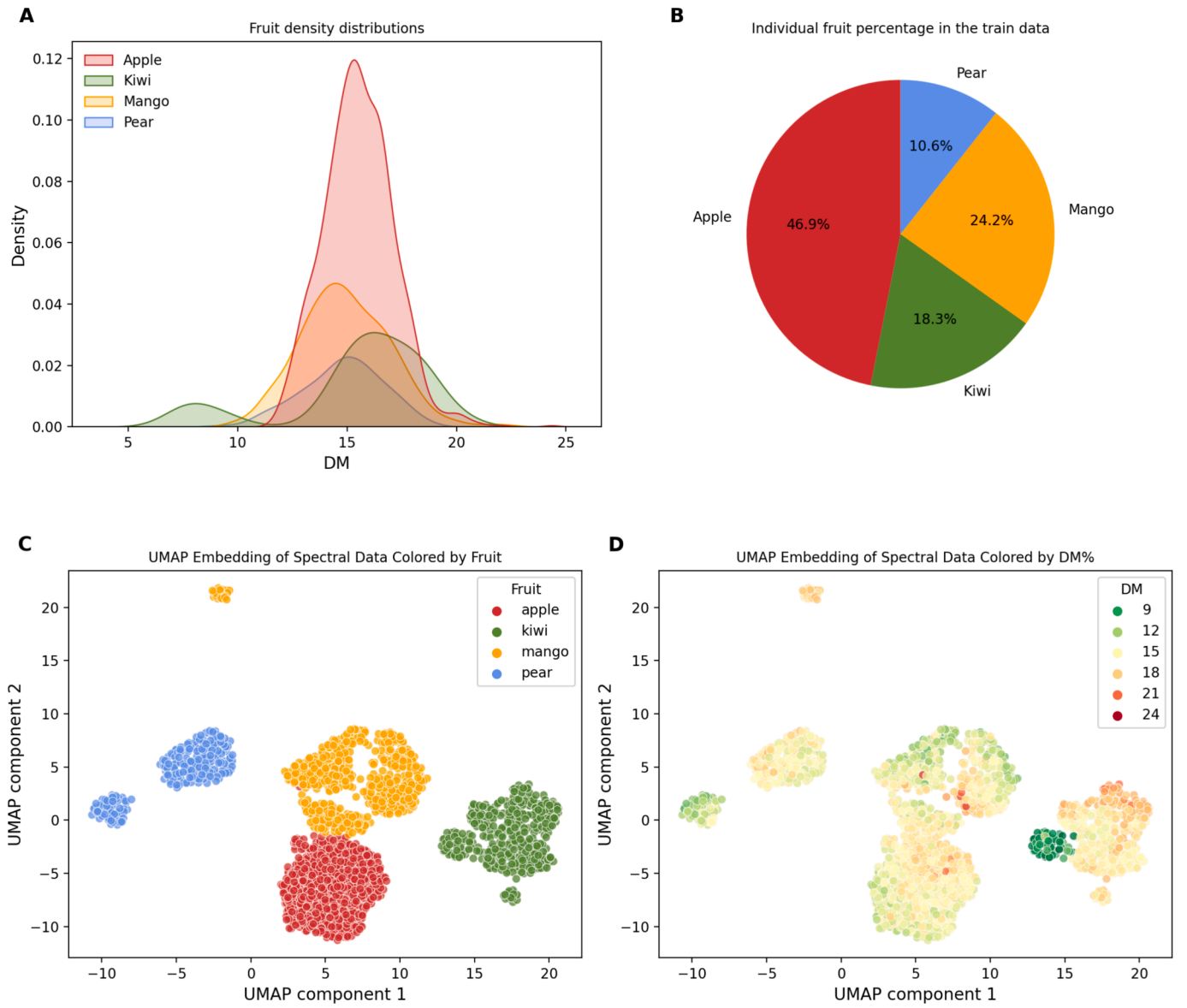


Fig. 1. A) Kernel density distribution of DM% of different fruits and B) percentage of each fruit in the training set. C) and D) show the UMAP representation (low dimensionality embedding) obtained from the spectra (coloured by fruit class in C and by DM% in D).

[30]. SHAP is based on Shapley values that fairly distribute the contribution of each feature to the model's prediction by considering all possible combinations of features. Unlike the two previous methods that provide local views of feature importance, SHAP provides consistent explanations that capture both local and global feature importance and their interactions. This method shows how the prediction result is affected by different sets of features (permutation of feature subsets or coalitions). These permuted feature coalitions are fed into the model that runs in inference mode and new predictions are produced. Then the marginal contribution of each coalition is computed (the increase in prediction value due to the addition or removal of a feature to the coalition). One of the advantages of SHAP is that Shapley explanation values are in the same units as the predicted variable which facilitates its interpretation. The sum of Shapley explanation values for a given sample equals the difference between the individual sample's prediction and the average prediction of the model. Since the number of permutations of features increases exponentially with the number of features, SHAP uses smart sampling techniques to reduce the computation costs. Since CNNs are differentiable models, we chose a SHAP implementation that takes advantage of the gradients computed by the models, the SHAP Gradient Explainer [31]. This method was designed for differentiable models and therefore is very suitable for deep neural networks. The SHAP Gradient Explainer uses the gradients of the model's output with respect to its input features to approximate SHAP values. From the three explainability methods presented here, SHAP is hailed as the more theoretical robust. The analysis presented here is based on the SHAP python package 0.46.0 (github SHAP).

The CNN models (different architectures of **paper I**) were instantiated by loading the pre-computed weights obtained during the optimization process (referred in **paper I** as the models used for single predictions). SHAP and LIME use the training set and the CNN models to construct the background explainability models and learn the overall features distribution. After this initial step, SHAP and LIME explanations were computed for all the samples in the multifruit test set. The Regression Coefficients method was applied directly to the test samples. The same feature importances were also computed for the 4 individual fruit subsets in the test set allowing to perform a comparison between feature importance in a global and individualized context. Since these methods are applied to individual samples, global features' importance is better represented by averaging the values obtained over the whole

test set. Following Molnar [11], overall feature importance is interpreted by computing the average of the absolute value of these coefficients, i.e. "*mean(abs(value))*". For comparison purposes, the SHAP, LIME and RC for a PLS model (using 7 latent variables), and its VIP scores were also computed.

2.3. Convolutional filter analysis

The weights of the convolutional filters in CNNs are learned during the model training phase. Although this is a topic still under research and seldom discussed in the literature, we believe that in CNNs applied to spectral analysis, these filters can provide different types of features learning depending on the topology of the network. In the case where multiple convolutional layers are used, the filters in the first layer tend to learn small scale patterns that are combined by the following convolutional layers into more general patterns across the whole spectra. This is the typical "hierarchical learning" behaviour of convolutional layers that is also described in computer vision tasks. On the other hand, if the model contains only one convolutional layer with one filter (or a few), then these filters perform a "kind of" data-driven preprocessing aimed at spectral optimization that enhances informative features. In this section, convolutional filter analysis is done following the methodology similar to that presented in Acquarelli et al [12], Cui & Fearn [14] and Zhang et al [17], i.e., for each CNN architecture, the filters'

