



Machine Learning Applied for Spectra Classification in X-ray Free Electron Laser Sciences

RESEARCH PAPER

YUE SUN

SANDOR BROCKHAUSER

*Author affiliations can be found in the back matter of this article

]u[ubiquity press

ABSTRACT

Spectroscopy experiment techniques are widely used and produce a huge amount of data especially in facilities with very high repetition rates. At the European XFEL, X-ray pulses can be generated with only 220ns separation in time and a maximum of 27000 pulses per second. In experiments at the different scientific instruments, spectral changes can indicate the change of the system under investigation and so the progress of the experiment. Immediate feedback on the actual state (e.g. time-resolved status of the sample) would be essential to quickly judge how to proceed with the experiment. Hence, we aim to capture two major spectral changes. These are the change of intensity distribution (e.g. drop or appearance) of peaks at certain locations, and the shift of the peaks in the spectrum. Machine Learning (ML) opens up new avenues for data-driven analysis in spectroscopy by offering the possibility for quickly recognizing such specific changes and implementing an online feedback system which can be used near real-time during data collection.

On the other hand, ML requires lots of data that are clearly annotated. Hence, it is important that experimental data should be managed along the FAIR principles. In the case of XFEL experiments, we suggest introducing NeXus glossary and the corresponding data format standards for future experiments.

An example is presented to demonstrate how Neural Network-based ML can be used for accurately classifying the state of an experiment if properly annotated data is provided.

CORRESPONDING AUTHOR:

Sandor Brockhauser

Institute of Informatics,
University of Szeged, Szeged,
Hungary; Open Scientific
Computing Group, European
XFEL GmbH, Schenefeld,
Germany

sandor.brockhauser@gmail.com

KEYWORDS:

data model; spectral data
classification; neural network;
weighting technique

TO CITE THIS ARTICLE:

Sun, Y and Brockhauser, S.
2022. Machine Learning
Applied for Spectra
Classification in X-ray Free
Electron Laser Sciences.
Data Science Journal, 21: 15,
pp. 1-18. DOI: <https://doi.org/10.5334/dsj-2022-015>

1 INTRODUCTION AND BACKGROUND

1.1 SPECTROSCOPY TECHNIQUES AT EUROPEAN XFEL

Spectroscopy is central to the natural sciences and engineering as one of the primary methods to investigate the real world, study the laws of nature, discover new phenomena, and characterize the properties of substances or materials (Ghosh et al. 2019).

However, experiments performed at photon science instruments in large facilities, like synchrotrons or XFELs always generate a large volume of data. As an advanced source, European XFEL is providing new research opportunities to scientists from different domains as diverse as physics, chemistry, geo- and planetary sciences, materials sciences or biology (Tschentscher et al. 2017). The unique feature of the European XFEL is the capability of the superconducting linac to accelerate trains of many (up to 2,700) electron bunches within one 600- μ s-long RF pulse at 4.5MHz (Decking et al. 2020) with an electron energy of up to 17.5 GeV, which means that X-ray pulses can be generated with only 220ns separation in time and a maximum of 27000 pulses per second. Combining them with fast 2D detectors, a huge amount of data is generated in a short time (Bernstein et al. 2020).

At the European XFEL, currently, six scientific instruments (e.g. SCS (Le Guyader et al. 2020; Mercurio et al. 2019), FXE (Zalden et al. 2019; Galler et al. 2019), and HED (Nakatsutsumi et al. 2014)) are in operation for specific applications using soft or hard X-ray techniques (Tschentscher et al. 2017). Among them, the High Energy Density (HED) scientific instrument focuses on the investigation of matter at high density, temperature, pressure, electric, and/or magnetic field (Nakatsutsumi et al. 2014). It offers a wide range of time-resolved X-ray techniques reaching from diffraction and imaging to different spectroscopy techniques for measuring various geometric, electronic and magnetic structural properties (Tschentscher et al. 2017).

In HED experiments with high-density materials, changes in pressure will cause changes in the measured spectral peaks (making them newly appear, vanish, shift, or split). In order to evaluate the state of the experiment, the measured spectra need to be classified so that each class is assigned to a different state of the system under investigation. The two major spectral changes that we aim to capture in this study are

- the change of intensity distribution (e.g. drop or appearance) of peaks at certain locations, or
- the shift of those in the spectrum.

The complexity of the experimental setup and the high data rates at the order of 10–15GB per second per detector at European XFEL (Kuster et al. 2014) demand an efficient approach of combining experiments with online data analysis as an active feedback system with low feedback latency (Fangohr et al. 2017; Brockhauser et al. 2013; Brockhauser et al. 2012; Basham et al. 2015).

Under such circumstances, it is crucial to properly use and reuse the previously collected big data, and also efficiently analyze, and accurately extract meaningful features during experiments. While data analysis algorithms that properly solve the relevant physics equations and optimization steps would be hopeless to perform in near real-time, Machine Learning based algorithms can provide a solution.

1.2 MACHINE LEARNING AND ITS REQUIREMENTS

Machine learning (ML) is a subfield of artificial intelligence centered on algorithms that can learn to complete tasks using data (Edelen et al. 2018). It applies broadly to a collection of computational algorithms and techniques that train systems from raw data rather than a priori models (Edelen et al. 2018), thus useful for research facilities that produce large, multidimensional datasets like European XFEL. With recent developments in ML, data-driven deep learning methods have turned out to be very good at discovering patterns and structures in high-dimensional data (LeCun, Bengio & Hinton 2015) during the training process. The trained models can then supply computational inexpensive decisions when applied on new data. Hence, this technique opens up new ways for data-driven analysis in spectroscopy by offering the possibility of recognizing specific features on-the-fly during data collection. It enables fast feedback and the implementation of near real-time downstream

data analysis, e.g. providing immediate information on the phase of the experiment based on the classification of the currently measured spectra. In order to train the model more effectively and accurately, ML requires lots of data which can only be supplied on a machine readable way if the experimental dataset are stored in a standard and well understandable format with clear definitions and annotations. It is important that research outputs should align with the **FAIR** principles, meaning that data, software, models, and other outputs should be Findable, Accessible, Interoperable, and Reusable ([Wilkinson et al. 2016](#)). The key in finding and re-using digital assets is metadata ([RDA COVID-19 Working Group, 2020](#)). For data re-use, their annotation and assignment to metadata with ontology definitions are very important. In the case of synchrotron or XFEL experiments, data collection and management processes of the experiment control systems shall be prepared to acquire all necessary metadata in a hierarchical data structure following a scientific experiment data model. Such an experimental data model shall connect the instrumentation and data acquisition to the physics model behind the respective experiment. The NeXus data format and ontology platform ([Donath, Rissi & Billich 2013](#); [Hester 2016](#); [Könnecke et al. 2015](#)) provides an ideal environment for this.

Whether the experiment is performed in a synchrotron or an XFEL facility, the corresponding ML algorithms could obtain the information always in the same machine readable format, and could easily and automatically pick up the data source to be processed during training. This requires all experiment-related metadata including those needed for processing, as well as those needed for filtering the available datasets to be captured and stored according to the ontology definitions and relationships set by the data model of the respective experiment type.

Unfortunately, in the case of XFEL sciences, the scientific data are not well annotated, and many key information are not machine-findable or accessible. Specifically, in our use case, the HED experimental spectra data we obtained lacked some key information, such as the pressure values and were provided with wrong metadata e.g. on the energy of the incident beam. Therefore, the selection and the preparation of the training dataset for our ML model required lots of manual work.

1.3 SCIENTIFIC EXPERIMENT DATA MODEL

To comply with FAIR principles and facilitate ML training, experimental data shall follow experiment data models which connect the experimental data items to the physics model used by the design of the experiment and also by the corresponding data processing applications. As the physics models and data processing applications can be different from experiment to experiment, the corresponding data models need to be defined for each experiment type respectively. Similarly to physics models which can be extended and reused during the investigation of related physics phenomena, scientific experiment data models shall also be modular and shall reuse base definitions.

The NeXus glossary defined using the NeXus Definition Language (NXDL) offers several base classes and allows annotating any data items stored in the NeXus format by connecting them the glossary definitions. NXDL supports modular extension of these base classes and declaration of application definitions. These form complete data models covering all the requirements of data processing applications for the respective experiment types. NXDL also provides an ontology which is based on the relationships declared between the defined glossary terms. Hence, scientists can define a schema for an application data model which connects all data fields in a NeXus data file to its machine-readable meaning ([Könnecke et al. 2015](#)). In this way, an experiment model can be expressed in an NXDL application definition and all the corresponding data and metadata can be stored in a file following the NeXus data format standard.

A good example for an application definition specifying a data model for a given scientific domain and so supporting data reuse is NXmx (NeXus for macromolecular crystallography), one of several standardized NeXus application definitions ([Bernstein et al. 2020](#)). Recording data and metadata in such an annotated data exchange format can guarantee automatic and proper data interpretation and support data reuse.