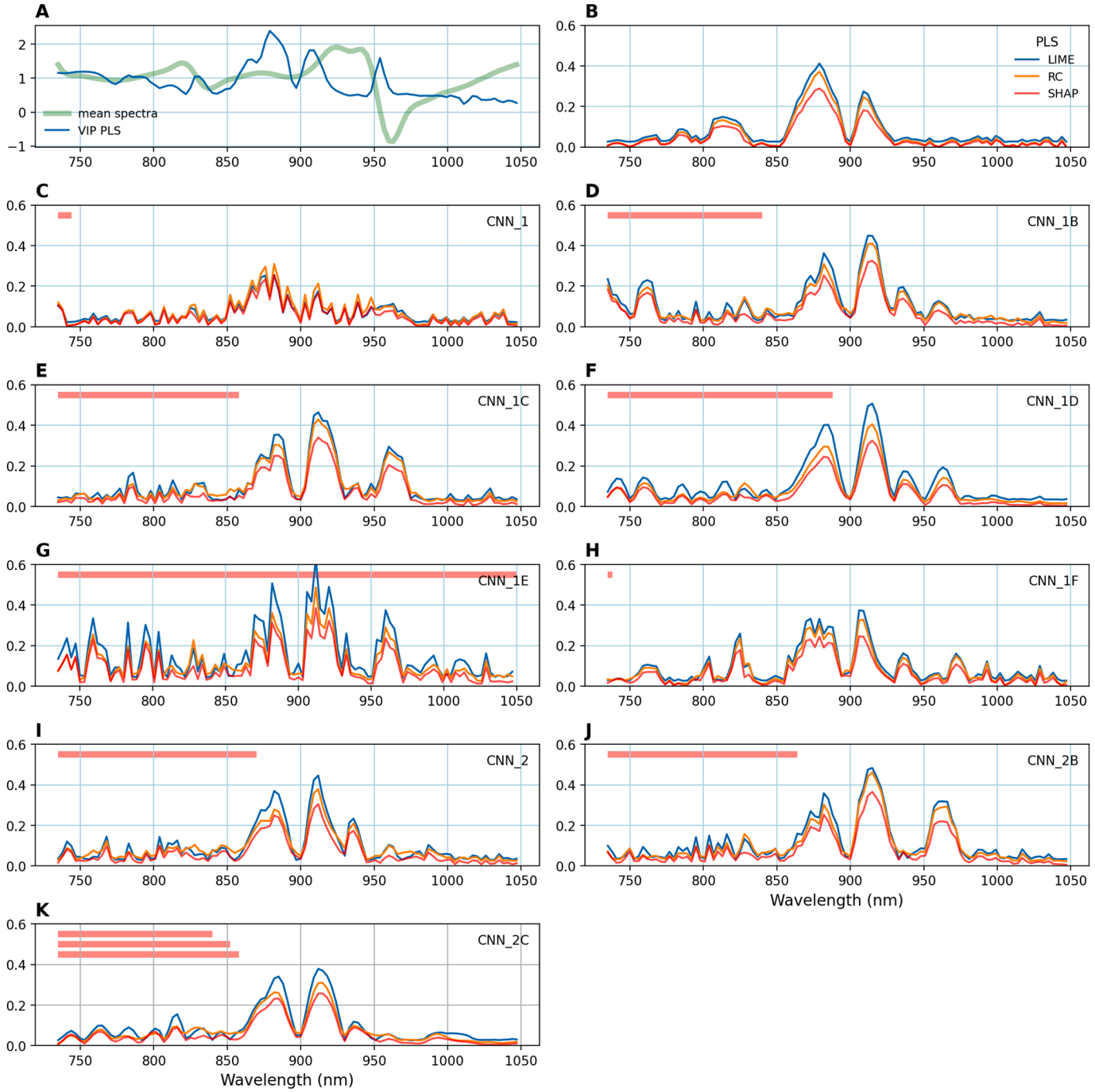


Fig. 2. A) Represents the PLS VIP scores (in blue) and the scaled 2nd derivative mean spectrum (in light green), for comparison purposes. Panels B) to K), show the *mean(abs(value))* of SHAP (red), LIME (blue) and RC (orange) for PLS and the CNN models (regression) as a function of spectral features (wavelengths). The red horizontal bar in the panels represents the width(s) of the filter(s) used by each CNN.

influence is studied by comparing the original input spectra with the convolutional layer output post-processed version (activation) and by looking at the specific filter patterns. This is followed by a tentative interpretation of these patterns in the context of spectral analysis. The CNNs architectures obtained through the NAS/HPO pipeline in **paper I** can be divided into 3 types:

i. 1 conv. layer with 1 filter (models: CNN_1, CNN_1B, CNN_1C, CNN_2, CNN_2B),
ii. 1 conv. layer with multiple filters (models: CNN_1D, CNN_1E, CNN_1F) and
iii. 3 conv. layers with multiple filters per layer (model: CNN_2C).

As the input spectra passes through the convolutional layer, it is convolved with the sliding window convolutional filter, a bias is added, and the resulting vector goes through an activation function (Exponential Linear Unit, ELU in the case of these models). As a linguistic simplification, sometimes the processed signal or output of a given layer is called the "activation" of that layer. For each type of architecture, the activations of the last convolutional layer are compared with the input spectra. Except for CNN_1, in all these architectures, the convolutional layers are followed by 1 hidden dense layer (with *n* units) and an output layer (with 1 unit). This means that the features extracted by the convolutional block are further non-linearly combined by these dense layers to form the final prediction.

3. Results

The results obtained from the explainability methods indicate that the three model-agnostic methods provide the same type of information. Despite the small differences in the amplitude of feature importances, the relevant bands identified by the different methods are basically the same.

3.1. Global feature importance interpretation

As it was mentioned earlier, SHAP can be used as a more intuitive guide for feature importance because its explanation values are in the same units as the predicted variable (DM% in this case). Therefore, for simplification purposes, in the following discussions, SHAP is used as the descriptor of feature importance. Fig. 2 shows the results from the different 3 methods as a function of wavelength for all the CNN models. For comparison purposes the PLS VIP scores and PLS features importance obtained using the 3 methods are also shown in the panels A and B of this figure.

When model-agnostic methods are applied to the PLS model, the relevant features identified differ slightly from those highlighted by VIP scores. The VIP scores identified as important (VIP ≥ 1) are found in the following ranges 735–760 nm, 780–795 nm, 828–832 nm, 857–893 nm, 900–917 nm and 950–958 nm. For the PLS SHAP explanations (represented here by *mean(abs(SHAP))*), there is no default threshold to define how important the features are. Instead, the regions where peaks are evident in comparison with the overall values are used. In this case, the relevant band are 783–789 nm, 805–826 nm, 857–896 nm, 903–925 nm. For details see Figs. S1 and S2 in the Supplementary Material appendix. The two main bands (centred around 880 nm and 908 nm) are coincident in VIP and SHAP. The peak at 954 nm in the VIP does not find a match in SHAP explanations and at shorter wavelengths, only the shallow peak around 784 nm coincides. Certain bands (e.g. 950–958 nm) seem important for the PLS internal workings (finding similar covariance directions between X and Y features) but do not seem relevant in terms of their aggregate contribution for the prediction of Y (i.e. DM%).

The different CNN models present resembling SHAP patterns, with two main broad spectral bands identified with peaks around 880 nm and 909 nm but with some difference between the location of secondary (shorter/narrower) peaks. The variability (jagged profile) of SHAP values for the CNNs seems to be proportional to the width of the convolutional filters used (represented by red horizontal bars in the subplots). CNN_1 uses a single narrow filter of width = 3 (stride = 1, padding='same'), CNN_1E uses 12 full range filters of width = 105 (padding = 'valid', i.e. it is a fixed filter, not sliding across) and CNN_1F uses 4 filters of width = 1 (stride 1, padding = 'same'). All the other architectures use wide filters with widths in the 35 to 51 range (and stride = 1, padding = 'same'). When a (sliding) wide filter is used, the convolution operation 'mixes' much more information from nearby features contributing to a 'dilution' of specific features influence and smoothing these curves.

Using as example CNN_1B, one can do the exercise of attributing known functional groups absorptions to the identified peaks. According to Golic et al [32], features between 735 nm and 760 nm can be attributed to the 3rd overtone of OH stretching and the 4th overtones of stretching vibrations of CH and CH2. Cen & He [33] and Shao et al. [34] report that around the 880 nm there are some bond absorptions for ArCH (CH aromatic groups present in phenolic compounds). The peaks around 914 nm and 932 nm can be matched to the overtones of CH and CH2 stretching (sugars) and the 960 nm smaller peak correspond to the 2nd overtone of OH stretching (water). This simple analysis indicates that the spectral features that CNN_1B relies on for prediction have chemical meaning and are consistent with compounds present in different fruits (water, sugars, phenolic compounds, etc). Some architectures are (more or less) sensitive to specific bands depending on how the spectral information is handled inside the network.

SHAP feature importance plots are an interesting tool to explore the influence that each spectral band has on the prediction. Fig. 3 shows the ordered feature importance according to its *mean(abs(SHAP))* value. This 'summary plot' combines feature importance with feature effects. Each dot on the right panel corresponds to the Shapley value for a feature and a sample. The position on the y-axis is determined by the feature importance and on the x-axis by the Shapley value. The colour represents the value of the feature, from low to high. Overlapping points are jittered in the y-axis direction, to get a sense of the distribution of the Shapley values per feature. A spectral feature with predominantly negative SHAP values indicates that it contributed to the decrease of the predicted DM value, while positive values indicate the opposite (compared to the average prediction).

A more detailed analysis of these spectral bands is beyond the scope of this work, but it is worth notice that, this type of feature importance plot can be used to gain further insight of what is happening in terms chemical compounds absorption. An example of a possible application is presented in the discussion section.

3.2. Single fruit feature importance interpretation

Instead of looking at the features importance by computing the average over all the fruit samples in the test set, one can compute the average of SHAP explanations for the individual fruit subsets. Before moving forward with the interpretation of features importance in this context, it is useful to note that the features in the train and test set were standardized using the mean and standard deviation of the multifruit train set. Even though the original 2nd derivative spectra show the same overall pattern for different fruits, once the feature standardization is done (required for input into the CNNs) the minute overall differences in spectra are accentuated and characteristic spectral patterns emerge for each fruit type (see Fig. 4).

The different post-standardization spectral patterns help explain why the CNN variants that were trained for parallel DM regression and fruit classification (see **paper I**) performed so well in the classification task even when the optimization process was not explicitly aimed at improving classification (the objective function used in the HPO pipeline was always the regression RMSECV). The training process itself (for each CV split) optimizes for the total loss function that, in this case, is the sum

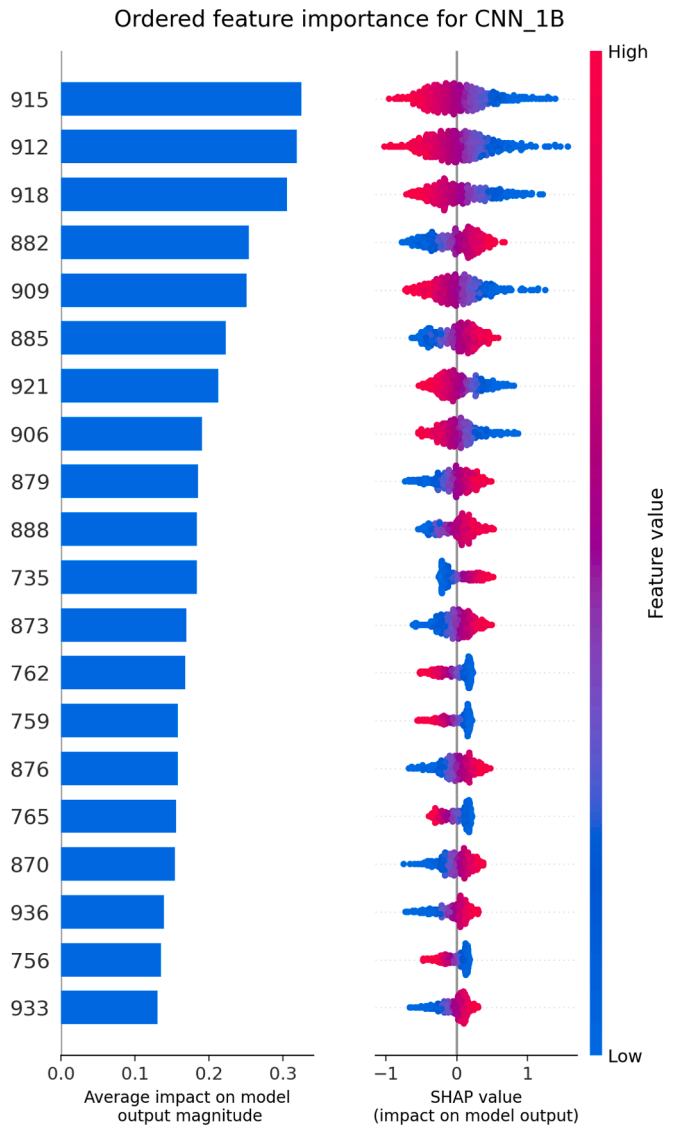


Fig. 3. SHAP explanations summary plot for model CNN_1B for the multifruit test set. Only the 20 most important features (wavelengths in nm) are displayed.

of the regression loss, the Mean Squared Error (MSE) and the classification loss, Categorical Cross-Entropy. For this data set, the former is of the order of 0.01 while the latter is a couple of magnitudes smaller. Therefore, the loss function that gradient descent minimizes in order to optimize the weights of the neural network was driven mainly by the regression task. Nonetheless, since the spectral patterns per fruit are indeed different (post-standardization), the CNNs were able to learn how to classify the spectra remarkably well. Now that we have established that the CNNs are actually "seeing" different spectral patterns for different fruits, we proceed to analyse the SHAP explanations for each fruit subset in the test set (presented in Fig. 5). Mean SHAP explanations extracted from CNN_1B are used as an example for features importance interpretation.

In Fig. 2 the displayed values correspond to the *mean(abs(SHAP))*. On the other hand, Fig. 5 displays the *mean(SHAP)* values, i.e., it allows to understand what is the average direction that a specific spectral feature is contributing to the prediction. The bottom panel displays the *mean(SHAP)* values for all the fruits in the test set while the preceding ones present the *mean(SHAP)* values per fruit class. While the *mean(abs(SHAP))* represents an overall measure of feature importance, the *mean(SHAP)* reflects both the magnitude and the direction (positive or negative) of the feature's impact on the model's predictions on average across all samples. As a result of the spectral differences that arise during standardization, for different fruits the same features might contribute towards increasing or decreasing the prediction value. Small shifts in peak locations are also visible between fruits. Moreover, it is worth remembering that this is an unbalanced data set dominated by apples data, and that also has an influence when computing global means. This type of post-training analysis can be used to help create future custom CNN architectures projected to pay "attention" to specific bands.

3.3. Convolutional filters and activations

In this section, the filters learned by the different architectures and the activations of the convolutional layers are presented with the aim of searching for common patterns that can help understand the internal dynamics of the neural networks. In CNNs whose architecture was optimized using cross-fruit validation (CNN_1 and CNN_1F), the best filters learned during HPO, are narrow (widths equal to 3 and 1 respectively). During this optimization phase these CNNs were trained on 3 fruits classes and validated on the remaining fruit class. Since spectral patterns in the training and validation data were different (see Fig. 4), the filters that worked best are narrow, indicating that the CNNs were relying on narrow bands. HPO was not able to find wider filters that cover larger spectral patterns (sets of features) that are common to both train and validation sets. On the other hand, the remaining architectures were optimized using 5-fold cross-validation where the train and validation subsets contain spectra from all fruit classes. In this case, the HPO leaned towards broader filters that mix information from many wavelengths in, what can be tentatively described as an attempt to find common patterns among different fruits. A broader filter is better at extracting features from different spectra that have peaks at different places.