1.4 SIGNIFICANCE OF THE STUDY

As described above, it is important to optimize the usability and reusability of the scientific data collected during XFEL experiments. A near real-time and accurate extraction of meaningful features during the experiments is vital for feedback systems, and essential for assessing the state of the experiments. The main contributions of this study are two-fold:

Firstly, we show an example how ML-based artificial intelligence (AI) methods can provide near real-time feedback on the status of the experiments and trigger a conceptual breakthrough towards data-driven spectroscopy ([Ghosh et al. 2019](#)). For this purpose, this study demonstrates a machine learning based statistical model which is applied for spectra classification. The presented solution automatically finds the regions (or bins) where the classes differ significantly in our training set (that is what we call separability). For this step, a two-layer neural network is applied to train our classification model of each bin separately and the obtained separability is used to calculate classification weighting factors for each bin respectively. By calculating the weighted sum of the ML based predictions obtained for the points of a spectrum (corresponding to the diffractogram in a certain experimental state), a class label is assigned finally which suggests the actual state of the experiment. We also investigated the performance of the model using different number of bins. We have found a relationship between the number of bins and the size of the ambiguous region in which the prediction of the model is not reliable. As a result an optimal number of bins can be determined as a compromise between the computational complexity and an expectable classification confidence.

Secondly, we highlight the importance of FAIR data management in XFEL experiments, and suggest the implementation of individual NeXus application definitions for the different experiment types. Specifically, for future HED spectral data, we recommend introducing a specific application definition which allows a machine-readable and machine-interpretable registration of important metadata, like the pressure and wavelength values alongside the raw detector data.

2 EXPERIMENT DATA

Spectral data was collected during an HED experiment performed at PETRA III Extreme Conditions Beamline P02.2 using the LAMBDA GaAs 2M detector system (a multi-megapixel hard X-ray detector to measure the scattering intensity for experiments at synchrotrons), as described in ([Pennicard et al. 2018](#)). A simple sketch of the experiment setup is shown in [Figure 1](#). The frame rate of the detector enables measurements up to 2000 frames per second, giving the possibility of 0.5 ms period per image. The sample used in this study was $(\text{Mg}_{0.2}\text{Fe}_{0.8})\text{O}$ magnesiowüstite which is a solid solution of the endmembers periclase (MgO)

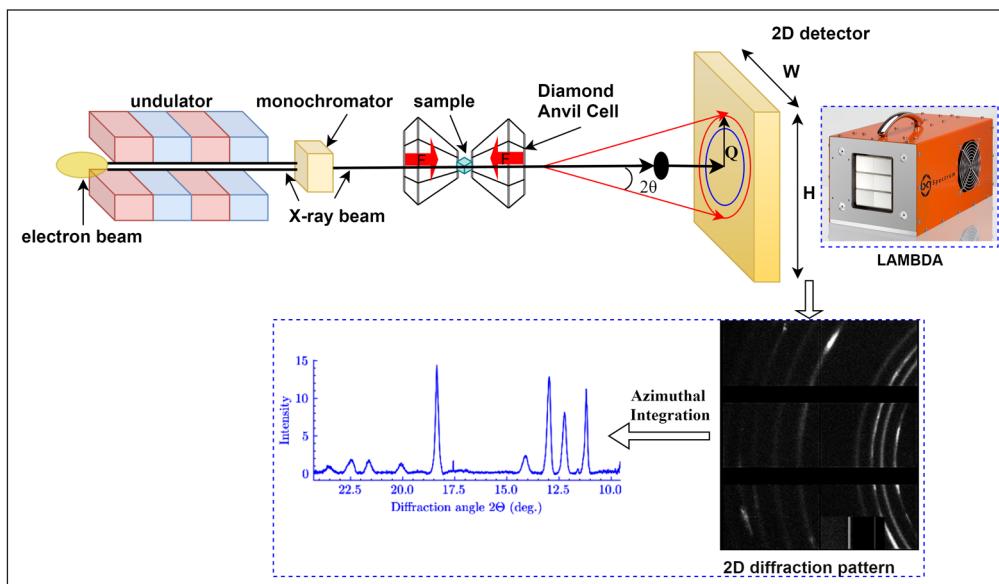


Figure 1 Schematic diagram of the High Energy Density (HED) experiment setup.

and wüstite (FeO). The sample was compressed from 1 to above 100 GPa over a period of 50 s while maintaining reasonable diffraction image quality in consecutive 100ms exposures. The measurement was performed at 25.6 keV beam energy (corresponding to the incident beam wavelength of 0.4828 Å). Diffraction images (as in [Figure 2](#)) were continuously collected every 100 ms during a trapezoidal ramp profile, tracking the pressure evolution of the sample and the Pt (platinum) pressure calibrant. Azimuthal integration and baseline subtraction ([Filik et al. 2017](#)) are applied to 2D raw diffraction images to obtain 1D integrated spectra. [Figure 2](#) shows an example of how the spectral peak positions in a spectral curve reflect the intensity distribution in the corresponding raw diffraction image as a function of the scattering angle. A peak in the diffractogram or spectrum corresponds to a diffraction ring ([Pennicard et al. 2018](#)).

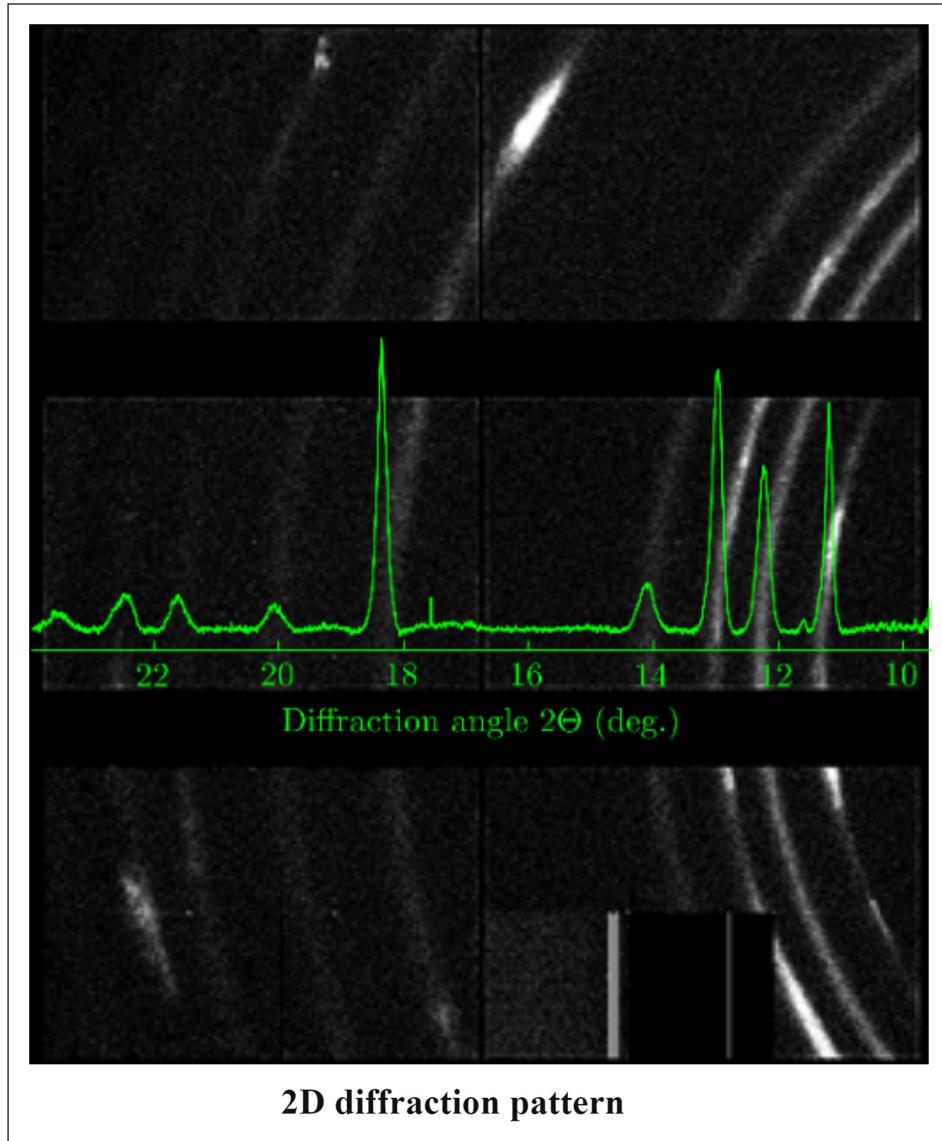
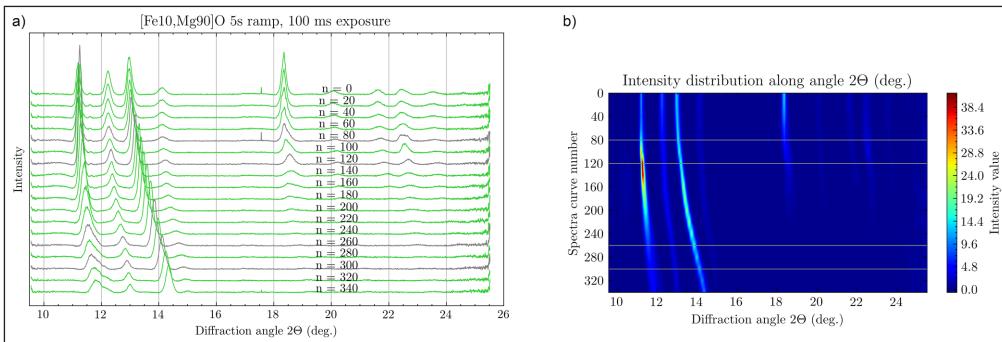


Figure 2 Raw X-ray diffraction image from a $(\text{Mg}_{0.2}\text{Fe}_{0.8})\text{O}$ magnesiowüstite sample collected with the LAMBDA GaAs 2M detector. This figure was measured at PETRA III beamline P02.2 using a 25.6 keV beam. The 2D raw diffraction image shows the starting condition at 1 GPa and with fcc/B1 ($\text{Mg}_{0.2}\text{Fe}_{0.8}\text{O}$) sample. After azimuthal integration, the corresponding 1D spectrum is shown in green on top of it.