3.4. Types of convolutional features extraction

Models CNN_1, CNN_1B, CNN_1E and CNN_2C are used as examples to demonstrate the different ways CNNs can handle feature extraction/processing. In Fig. 6 the mean (standardized) spectrum of pears (in black) is used as input sample to exemplify what type of "signal processing" is done by the convolutional layer (layer output in blue). The standard "deep-learning-style" convolution operation (which is actually a cross-correlation) between a set of input features x_i and a filter k with j weights, k_j, is given by

$$h_i = \sum_j k_j \cdot x_{i+j} + b$$

where b represents a bias term. The convolution output h_i is then further transformed by an activation function (ELU in this case), $z_i = ELU(h_i)$, where $ELU(h) = \begin{cases} h, h \geq 0 \\ \alpha(e^h - 1), x < 0 \end{cases}$, with $\alpha = 1$ (the default value in the tensorflow implementation). The filter weights used in the convolutional layer are presented in the inset panels of Fig. 6.

In the case of CNN_1 (Fig. 6A), the single filter learned (width = 3) resembles the coefficients of a second order numerical derivative (usually [1, −2, 1]). However, this filter's weights [0.70, −0.81, 0.61] are not integers nor symmetrical with respect to the central point which means that the filter does not fully cancel out baseline effects. Instead, it allows some of the original baseline signal to pass through. The slight asymmetry (left and right weights) can also indicate that the filter is not as efficient in detecting curvature changes in the spectra as a standard second derivative. The blue curve in Fig. 6A shows small bumps/peaks in the places where the original spectrum (in black) presents pronounced amplitude changes. This convolution filter works as a second derivative but it less sensitive to changes in spectral amplitudes and preserves some baseline effects. The ELU activation function further

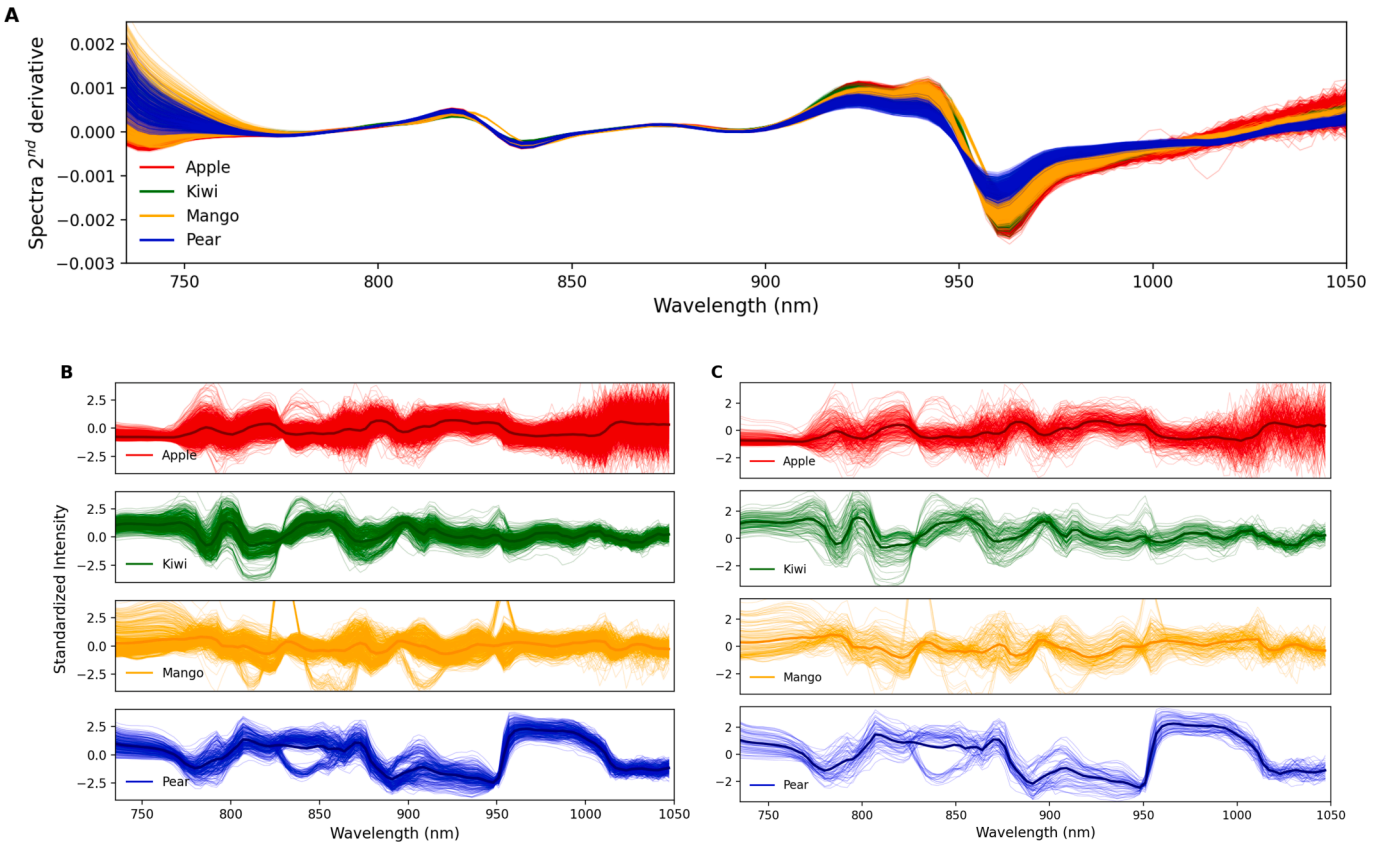


Fig. 4. A) 2nd derivative of the multifruit data. Standardized spectra split by fruit: B) train set, C) test set. The thicker/darker line represents the mean standardized spectrum.

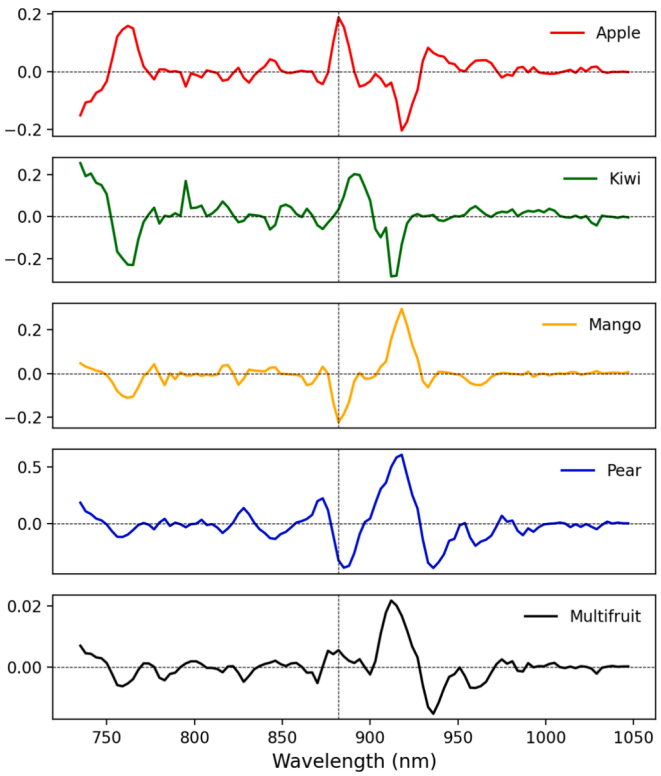


Fig. 5. *Mean(SHAP)* explanations for CNN_1B for each individual fruit subsets and the full multifruit set.

decreases the negative part of the signal.