The obtained spectral data collected during the experiment is shown in [Figure 3](#). To show more clearly how the diffraction changes while the pressure on the sample is changing, we show one for every 20 diffractograms. Hence, the time interval between adjacent diffractograms in the [Figure 3](#) (a) is 2 s. As it can be clearly seen from this figure, the amplitude of the spectral peaks change (increase, decrease, or even vanish) at certain locations, and peaks also shift in their 2θ -angle position, split, or start to broaden.

The changes correspond to the modification of the crystal lattice (e.g. indicating phase changes). During the experiment, we should be able to follow these changes and determine the actual state of the system in near real-time. Scientifically, the most relevant question is whether the phase transition in the sample has already happened. To determine this using Machine Learning, we selected representative measured spectra (and also produced simulated spectra) at both the initial and final stages. Based on this input, we have to provide a judgment at each point during the experiment with the minimum ambiguity.



In this experiment, the most important external condition that causes phase transition is the change of pressure. One of the Raw data files produced during the HED experiment is shown in Figure 4. It recorded data from one module of the three-module detector system and stored it in hdf5 format. Note that NeXus base classes were used by the facility to store the data, but no application definition was followed which would define where to find certain experiment settings as metadata for this experiment type (e.g. pressure, or incident X-ray beam energy). The root element, an NXentry group represents one experimental scan or run (Könnecke et al. 2015). Some instrument settings can also be retrieved from the base class definitions as of NXdetector, for example, detector acquisition mode, calibration data, geometry, and translation information, but lots of metadata including the incident beam energy appears only as free text key-value pairs under the group NXcollection without any definition and link to a physics model. As a consequence, no data processing application can rely on the generated file to automatically find these settings and use them. Instead, beamline scientists can help with a manual interpretation of the dataset. Even more, it has turned out that the free-text key-value pair provided for human interpretation “beam energy – 25000” is misleading, as the incident beam energy was actually 25.6keV and not 25000[eV] as suggested by the certainly unmaintained and undocumented entry above provided with no units. Note that without knowing the beam energy, no conversion between diffraction angle and Q-space is possible which blocks the possibility of proper calibration and determination of key information such as the pressure corresponding to each diffractogram. As a result, we cannot automatically select the training set for our ML model, but rather picked up spectra for the training manually.

Figure 3 (a) Spectral data (one for every 20 diffractograms) collected during the experiment after baseline subtraction, and (b) Contour maps of Intensity distribution of each diffractogram along 2θ axis. Note that 4 spectra are grayed out which were randomly picked as the basis for the ML training set for representing the two experimental states (before and after phase transition).

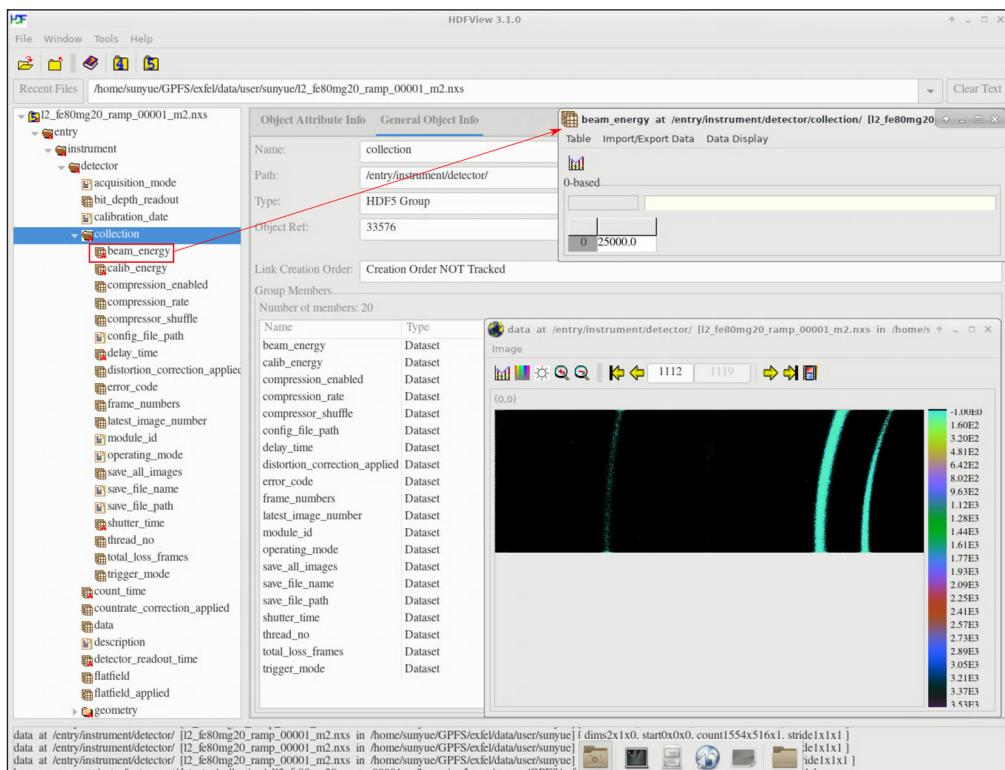


Figure 4 An hdf5 experimental raw data file is displayed in HDFView software.

3 EXAMPLE FOR DATA REUSE

Machine learning with its big data demand provides a perfect use case for data reuse. In ML, a classification algorithm is usually considered a type of supervised learning, which is used when the outputs are restricted to a limited set of categories or values.

In the case of supervised learning, the training process will tweak the parameters of the model for correcting the output towards the one defined as a good solution. It infers a function from a labeled training dataset (Mohri, Rostamizadeh & Talwalkar 2018). The trained mathematical model can then be used as a classifier to predict the class of previously unseen instances. Hence, after training the model using spectral data belonging to either the state before or after the phase transition, the neural network can classify each newly acquired spectrum during data collection and determine the actual state of the system under investigation during the experiment.

3.1 MATHEMATICAL MODEL APPLIED FOR MACHINE LEARNING

A. Notation and terminology

In this study, scalar value variables are denoted as normal font letters (i.e. x or X), matrices are denoted as bold capitals (i.e. \mathbf{X} or \mathbf{W}), vectors are denoted as bold lowercase letters (i.e. \mathbf{x}). Variables with subscripts represent elements in a vector or matrix, for example, we use x_i and x_{ij} to indicate the element in vector \mathbf{x} and matrix \mathbf{X} respectively. The transpose of a matrix or vector is represented by the symbol $(\cdot)^T$, and $(\cdot)^{(i)}$ is associated with the i -th data item in the training set.

B. Supervised learning for classification

Given the inputs, $\mathbf{X} \in \mathbb{R}^{n \times m}$ representing m training samples (or observations) with n data points in each can be denoted as

$$\mathbf{X} = \left[\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots, \mathbf{x}^{(m)} \right] \quad (1)$$

The training samples are stacked up in the matrix. Each sample has n data points or features, which is often called *feature vector*. The i -th training example can be denoted as

$$\mathbf{x}^{(i)} = \left[x_1^{(i)}, x_2^{(i)}, \dots, x_n^{(i)} \right]^T, \quad i = 1, \dots, m \quad (2)$$

The corresponding output or classification response $\mathbf{y} \in \mathbb{R}^{1 \times m}$ is the class labels or target values for \mathbf{X} , represented as

$$\mathbf{y} = \left[y^{(1)}, y^{(2)}, \dots, y^{(m)} \right] \quad (3)$$

where each element $y^{(i)}$ in the output vector \mathbf{y} corresponds to the input vector $\mathbf{x}^{(i)}$. And the input-output pairs $(\mathbf{x}^{(i)}, y^{(i)})$ constitute the training data set, S ,

$$S = \left\{ (\mathbf{x}^{(1)}, y^{(1)}), (\mathbf{x}^{(2)}, y^{(2)}), \dots, (\mathbf{x}^{(m)}, y^{(m)}) \right\} \quad (4)$$

The function f to convert from \mathbf{X} to \mathbf{y} is the model that the ML algorithm needs to learn to accurately predict the outcomes. In most cases, the observation data \mathbf{X} contains noise and may also contain contradictions. Hence, it is usually impossible to find a function that can accurately classify all training samples. The goal of ML is to develop a model f that can best match each training example $(\mathbf{x}^{(i)}, y^{(i)})$, so that f can be used as a predictor for new data or observations $\mathbf{x}^{(new)}$. Modeling the inaccuracy as noise, the generic data model for ML can be described as

$$\mathbf{y} = f(\mathbf{X}) + \varepsilon \quad (5)$$

where ε represents discrepancies such as measurement errors or other inconsistencies.