This filter's behaviour is consistent with the hypothesis that during cross-fruit validation, CNN_1 focuses on narrow wavelength bands, and only slightly highlights specific peaks information.

Fig. 6B shows a different scenario for CNN_1B. In this case the learned filter is broad (width = 35) and has weights alternating between positive and negative values. This behaviour resembles a high-pass/edge detection filter that tends to emphasize transitions or amplitude changes in the spectral features. The output of the convolutional layer (blue curve) shows this behaviour where certain spectral transitions are highlighted as peaks. It is important to recall that, in the case CNN_1B, the filter is trained using samples from all fruit classes (in 5-fold cross validation) and that means that it learns to identify these transitions from the 4 different types of spectra (see Fig. 2B). For this reason, there isn't a perfect match between the transitions of the black curve (mean spectra of pears) and the activation of that spectra. This is expected because, since the 4 standardized spectra classes are different, it is likely that the filters highlight different parts of the signals.

In the case of CNN_1E (see Fig. 7), the filters width (and padding='valid') was a constraint imposed *a priori*. The only filter related hyperparameter that the HPO optimized for this model was the number of filters.

With these settings these filters do not "slide" over the signal and, since the filters width is the same as input signal, the convolution operation is reduced to the dot product between the input signal and the filter, i.e. a weighed linear combination of input features (plus a bias and ELU activation). The output of this operation is a single number per filter, i.e., this convolutional layer acts as a feature dimensionality reduction algorithm that transforms the 105 input features into a low dimensional representation of 12 features. This dimensionality reduction is somewhat analogous to the type of operation performed by PLS. The filters weighted sum across the entire input signal is analogous to

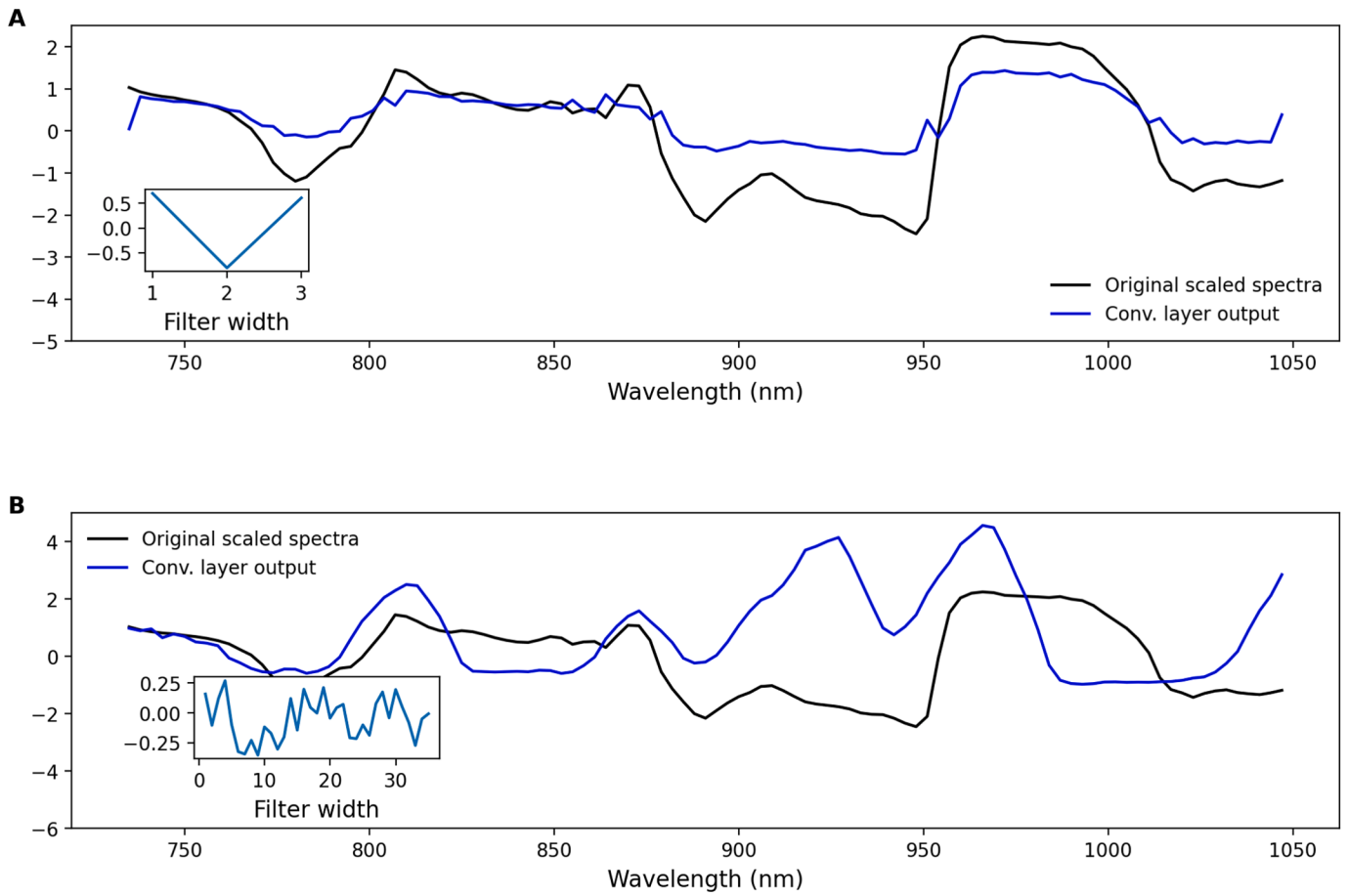


Fig. 6. Input sample spectra (black), output of convolutional layer post-activation (blue) and filter shape/weights in the inset panel for A) CNN_1 and B) CNN_1B.

the PLS loadings which describe how much each original feature contributes to the latent variables. The CNN feature maps represent the transformed version of the input, highlighting important aspects for prediction, just like how the PLS scores are the projections of the samples in the latent space. Of course, these PLS and CNN characteristic vectors are optimized with different objectives. While PLS aims at maximizing the covariance between features and response variable by projecting those into a latent space, CNNs minimize a loss function directly related to the prediction error using gradient-based optimization. Furthermore, the outputs of the convolutional layer are further non-linearly combined by the following dense layers in the model. This makes it challenging to map the features extracted by the convolutional filters to the input spectra akin to PLS loadings and scores. To do so, a deeper analysis of the model involving grad-CAM techniques would have to be applied.

The final model explored here is CNN_2C that, unlike the previous case, has three convolutional layers, each of which learned a different number of filters (27, 15, 15) with different widths (41, 39, 35). The different filters are presented in Fig. S7 of the Supplementary Materials. For this type of CNN architecture (with multiple convolutional layers), the standard interpretation of the filters is that in the first layers, the filters learn local features (peaks, valleys, etc.) and then combine these features into more complex patterns by the following layers (hierarchical feature learning). Usually, the rule of thumb for CNN architecture crafting suggests that the number of filters should increase from layer to layer, and that the width of these filters should be kept narrow (like in the work of [17]). This is not the case of CNN_2C because the filter numbers and widths were automatically optimized by the HPO pipeline and not handcrafted. For NIR spectra, this optimization process tends to choose wide filters instead of the narrow ones mentioned before. The filters from the first convolutional layer show pronounced oscillations in their weights, some of them periodic. This type of filter tends to highlight features that are distributed over a wide range of wavelengths. There are also several filters that have a flat behaviour with widths around zero. The contributions of these filters then become negligible for the overall inner processing of the CNN. A possible cause for this "filter suppression" is the L2 regularization applied over all layers. L2 penalizes weights that are too large or redundant for the overall prediction objective. This can also be interpreted as an over-parameterization of the model indicating that a shallower (leaner) version of this model could achieve the same results. The filters on the following two convolutional layers are also wide but their shapes are not periodic anymore which means that they became more specialized in identifying specific spectral patterns/regions.