It yields the prediction for new data $\mathbf{x}^{(new)}$ as

$$y^{(new)} \approx \hat{y}^{(new)} = f(\mathbf{x}^{(new)}) \quad (6)$$

During the process of training the model, its parameters (which can be expressed as parameter vector $\theta \in \mathbb{R}^n$) are adjusted. Therefore, the vector may be estimated by solving a (convex) optimization problem (Kulin et al. 2018).

A loss (or cost) function $l(\cdot)$ is used to evaluate the performance of a neural network by calculating the error between the actual value $y^{(i)}$ and the predicted value $\hat{y}^{(i)}$. It is a very important indicator for monitoring the training process. Cross-entropy loss is the most commonly used loss function for multi-class classification problems, expressed as

$$l(\theta) = -\sum_{k=0}^{C-1} y_k^{(i)} \log \hat{y}_k^{(i)} \quad (7)$$

where C is the number of classes, $\hat{y}_k^{(i)}$ and $y_k^{(i)}$ are the predicted probability if the i -th data point belongs to the class k and the One-Hot Encoded ground truth respectively.

Averaging the loss over the whole training set S yields to the training loss J as

$$\begin{aligned} J(\theta) &= \frac{1}{m} \sum_{\{\mathbf{x}^{(i)}, y^{(i)}\} \in S} l(\mathbf{x}^{(i)}, y^{(i)}, \theta) \\ &= -\frac{1}{m} \sum_{\{\mathbf{x}^{(i)}, y^{(i)}\} \in S} \sum_{k=0}^{C-1} y_k^{(i)} \log \hat{y}_k^{(i)} \end{aligned} \quad (8)$$

The goal of training the neural network is to minimize the value of this loss function. Therefore, the process of model training is transformed into the process of solving the minimum value of the training loss $J(\theta)$, which is described as

$$\underset{\theta \in \mathbb{R}^n}{\operatorname{argmin}} J(\theta) \quad (9)$$

After this calculation process, the best mathematical parameters of the function f on the training set S will be obtained, and then the output y for any given input \mathbf{x} can be predicted by the formula $\hat{y} = \theta^T \mathbf{x}$.

C. Deep learning

The performance of most of the traditional ML algorithms depends on how accurately the features are identified and extracted (Suganya & Charles 2019). Deep learning is a branch of ML algorithms (Deng & Yu 2014) which offers an efficient method for learning features via applying multiple layers of neural networks which extract different features at each layer. The initial data is provided to the network in its visible input layer, while the different abstract features are generated in the subsequent hidden layers. Hence, a hierarchical feature extraction derives here finally an abstract representation.

1) Fully connected neural networks

With the advantage of being “structure agnostic”, fully connected networks (as shown in Figure 5) are capable of learning any function which boosts them to become a universal learning architecture, and are regarded as the workhorses of deep learning.

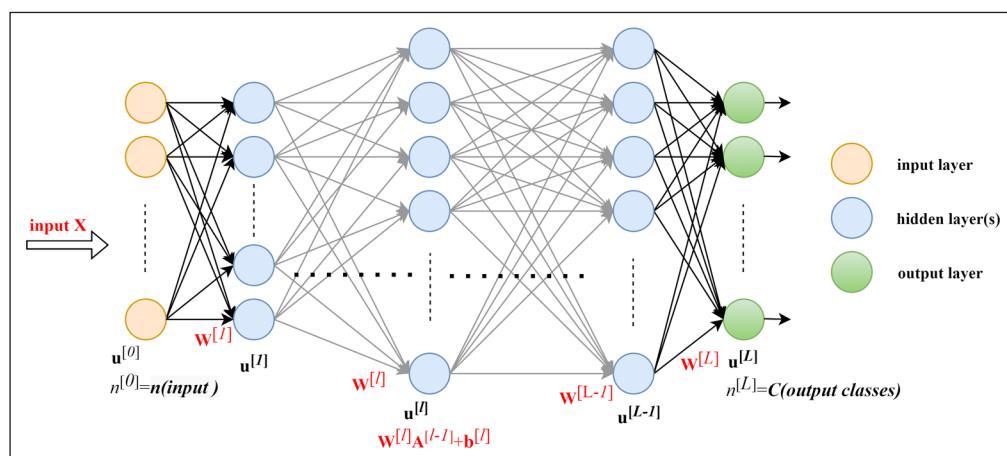


Figure 5 A multilayer deep fully connected network.

A fully connected neural network consists of a series of fully connected layers (Ramsundar & Zadeh 2018), which is a mathematic function that describes its input-output relationship. Consider the case of one training example $x^{(i)}$ for any layer l the relationship between its input and output can be described as

$$a_j^{[l](i)} = \sigma(\mathbf{w}_j^{[l]} \cdot \mathbf{a}^{[l-1](i)} + b_j^{[l]}), \quad j = 1, \dots, n^{[l]} \quad (10)$$

where $a_j^{[l](i)}$ represents the output of the j -th neuron $u_j^{[l]}$, $\mathbf{a}^{[l-1](i)} \in \mathbb{R}^{n^{[l-1]}}$ represents the input vector from the previous layer $l-1$, $\mathbf{w}_j^{[l]}$ is the weight vector associated with inputs of neuron $u_j^{[l]}$, $b_j^{[l]}$ is the bias associated with neuron $u_j^{[l]}$, $n^{[l]}$ is the number of neurons in layer l , and $\sigma(\cdot)$ is an activation function as explained below.

Consider a standard L -layer fully-connected neural network with all the training examples X we can denote the input layer as $\mathbf{A}^{[0]} \in \mathbb{R}^{n \times m} (\mathbf{A}^{[0]} = X)$, the outputs of the hidden layer l as $\mathbf{A}^{[l]} \in \mathbb{R}^{n^{[l]} \times m}$, $l=1, \dots, L-1$, the very last, so called output layer as $\mathbf{A}^{[L]} \in \mathbb{R}^{n^{[L]} \times m}$. At each layer, weights are introduced as $\mathbf{W}^{[l]} \in \mathbb{R}^{n^{[l]} \times n^{[l-1]}}$, $l=1, \dots, L$, together with bias vectors $\mathbf{b}^{[l]} \in \mathbb{R}^{n^{[l]}}$, $l=1, \dots, L$ which yield to the generic calculation model in the network:

$$\mathbf{A}^{[l]} = \begin{cases} h(\mathbf{W}^{[l]} \mathbf{A}^{[l-1]} + \mathbf{b}^{[l]}) = h(\mathbf{Z}^{[l]}), & l=1, \dots, L-1 \\ g(\mathbf{W}^{[l]} \mathbf{A}^{[l-1]} + \mathbf{b}^{[l]}) = g(\mathbf{Z}^{[l]}), & l=L \end{cases} \quad (11)$$

where $h(\cdot)$ and $g(\cdot)$ are some nonlinear activation functions. The commonly used activation functions for hidden layers are ReLU (Nair & Hinton 2010), sigmoid function (Han & Claudio 1995), and hyperbolic tangent (Namin et al. 2009). In our study, we use ReLU, which can greatly accelerate the learning speed of the classification model. For the output layer, $g(\cdot)$ always represents the softmax function (Bishop 2006), which provides the probability distribution of each outcome or label over C classes $\hat{y}_k^{(i)}$, defined by the formula

$$\hat{y}_k^{(i)} = \frac{e^{z_k^{(i)}}}{\sum_{j=0}^{C-1} e^{z_j^{(i)}}}, \quad k = 0, \dots, C-1 \quad (12)$$

where $z_k^{(i)}$ (also called logits) represents the score of the i -th data point belonging to class k as obtained in the output layer. Note that in our classification model the number of neurons in the last layer (layer L) is equal to the number of classes ($n^{[L]} = C$). The predicted class $\hat{C}^{(i)}$ (label) for the i -th data point is obtained as

$$\begin{aligned} \hat{C}^{(i)} &= \operatorname{argmax}_k \hat{y}_k^{(i)} \\ &= \operatorname{argmax}_k \frac{e^{z_k^{(i)}}}{\sum_{j=0}^{C-1} e^{z_j^{(i)}}} \\ &= \operatorname{argmax}_k e^{z_k^{(i)}}, \quad k = 0, \dots, C-1 \end{aligned} \quad (13)$$

Unfortunately, parameter vector θ in deep fully connected neural network models (composed of $\mathbf{W}^{[l]}$ and $\mathbf{b}^{[l]}$ for layer l) makes the computational complexity grow exponentially with the number of layers, and at the same time slows down the training process and increases the chance of overfitting, so the number of such layers is limited in practice.

2) Gradient descent optimization algorithm

Gradient descent is one of the most popular algorithms to perform optimization and by far the most common way to optimize neural networks (Ruder 2016). Learning rate is a parameter of the training process, which determines the step size to reach the local minimum of the objective function J (equation (8)). Choosing its proper value can be difficult. A learning rate that is too small will lead to very slow convergence, while a learning rate that is too large can hinder convergence and cause the loss function to fluctuate around the minimum or even diverge (Ruder 2016).

To overcome this, the Adaptive Moment Estimation method (Adam) ([Kingma & Ba 2014](#)) computes adaptive learning rates for each parameter and also adds bias correction and momentum to RMSprop, the Root Mean Square Propagation algorithm, a similar but simpler algorithm ([Hinton, Srivastava & Swersky 2012](#)). While Adam, RMSprop and also Adadelta ([Zeiler 2012](#)), an extension of Adagrad ([Duchi 2011](#)), are very similar algorithms, Kingma et al. ([2014](#)) show that Adam carries the benefits of all, and it can actually slightly outperform towards the end of optimization as gradients become sparser ([Ruder 2016](#)).

Using Adam as the optimizer, the learning rate doesn't need to be manually adjusted at different stages, but only a default value has to be provided.

3.2 SPECTRA CLASSIFICATION

In this work, we give an example of using experimental data to determine if our multi-crystal powder system has already passed a phase transition during a pressure ramping experiment protocol or not.

Since pressure was not recorded directly with the raw dataset, a training set containing elements from both phases could not be automatically identified, but rather the representative curves had to be selected manually. For this purpose, four representative spectral curves (2 from both classes) are included in the training set (as shown in [Figure 3](#)). To learn the characteristic differences between the 2 classes, Neural Network-based ML is used. To increase the robustness of the neural network model and its tolerance to small Gaussian measurement errors, 10 simulated spectral curves for each original spectrum were also included in the training set by adding some random noise, which is small enough to enable finding the characteristic features in the spectra. The random noise is generated using the Mersenne Twister ([Matsumoto & Nishimura 1998](#)) as the core generator. The spectral data used for the training can be seen in [Figure 7](#).

As described below, with the introduction of a new metric *separability* learnt from the final accuracy of the trained neural network model in each bin, we assign *believability weighting* factors to the bins to describe how much the classification prediction of the ML model in the given bin can be trusted. A weighted average of the individual predictions from all bins yields the final classification prediction for any new spectra.

A. Neural Network Structure

We choose neural networks due to their ability to learn complex mappings between input and target spaces which makes them perfect for our task. Neural network models have gained increased popularity recently, since they can express complex function mappings using inputs with very little or no feature engineering ([Ghosh et al. 2019](#)). In this study, we explored a two-layered neural network architecture with 150 hidden units. Each layer accepts the output of the previous layer as its input, and returns a transformation function as its output ([Ghosh et al. 2019](#)). We transform our problem of finding and distinguishing features in a set of 1D curves where the input is provided by the sequence of the spectral intensity values, into a 2D segmentation problem where we input every point in the spectra with their 2 coordinates (scattering angle, and azimuthally integrated intensity as in [Figure 2](#)). Hence, our input layer has 2 input neurons (corresponding to the coordinates of the given points in the 2D space) and the hidden layer has 150 hidden neurons (as an abstract feature map) followed by a non-linear ReLU function. Their outputs are fed into the second layer (which is the output layer) with 2 output neurons (corresponding to the number of classes being measured before or after the phase transition), followed by a softmax activation function, as in [Figure 6](#). The ReLU activation units are selected here to speed up the model learning and prevent gradient vanishing/exploding ([Yang et al. 2019](#)).

Such a neural network described above can be applied for the general 2D space segmentation problem as discussed in ([Singh 2001](#)). Here, we investigate the application of the same NN architecture for the special case of real experimental spectra classification.

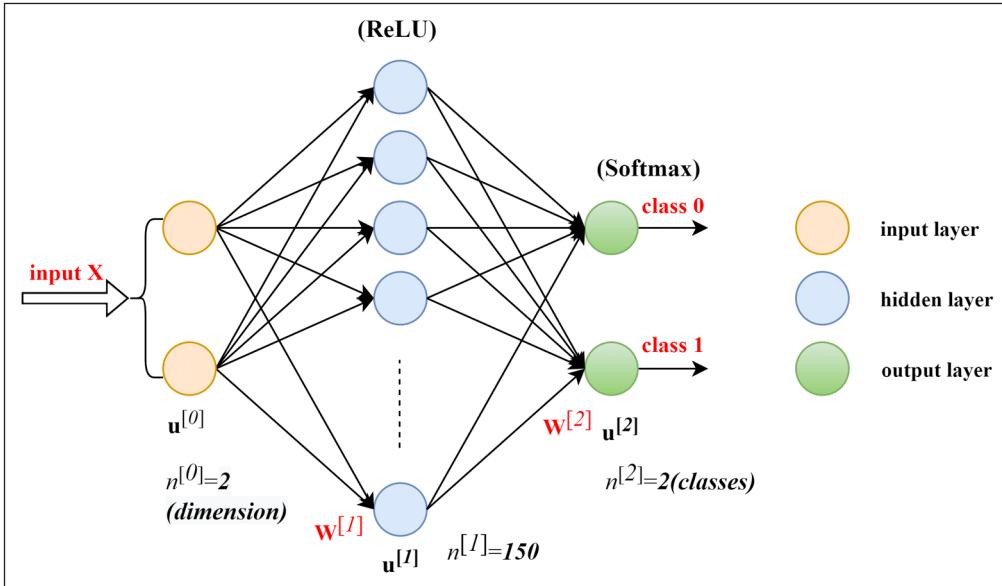


Figure 6 Two-layered neural network structure used for spectra classification.

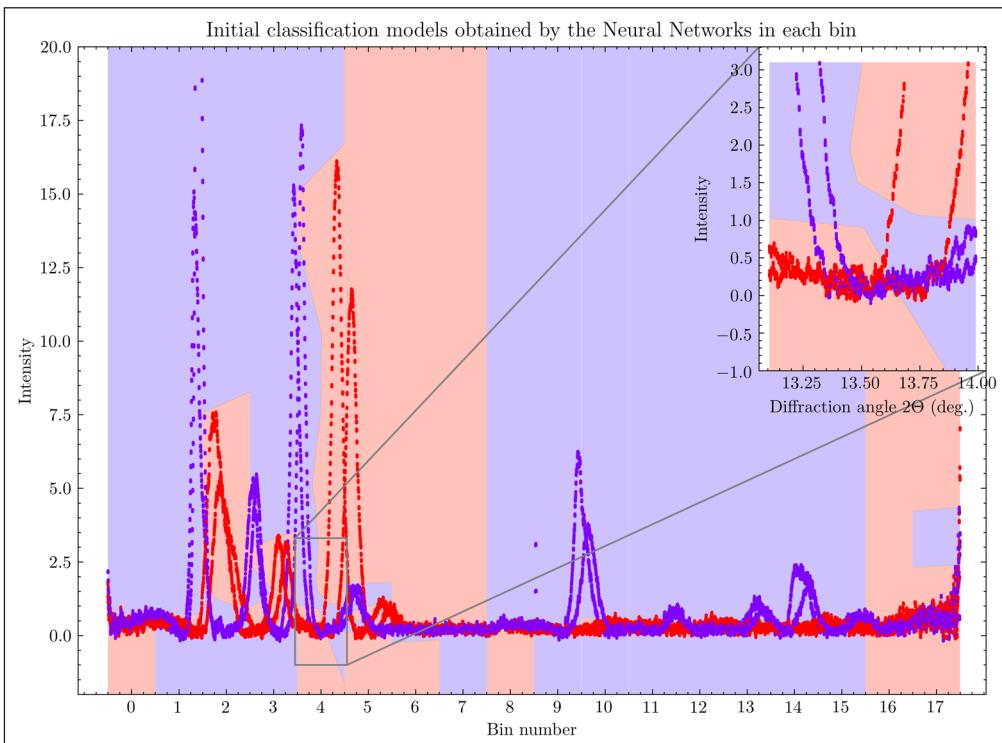


Figure 7 The initial classification model for each bin obtained by the respective trained two-layered network models.