4. Discussion and additional experiments

The post-training and analysis of CNN models in chemometrics is useful (and necessary) to understand how these algorithms work, improve confidence in their reliability and learn additional lessons that can be used for improving future models. The different CNN architectures developed in **paper I** and dissected here in detail, all shown consistent behaviour in terms of spectral feature reliance for prediction purposes. As seen by the SHAP explanation analysis, all CNNs rely heavily on the same spectral bands, despite small differences related to the width of the convolutional filters used. The extracted feature importance patterns also closely match the one obtained by PLS modelling (via both VIP scores and SHAP values). Moreover, these spectral bands could be assigned to known functional vibrational groups that are related with chemical compounds present in different fruits. As

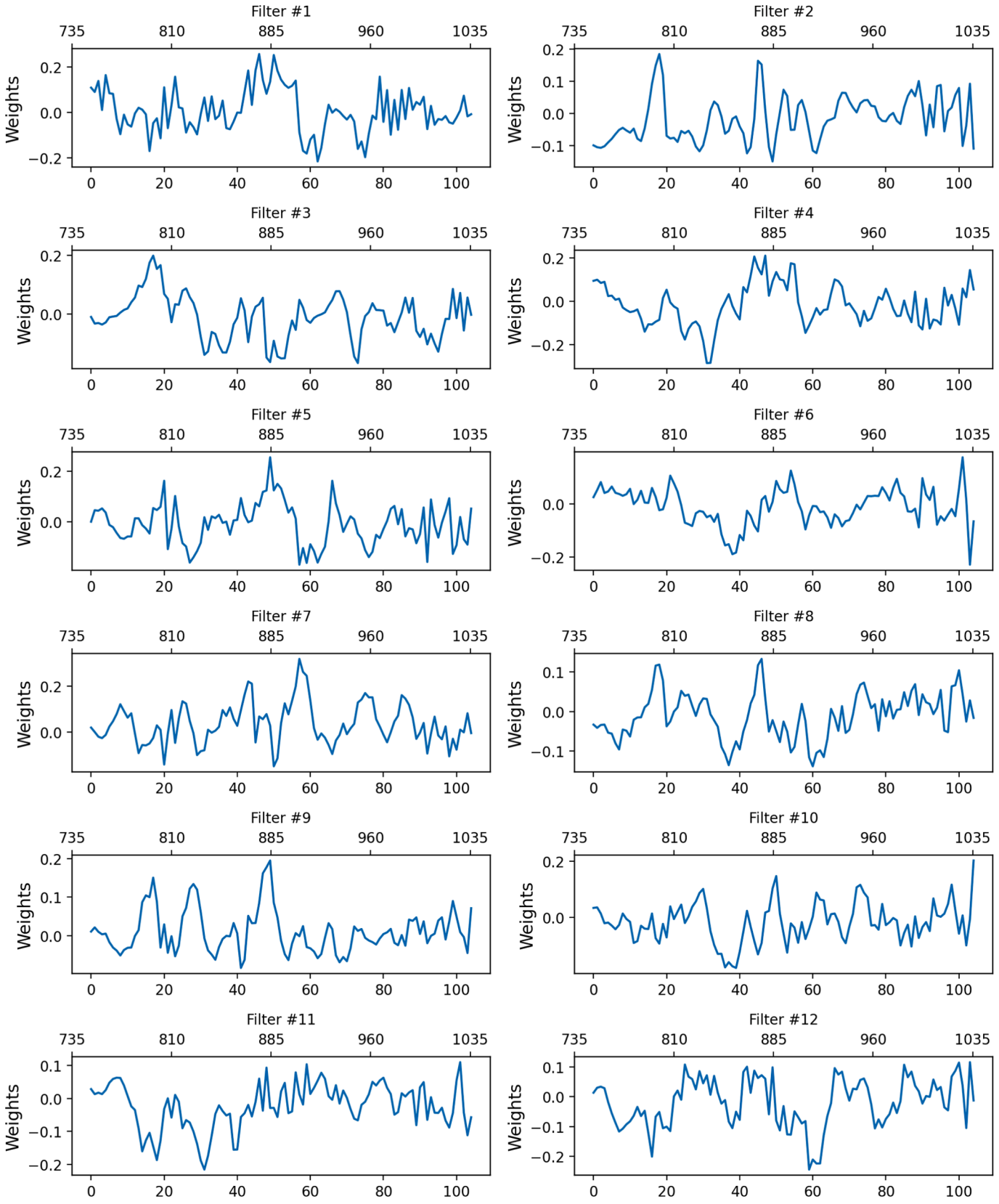


Fig. 7. The 12 (full width) filters of CNN_1E. Top XX scale represents wavelength and bottom XX scale represents feature number.

a result, this analysis shows that CNNs as chemometric models for dry matter prediction (and most probably for other regression tasks of the same kind), are indeed using information from known absorption bands and not random combinations of spurious features. The 3 model-agnostic methods (Regression Coefficients, LIME and SHAP) used to identify feature importance provide the same type of information. Due to its mathematical construction and natural units, SHAP explanations are the easiest to interpret (same units as the predicted variable). By looking at explanations SHAP explanations computed for individual fruit classes, one is able to identify how different spectral bands contribute to the overall prediction (in the context of a model trained as a global model applicable to different fruit classes).

Moreover, the SHAP explanations can be further used to identify which spectral bands are responsible for uncertainty on the predictions. To exemplify this use of the SHAP explanations, an additional experiment was conducted. Pre-trained models CNN_1B, CNN_1E and CNN_2C were used to predict the test set. The test RMSE was used as a threshold to select the 50 samples with lower RMSE and the 50 samples with higher RMSE. For each of these two subsets, the SHAP explanations were computed, and the results are shown in Fig. 8. The peak around 880 nm is the most impacted spectral region when low error and high error samples are compared. The worst predictions rely more heavily on features around this peak than the best predictions. Although this band seems important for prediction of DM, it is also a source of uncertainty in the predictions. The same thing happens, although in a less pronounced way, to a spectral band around 760 nm. This information can be leveraged in future works by actively developing architecture that suppress or "down weight" these bands.

Another interesting fact is that the CNNs SHAP profiles shown in Fig. 2 for multifruit data closely match the domain-invariant features between different pear batches identified by the di-CovSel variable selection method presented in [35] (see their Figs. 6 and 7). The di-CovSel algorithm is a novel variable selection methodology aimed at feature selection across different data domains that works by maximizing the covariance between the domains and the target variable while ensuring that these variables are stable across different domains. In [35], one of the examples the authors present is based on the same pear data set (two seasons) used in this work. In their case they used the two harvest seasons of the pear data set as two different domains and di-CovSel to select features that are both highly predictive and consistent across these domains. Using an analogous way of thinking, in this work CNNs find the most predictive features across different types of fruits (the different domains here). In this case, the use of a multifruit data set implies that during training, CNNs adjusts their weights to minimize prediction errors across all domains simultaneously. Features that are only predictive in specific domains but not on others, introduce inconsistencies and higher loss when predicting samples from different domains. Consequently, the optimization process implicitly favours features that have consistent relationships with the target variable across domains and reduce overall loss by being predictive in all domains. Since SHAP values quantify the contribution of each feature to the model's prediction, it ends up highlighting which features (wavelengths) are in fact domain-invariant.

The convolution filter analysis revealed that different architectures process the input spectra in different ways. In architectures with one convolutional layer only, the input spectra are transformed by the different filters. These transformations can be interpreted as a data-driven preprocessing method aimed at highlighting relevant features that minimize prediction error. In the case of models with 2 or more convolutional layers, the filters in the first layer seem to extract distinct patterns that are afterward combined in specific features maps. This is compatible with the classical view of the CNN block acting as a low-level extractor of features and their combination into more complex patterns. Unlike the usual case of CNN in 2D (for images), the filters widths picked by automatic hyperparameter optimization leans towards broad filters instead of narrow ones. We speculate that that is a consequence of the smooth nature of the NIR spectra caused by the overlap of several absorption bands. Only a broad filter can capture spectral patterns that span several tens of wavelengths, like the overlap of broad peaks or smooth spectral slopes. If this long-range dependency is in fact what the broad filters intend to capture, then the use of dilated convolutions should be interesting to explore.

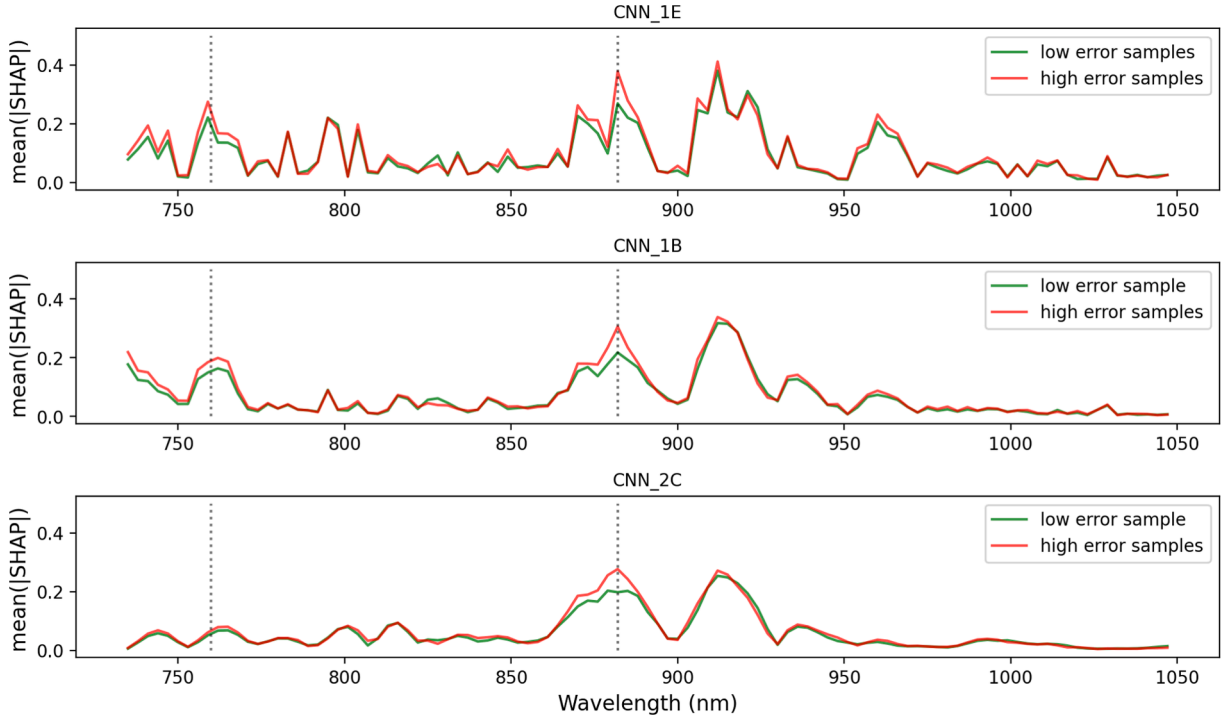


Fig. 8. *Mean(abs(SHAP))* values computed for the 50 samples of the test set with higher (red) and lower (green) RMSE as predicted by models CNN_1E, CNN_1B and CNN_2C. Vertical dotted lines indicate the 760 nm and 880 nm wavelengths.

In the case of architecture CNN_1E (full width, 'static' filters), the convolution operation is simplified, and the result is a feature dimensionality reduction from 105 to 12. This is the same concept used by PLS (projection into lower dimensional latent space) although driven by a different optimization objective. Moreover, CNNs use non-linear combinations of these low dimensional features for the final prediction. Due to these non-linear combinations of extracted features performed by the dense layers in the final block, a direct interpretation of convolutional layer activations (especially for models with more than 1 convolutional layer) is not straightforward.

5. Conclusions

In the field of food analysis using NIR spectra, with an emphasis on the prediction of fruit internal quality parameters, CNN chemometric models have been shown to provide state-of-the-art predictions. Nonetheless, adoption by the general community has been slow due to the "black box" or "inexplicability" connotation attributed to deep learning algorithms and the unavailability of a general CNN architecture for most chemometric tasks. CNNs provide SOTA results when they are properly initialized and optimized, which can be cumbersome for many applications due to computational resources needed and the degree of multidisciplinarity required. Despite this "slow start", a few hardware companies are already testing and using this type of algorithms in their commercial products. Also contributing to slow CNN adoption is the "good" performance of PLS(−like) models, its easiness of application and availability in multiple software packages.

The present work focused on presenting ways of understanding how different CNNs architectures work internally when applied to a classical chemometric regression task. Given the limited size of the data sets usually available in this field (with a few exceptions, e.g. Anderson et al 2020), most of the CNN chemometric models published can be considered shallow neural networks whose internal working mechanisms can be understood. Using the type of post-training analysis techniques presented here, a few lessons can be learned and synthesized into future works. While in the case of NIR spectra, physicochemical interpretation of features might still be difficult (due to the multiple wavebands overlaps and scattering effects), the method can be readily applied to other types of spectra (IR or RAMAN) where interpretability might be easier. Since the feature interpretation methods used here are model agnostic, it will be possible to apply them to future models that use deeper architectures. As experimentation with this type of model increases, and more lessons are learned, there is hope that the chemometrics community can converge towards a general CNN architecture suitable to most chemometric tasks.

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.

Acknowledgments

D. Passos acknowledges funding by projects UIDB/00631/2020 CEOT BASE, UIDP/00631/2020 CEOT PROGRAMÁTICO, FCT/RNCA project CPCA-IAC/AV/477942/2022 and FCT/RNCA project 2022.75263.CPCA.A0. The author also thanks P. Mishra for providing the multifruit data set and the SensorFINT COST Action (CA19145) for providing collaboration network support.

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.saa.2025.126068.

Data availability

Data will be made available on request.