B. Model Training and believability weighting factors

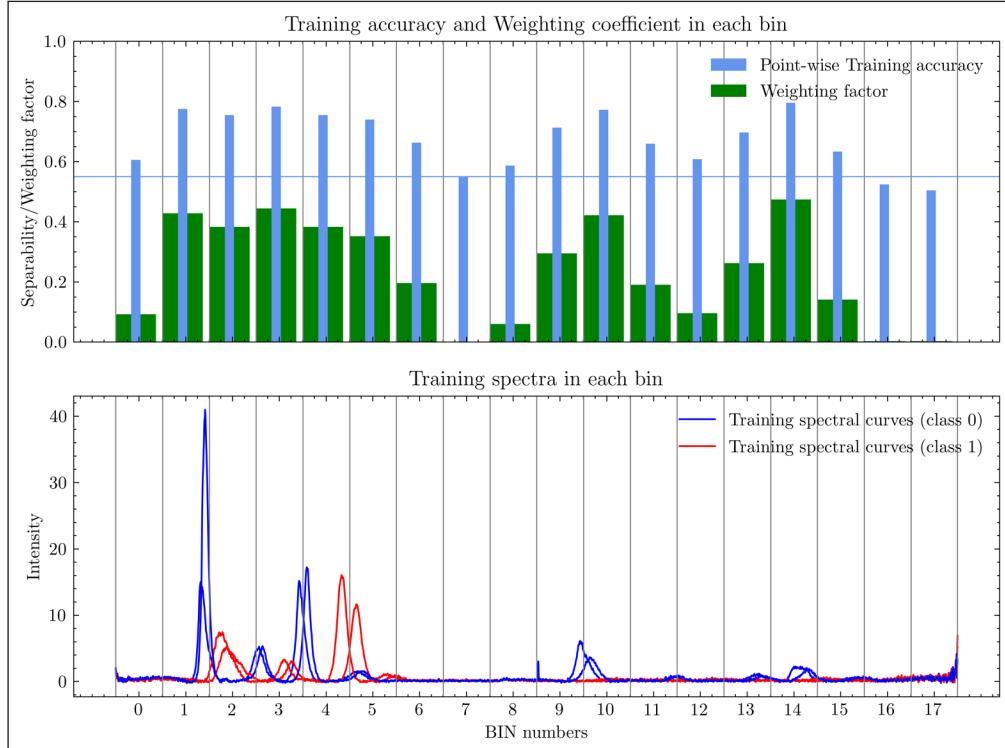
As Figure 7 shows, the spectra in the two training classes partially overlap. Hence, classification (or segmentation) at the overlapping parts is difficult, meaningless, and of low confidence. The more our spectra overlap, the lower the overall accuracy, independently how precisely our model performs the segmentation at locations with well-separable features. Because of this, we divide the spectrum into several intervals, called bins, so we can learn the local separability of the spectra in each bin separately, which is provided by the classification accuracy at the end of the training. The principle of selecting the number of bins is to ensure that at least one spectral bin has a high separability indicated by the classification accuracy. Figure 7 shows the binning for the case of dividing the spectrum into 18 bins.

After splitting the spectra, two-layered neural networks with the identical structure described above is used to perform classification training in each bin separately. The obtained classification accuracy as a separability indicator is used for automatically calculating a believability weighting factor for each bin (equation (15)).

The neural network is trained by backpropagation using gradient descent, with the Adam update scheme as described above. We use the cross-entropy as the loss function, ReLU and Softmax function as the activation function for our classification task in each bin. The statistical model is obtained by minimizing the loss function on the training data set by checking if a point

is properly classified. 700 epochs are used for iteration, in which the point-wise probabilities $\hat{y}_k^{(i)}$ and then the class id $\hat{C}^{(i)} \in \{0, 1, \dots, C - 1\}$ is predicted for each point by applying equation (12) and (13). The final classification (or space segmentation) model in each bin is shown in Figure 7 by the two background colors. As the figure inlet also highlights, the segmentation is determined by the point distribution of the spectra in each bin, and certain data will inevitably be misclassified.

In this process, the training accuracy of classifying each point in each bin is obtained. The corresponding statistics are shown in Figure 8. The optimized accuracy as a quality index for our neural network model is used as a measure for separability in each bin. High-quality classification is obtained in bins with high separability, whereas the point-wise classification in bins with low separability is misleading.



In this work, we use A_b , $\{b = 1, \dots, B\}$ to represent the training classification accuracy and so the spectral separability in bin b where B represents the number of bins. The values of A_b can vary between 0 and 1. Since 50% classification accuracy corresponds to maximum ambiguity, we consider that an accuracy below 55% corresponds to a confidence weighting factor of 0, and an accuracy of 100% corresponds to 1. To increase the impact of the training accuracy, a square operator is also introduced. Based on this, a *believability weighting factor* w_b is calculated as

$$w_b = \max(0, 1.4337A_b^2 - 0.4337) \quad (15)$$

The result shows (see Figure 8) that the more obvious the feature difference in bins, the higher the classification accuracy (separability), thus larger weighting factors. At the same time, it can also be seen that the weighting factors corresponding to non-separable regions with accuracy less than 55% are 0.

The final classification label for each spectral curve is calculated by the weighted sum of the point-wise classification results assigned by our neural networks in each bin:

$$L_{curve} = \operatorname{argmax}_{k \in \{0, \dots, C-1\}} \sum_{b=1}^B w_b \sum_{i=1}^{N_B} \hat{C}_k^{(i)} \quad (16)$$

where $\hat{C}_k^{(i)}$ equals to 1 if the i -th feature or observation $\hat{C}^{(i)}$ is labeled as k otherwise it equals to 0. $N_B = \frac{N}{B}$ represents the number of data points in each bin, where N represents the number of data points in each spectral curve. Based on the classification prediction L_{curve} for any measured spectrum, an immediate information on the phase of the system, and so on state of the experiment can be provided.

Figure 8 The distribution of point-wise training accuracy (separability), and the believability weighting factor for each bin.

C. Performance metrics

Next to classifying the spectra into 2 phases, our aim also includes finding a transition point during an experiment. It is difficult and inaccurate to manually determine such a boundary. Instead, we are interested in the robustness of the ML-based classification and want to minimize any ambiguity zone during the experiment when the classification prediction jumps inconsistently between the two phases. Hence, our performance metric of the classification model shall show how small such a zone of ambiguity is. From the physics point of view, a proper interpretation would require the phases and the ambiguity zone to be linked to specific pressure ranges. Unfortunately, the available data is not complete and does not contain such information, so an objective measure for the classification quality is not available. This also highlights the importance of storing all relevant metadata and following FAIR principles in data management.

Instead, we define the classification confidence as described here. Let N_f represent the number of spectral curves in ambiguous region, N_t represents the total number of all test spectral curves, then the classification confidence can be defined as

$$P_{conf} = 1 - \frac{N_f}{N_t} \quad (17)$$

If there is a clear division between the two types of spectra, it is considered to be 100% confident. In the particular case, if a phase change has not been found and all spectra are assigned to the same class without any boundary between the phases, it is considered that all spectra are in the ambiguous region and the classification confidence is 0.

3.3 CLASSIFICATION RESULTS

The full dataset used for the evaluation of the neural network learning model consists of 349 spectral curves, each of which has $N = 4023$ data points. Part of the dataset (one for 20 diffractograms) is shown in [Figure 3](#) as described above.

According to the performance metric equation (17), a curve-wise classification confidence for the dataset can be calculated. Applying 18 bins, the ambiguous zone is small and the final curve-wise classification confidence is 98.854%. For different number of bins applied, [Figure 9](#) shows the respective ambiguous zones, and [Figure 10](#) plots the calculated values of classification confidence. [Figure 9](#) shows the ambiguous zones around the phase change by displaying the spectra with colors according to the classification prediction results and applying a continuous movement of the intensity baselines for better visibility.

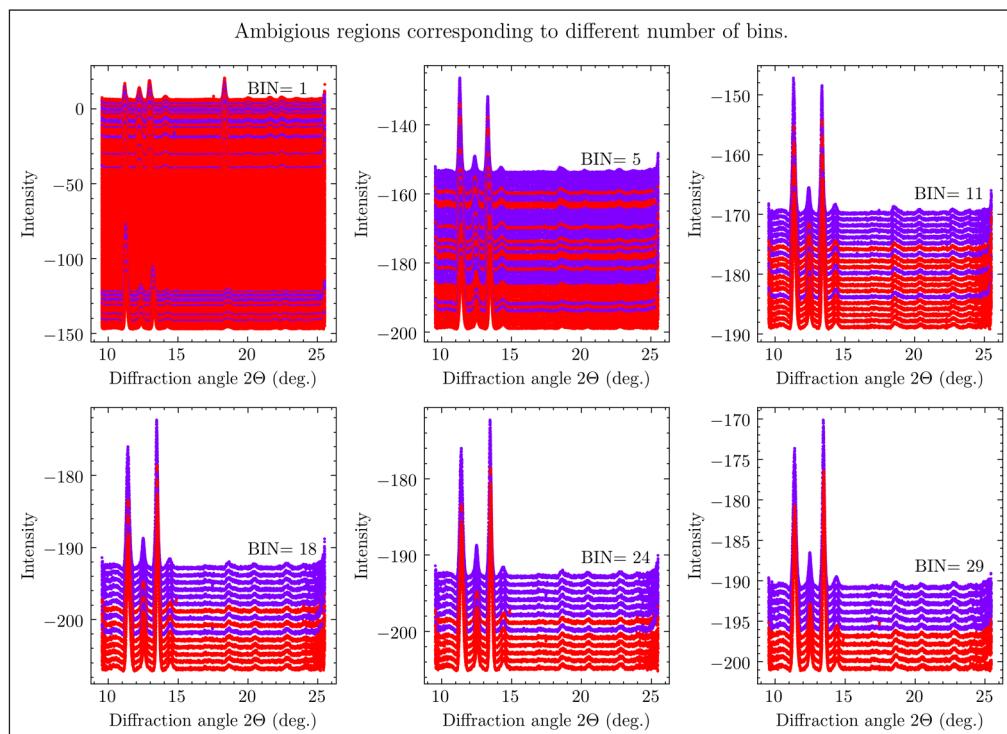


Figure 9 Final classification result of test spectral curves near the boundary (using 1, 5, 11, 18, 24, 29 bins). The spectrum is pulled evenly by moving the Intensity baseline for better display.

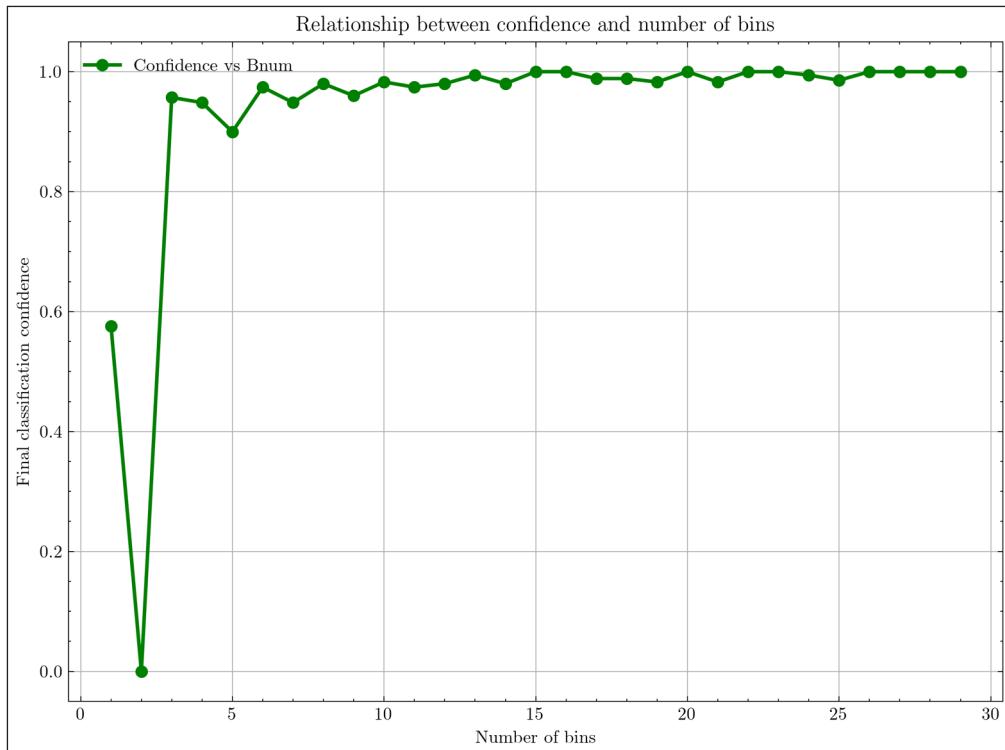


Figure 10 Relationship between final classification accuracy and number of bins.

It can be clearly seen that the more bins applied, the higher the classification accuracy, thus smaller the ambiguous zone. This is achieved by weighting the individual classification predictions calculated at the different bins, and so making the bins containing well-separable spectral features dominant and weakening the impact of misclassifications in the bins with low spectral separability and corresponding low believability weighting factor. The key in achieving a high classification confidence is finding bins with high separability index. Therefore, the optimal number of bins should be determined by increasing the number of bins until it is ensured that bins with high separability have been found.

The relationship between the classification confidence and the number of bins applied is shown in Figure 10. The overall trend is that as the number of bins increases, the ambiguous region becomes smaller and smaller. The presence of bins with high separability and corresponding high believability weighting factors immediately shows that the characteristic features of the different classes have been captured and high overall performance is expectable.

4 CONCLUSION

In this study, we presented an example of classifying experimental spectra using Neural Network based ML. We highlight the shortcoming of missing metadata and propose a solution for overcoming this by applying NeXus glossary and data format standard in XFEL sciences. Data reuse has been presented with a focus on classifying the measurement spectra into two classes either being before a phase transition or being after it. The classification is based on spectral features specific to an experimental phase. Hence, changes in spectral peaks (increasing, decreasing, vanishing, shifting) had to be found. Since the spectra of the 2 classes are overlapping in some regions, their separability is low there. We have presented that binning the data, and down-weight the classification predictions from those bins where the classification accuracy is low results in a reliable final classification. In each bin, a two-layer neural network was trained for the classification and also for determining the separability of the classes within the bin. Hence, a weighted sum of the predicted class labels of each data point from a whole spectrum leads us to the final classification of the curve. This can minimize or even eliminate the effects of misclassifications of the data points in overlapping regions.

The result shows that our spectral classification model is robust to random noise and can identify peak intensity changes or peak shifts. The classifier can provide us with an immediate feedback on the spectral class and so on the actual phase of the experiment.

5 PERSPECTIVES AND RECOMMENDATIONS

Sun and Brockhauser
Data Science Journal
DOI: 10.5334/dsj-2022-015

15

Although we obtained nice result on ML based spectral classification, the process of selecting, creating the training set is still limited, because the data is not properly annotated and some key metadata (such as incident beam energy, or the pressure value for each spectral curve) is missing. To improve the situation and ensure that data aligns with FAIR principle, a specific experiment data model shall be built to connect the instrumentation and data acquisition entries to physics models, so ML can be set up much easier and more efficiently because all required metadata is registered for proper interpretation without the need for personal consultation. Since the diffractograms in our study are measured in a course of pressure ramping, an appropriate NeXus application definition should have been made to define all relevant metadata entries that need to be recorded. Experiments recording data and metadata according to such experiment data model would then help data interpretation, especially machine readability, and reuse.

In our ML algorithm, the weighting factors are calculated based on the training accuracy in each bin. As a next step, the Neural Network can be extended to form an end-to-end ML structure which can automatically learn these weighting factors and output directly the final classification label.

Another development possibility is to handle the spectral data as a set of one-dimensional time series and instead of a space-segmentation approach presented here, the classification problem could check for intra-spectrum characteristics, too. In this setting, we can apply different supervised deep learning neural network architectures for the spectra classification, such as the convolutional neural network (CNN), Residual Networks (ResNets) (He et al. 2016), LSTM-based architecture (Hochreiter et al. 1997), attention-based architectures(Vaswani et al. 2017). At the same time, unsupervised learning such as clustering is also very suitable for such classification tasks.

ABBREVIATIONS

AI—Artificial Intelligence

CNN—Convolutional Neural Network

DL—Deep Learning

FAIR—Findable, Accessible, Interoperable, Reusable

FXE—Femtosecond X-ray. Experiments

HED—High Energy Density

hdf5—Hierarchical Data Format (<https://www.hdfgroup.org/solutions/hdf5/>)

HDFView—hdf5 viewer software (<https://www.hdfgroup.org/downloads/hdfview/>)

LAMBDA—Large Area Medipix-Based Detector Array

LSTM—Long-Short Term Memory

MID—Materials Imaging and Dynamics

ML—Machine Learning

NeXus—Neutron, x-ray, and muon science data format.

NN—Neural Network

NXDL—NeXus Definition Language

NXmx—NeXus for macromolecular crystallography

P02.2—Extreme Conditions Beamline at the synchrotron PETRA III

PETRA III—Synchrotron facility at DESY in Hamburg, Germany

ReLU—Rectified Linear Unit

SCS—Spectroscopy and Coherent Scattering

XFEL—X-ray Free Electron Laser

ACKNOWLEDGEMENT

The authors would like to thank Christian Plueckthun and Zuzana Konopkova at European XFEL for providing the HED experimental spectral data (DOI: [10.5281/zenodo.4424866](https://doi.org/10.5281/zenodo.4424866)). Special thanks to Christian Plueckthun for careful checking and corrections of ‘spectral data description’ part of the article, as well as the explanation, discussion and comment on the HED experiment and data. Funding for this research is provided by China Scholarship Council (CSC) and the Photon and Neutron Open Science Cloud (PaNOSC) EU H2020 project (#823852).

Sun and Brockhauser
Data Science Journal
DOI: 10.5334/dsj-2022-015

16

COMPETING INTERESTS

The authors have no competing interests to declare.