References

[1] P. Mishra, D. Passos, F. Marini, J. Xu, J.M. Amigo, A.A. Gowen, J.J. Jansen, A. Biancolillo, J.M. Roger, D.N. Rutledge, A. Nordon, Deep learning for near-infrared spectral data modelling: Hypes and benefits, TrAC Trends Anal. Chem. 157 (2022) 116804, https://doi.org/10.1016/j.trac.2022.116804.
[2] J. Walsh, A. Neupane, M. Li, Evaluation of 1D convolutional neural network in estimation of mango dry matter content, Spectrochim. Acta Part A: Mol. Biomol. Spectrosc. 311 (2024) 1–15, https://doi.org/10.1016/j.saa.2024.124003.
[3] W. Jia, K. Georgouli, J. Martinez-Del Rincon, A. Koidis, Challenges in the Use of AI-Driven Non-Destructive Spectroscopic Tools for Rapid Food Analysis, Foods 13 (2024) 846, https://doi.org/10.3390/foods13060846.
[4] L.N. Xu, D.M. Xing, B. Yang, X. Sun, Principles and applications of convolutional neural network for spectral analysis in food quality evaluation: A review, J. Food Compos. Anal. 128 (2024) 105996, https://doi.org/10.1016/j.jfca.2024.105996.
[5] S. Wold, M. Sjöström, L. Eriksson, PLS-regression: a basic tool of chemometrics, Chemometr. Intell. Labor. Syst. 58 (2) (2001) 109–130, https://doi.org/10.1016/S0169-7439(01)00155-1.
[6] I.-G. Chong, C.-H. Jun, Performance of some variable selection methods when multicollinearity is present, Chemometr. Intell. Labor. Syst. 78 (1–2) (2005) 103–112, https://doi.org/10.1016/j.chemolab.2004.12.011.
[7] K.B. Walsh, M. Golic, C.V. Greensill, Sorting of fruit using near infrared spectroscopy: Application to a range of fruit and vegetables for soluble solids and dry matter content, J. Near Infrared Spectrosc. 12 (3) (2004) 141–148, https://doi.org/10.1255/jnirs.419.
[8] T.N. Tran, N.L. Afanador, L.M.C. Buydens, L. Blanchet, Interpretation of variable importance in Partial Least Squares with Significance Multivariate Correlation (sMC), Chemometr. Intell. Labor. Syst.: Int. J. Sponsored by the Chemometrics Society 138 (2014) 153–160, https://doi.org/10.1016/j.chemolab.2014.08.005.
[9] A.M. Cavaco, R. Pires, M.D. Antunes, T. Panagopoulos, A. Brázio, A.M. Afonso, R. Guerra, Validation of short wave near infrared calibration models for the quality and ripening of 'Newhall' orange on tree across years and orchards, Postharvest Biol. Technol. 141 (2018) 86–97, https://doi.org/10.1016/j.postharvbio.2018.03.013.
[10] A. Barredo Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, F. Herrera, Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI, Int. J. Inform. Fusion 58 (2020) 82–115, https://doi.org/10.1016/j.inffus.2019.12.012.
[11] Molnar, C. (2024). Interpretable machine learning: A guide for making black box models explainable. ISBN: 979-8423548189. Retrieved from https://christophm.github.io/interpretable-ml-book/.
[12] 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.
[13] Bjerrum, E. J., Glahder, M., & Skov, T. (2017). Data augmentation of spectral data for convolutional neural network (CNN) based deep chemometrics. Doi: 10.48550/ARXIV.1710.01927.
[14] C. Cui, T. Fearn, Modern practical convolutional neural networks for multivariate regression: Applications to NIR calibration, Chemometr. Intell. Labor. Syst. 182 (2018) 9–20, https://doi.org/10.1016/j.chemolab.2018.07.008.
[15] P. Mishra, D. Passos, A synergistic use of chemometrics and deep learning improved the predictive performance of near-infrared spectroscopy models for dry matter prediction in mango fruit, Chemometr. Intell. Labor. Syst. 212 (104287) (2021) 104287, https://doi.org/10.1016/j.chemolab.2021.104287.
[16] Zhou, B., Khosla, A., Lapedriza, A., Oliva, A. and Torralba, A. (2016). Learning Deep Features for Discriminative Localization. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, pp. 2921-2929. Doi: 10.1109/CVPR.2016.319.
[17] X. Zhang, J. Xu, J. Yang, L. Chen, H. Zhou, X. Liu, Y. Ying, Understanding the learning mechanism of convolutional neural networks in spectral analysis, Anal. Chim. Acta 1119 (2020) 41–51, https://doi.org/10.1016/j.aca.2020.03.055.
[18] L. Zhao, Y. Zeng, P. Liu, G. He, Band selection via explanations from convolutional neural networks, IEEE Access: Practical Innovations, Open Solutions 8 (2020) 56000–56014, https://doi.org/10.1109/access.2020.2981475.
[19] D. Passos, P. Mishra, An automated deep learning pipeline based on advanced optimisations for leveraging spectral classification modelling, Chemometr. Intell. Labor. Syst. 215 (104354) (2021) 104354, https://doi.org/10.1016/j.chemolab.2021.104354.
[20] 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: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 618–626.
[21] J.A. Martins, R. Guerra, R. Pires, M.D. Antunes, T. Panagopoulos, A. Brázio, A. M. Cavaco, SpectraNet–53: A deep residual learning architecture for predicting soluble solids content with VIS–NIR spectroscopy, Comput. Electron. Agric. 197 (106945) (2022) 106945, https://doi.org/10.1016/j.compag.2022.106945.
[22] R. Schiemer, M. Rüdt, J. Hubbuch, Generative data augmentation and automated optimization of convolutional neural networks for process monitoring, Front.

Bioeng. Biotechnol. 12 (2024) 1228846, https://doi.org/10.3389/fbioe.2024.1228846.
[23] D. Passos, P. Mishra, Deep Tutti Frutti: Exploring CNN architectures for dry matter content prediction in fruits, Chemometr. Intell. Labor. Syst. 243 (2023) 105023, https://doi.org/10.1016/j.chemolab.2023.105023.
[24] P. Mishra, E. Woltering, Handling batch-to-batch variability in portable NIR spectroscopy of fruit with deep learning domain adaptation, Anal. Chim. Acta 1181 (2021) 338771, https://doi.org/10.1016/j.aca.2021.338771.
[25] P. Mishra, D. Passos, Deep chemometrics: Validation and transfer of a global deep near-infrared fruit model to use it on a new portable instrument, J. Chemometr. 35 (10) (2021) e3367.
[26] McInnes, L., Healy, J., & Melville, J. (2018). UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. Doi: 10.48550/ARXIV.1802.03426.
[27] Ribeiro, M. T., Singh, S., Guestrin, C. (2016). "Why should I trust you?: Explaining the predictions of any classifier." Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. ACM (2016). Doi: 10.48550/arXiv.1602.04938.
[28] LIME: Explaining the predictions of any machine learning classifier. Retrieved July 8, 2024, from https://lime-ml.readthedocs.io/en/latest/index.html# and https://github.com/marcotcr/lime.
[29] SHAP (SHapley Additive exPlanations). Retrieved July 8, 2024, from https://shap.readthedocs.io/en/latest/# and https://github.com/lrjball/shap.
[30] Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. arXiv preprint arXiv:1705.07874. Retrieved from https://arxiv.org/abs/1705.07874.
[31] Sundararajan, M., Taly, A., & Yan, Q. (2017). Axiomatic attribution for deep networks. arXiv preprint arXiv:1703.01365. Retrieved from https://arxiv.org/abs/1703.01365.
[32] M. Golic, K. Walsh, P. Lawson, Short-wavelength near-infrared spectra of sucrose, glucose, and fructose with respect to sugar concentration and temperature, Appl. Spectrosc. 57 (2) (2003) 139–145, https://doi.org/10.1366/000370203321535033.
[33] H. Cen, Y. He, Theory and application of near infrared reflectance spectroscopy in determination of food quality, Trends Food Sci. Technol. 18 (2) (2007) 72–83, https://doi.org/10.1016/j.tifs.2006.09.003.
[34] Y. Shao, Y. He, Y. Bao, J. Mao, Near-infrared spectroscopy for classification of oranges and prediction of the sugar content, Int. J. Food Prop. 12 (2009) 644–658, https://doi.org/10.1080/10942910801992991.
[35] V.F. Diaz, P. Mishra, J.-M. Roger, W. Saeys, Domain invariant covariate selection (DI-CovSel) for selecting generalized features across domains, Chemometr. Intell. Labor. Syst. 222 (2022) 104499, https://doi.org/10.1016/j.chemolab.2022.104499.