AUTHOR AFFILIATIONS

Yue Sun  orcid.org/0000-0001-5691-9401

Institute of Informatics, University of Szeged, Szeged, Hungary; Open Scientific Computing Group, European XFEL GmbH, Schenefeld, Germany

Sandor Brockhauser  orcid.org/0000-0002-9700-4803

Institute of Informatics, University of Szeged, Szeged, Hungary; Open Scientific Computing Group, European XFEL GmbH, Schenefeld, Germany

REFERENCES

- Basham, M, Filik, J, Wharmby, MT, Chang, PC, El Kassaby, B, Gerring, M, Aishima, J, Levik, K, Pulford, BC, Sikkharulidze, I and Sneddon, D.** 2015. Data analysis workbench (DAWN). *Journal of synchrotron radiation*, 22(3): 853–858. DOI: <https://doi.org/10.1107/S1600577515002283>
- Bernstein, HJ, Förster, A, Bhowmick, A, Brewster, AS, Brockhauser, S, Gelisio, L, Hall, DR, Leonarski, F, Mariani, V, Santoni, G and Vonrhein, C.** 2020. Gold Standard for macromolecular crystallography diffraction data. *JUCrJ*, 7(5). DOI: <https://doi.org/10.1107/S2052252520008672>
- Bishop, CM.** 2006. Pattern Recognition and Machine Learning. Springer.
- Brockhauser, S, Ravelli, RB and McCarthy, AA.** 2013. The use of a mini-goniometer head in macromolecular crystallography diffraction experiments. *Acta Crystallographica Section D: Biological Crystallography*, 69(7): 1241–1251. DOI: <https://doi.org/10.1107/S0907444913003880>
- Brockhauser, S, Svensson, O, Bowler, MW, Nanao, M, Gordon, E, Leal, RM, Popov, A, Gerring, M, McCarthy, AA and Gotz, A.** 2012. The use of workflows in the design and implementation of complex experiments in macromolecular crystallography. *Acta Crystallographica Section D: Biological Crystallography*, 68(8): 975–984. DOI: <https://doi.org/10.1107/S090744491201863X>
- Decking, W, Abegyan, S, Abramian, P, Abramsky, A, Aguirre, A, Albrecht, C, Alou, P, Altarelli, M, Altmann, P, Amyan, K, Anashin, V, Apostolov, E, Appel, K, Auguste, D, Ayvazyan, V, Baark, S, Babies, F, Baboi, N, Bak, P and Zybin, D.** 2020. A MHz-repetition-rate hard X-ray free-electron laser driven by a superconducting linear accelerator. *Nature Photonics*, 14: 1–7. DOI: <https://doi.org/10.1038/s41566-020-0607-z>
- Deng, L and Yu, D.** 2014. “Deep Learning: Methods and Applications” (PDF). *Foundations and Trends in Signal Processing*. 7(3–4): 1–199. DOI: <https://doi.org/10.1561/2000000039>
- Donath, T, Rissi, M and Billich, H.** 2013. Meeting report: workshop on beamline integration and data formatting. *Synchrotron Radiation News*, 26(5): 34–35. DOI: <https://doi.org/10.1080/08940886.2013.832589>
- Duchi, J, Hazan, E and Singer, Y.** 2011. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7).
- Edelen, A, Mayes, C, Bowring, D, Ratner, D, Adelmann, A, Ischebeck, R, Snuverink, J, Agapov, I, Kammering, R, Edelen, J and Bazarov, I.** 2018. Opportunities in machine learning for particle accelerators. *arXiv preprint arXiv:1811.03172*.
- Fangohr, H, Beg, M, Bondar, V, Aplin, S, Barty, A, Kuhn, M, Mariani, V and Kluyver, T.** 2017. Data analysis support in Karabo at European XFEL. DOI: <https://doi.org/10.18429/JACoW-ICALEPCS2017-TUCPA01>
- Filik, J, Ashton, AW, Chang, PCY, Chater, PA, Day, SJ, Drakopoulos, M, Gerring, MW, Hart, ML, Magdysyuk, OV, Michalik, S and Smith, A.** 2017. Processing two-dimensional X-ray diffraction and small-angle scattering data in DAWN 2. *Journal of applied crystallography*, 50(3): 959–966. DOI: <https://doi.org/10.1107/S1600576717004708>

- Galler, A, Gawelda, W, Biednov, M, Bomer, C, Britz, A, Brockhauser, S, Choi, TK, Diez, M, Frankenberger, P, French, M and Görries, D.** 2019. Scientific instrument Femtosecond X-ray Experiments (FXE): instrumentation and baseline experimental capabilities. *Journal of synchrotron radiation*, 26(5). DOI: <https://doi.org/10.1107/S1600577519006647>
- Ghosh, K, Stuke, A, Todorović, M, Jørgensen, PB, Schmidt, MN, Vehtari, A and Rinke, P.** 2019. Deep Learning Spectroscopy: Neural Networks for Molecular Excitation Spectra. *Advanced Science*, 6: 1801367. DOI: <https://doi.org/10.1002/advs.201801367>
- Han, J and Moraga, C.** June 1995. The influence of the sigmoid function parameters on the speed of backpropagation learning. In *International Workshop on Artificial Neural Networks* (pp. 195–201). Berlin, Heidelberg: Springer. ISBN 978-3-540-59497-0.
- He, K, Zhang, X, Ren, S and Sun, J.** 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770–778).
- Hester, J.** 2016. A Robust, Format-Agnostic Scientific Data Transfer Framework. *Data Science Journal*, 15: 12. DOI: <https://doi.org/10.5334/dsj-2016-012>
- Hinton, G, Srivastava, N and Swersky, K.** 2012. Neural networks for machine learning lecture 6a overview of mini-batch gradient descent. *Cited on*, 14(8).
- Hochreiter, S and Schmidhuber, J.** 1997. Long short-term memory. *Neural computation*, 9(8): 1735–1780.
- Kingma, DP and Ba, J.** 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
- Könnecke, M, Akeroyd, FA, Bernstein, HJ, Brewster, AS, Campbell, SI, Clausen, B, Cottrell, S, Hoffmann, JU, Jemian, PR, Männicke, D and Osborn, R.** 2015. The NeXus data format. *Journal of applied crystallography*, 48(1): 301–305. DOI: <https://doi.org/10.1107/S1600576714027575>
- Kulin, M, Kazaz, T, Moerman, I and De Poorter, E.** 2018. End-to-end learning from spectrum data: A deep learning approach for wireless signal identification in spectrum monitoring applications. *IEEE Access*, 6: 18484–18501. DOI: <https://doi.org/10.1109/ACCESS.2018.2818794>
- Kuster, M, Boukhelef, D, Donato, M, Dambietz, JS, Hauf, S, Maia, L, Raab, N, Szuba, J, Turcato, M, Wrona, K and Youngman, C.** 2014. Detectors and calibration concept for the European XFEL. *Synchrotron radiation news*, 27(4): 35–38. DOI: <https://doi.org/10.1080/08940886.2014.930809>
- Le Guyader, L, Döring, F, Rösner, B, Eschenlohr, A, Beye, M, Agarwal, N, Broers, C, Carley, R, Delitz, JT, Gerasimova, N and Gort, R.** 2020. Beam-splitting off-axis zone plate for time-resolved X-ray absorption spectroscopy at the SCS instrument. In *European XFEL Users' Meeting 2020* (No. POSTER-2020-010).
- LeCun, Y, Bengio, Y and Hinton, G.** 2015. Deep learning. *Nature*, 521(7553): 436–444. DOI: <https://doi.org/10.1038/nature14539>
- Matsumoto, M and Nishimura, T.** 1998. Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator. *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, 8(1): 3–30.
- Mercurio, G, Broers, C, Carley, R, Delitz, JT, Gerasimova, N, Le Guyader, L, Mercadier, L, Reich, A, Schlappa, J, Teichmann, M and Yaroslavtsev, A.** September 2019. First commissioning results of the KB mirrors at the SCS instrument of the European XFEL. In *Advances in Metrology for X-Ray and EUV Optics VIII* (Vol. 11109, p. 111090F). International Society for Optics and Photonics. DOI: <https://doi.org/10.1117/12.2530725>
- Mohri, M, Rostamizadeh, A and Talwalkar, A.** 2018. *Foundations of machine learning*. MIT press.
- Nair, V and Hinton, GE.** January 2010. Rectified linear units improve restricted boltzmann machines. In *ICML*.
- Nakatsuji, M, Tschentscher, T, Cowan, T, Ferrari, A, Schlenvoigt, HP, Appel, K, Strempfer, J and Zimmermann, MV.** 2014. Scientific Instrument High Energy Density Physics (HED).
- Namin, AH, Leboeuf, K, Muscedere, R, Wu, H and Ahmadi, M.** May 2009. Efficient hardware implementation of the hyperbolic tangent sigmoid function. In *2009 IEEE International Symposium on Circuits and Systems* (pp. 2117–2120). IEEE.
- Pennicard, D, Smoljanin, S, Pithan, F, Sarajlic, M, Rothkirch, A, Yu, Y, Liermann, HP, Morgenroth, W, Winkler, B, Jenei, Z and Stawitz, H.** 2018. LAMBDA 2M GaAs—A multi-megapixel hard X-ray detector for synchrotrons. *Journal of Instrumentation*, 13(01): C01026. DOI: <https://doi.org/10.1088/1748-0221/13/01/C01026>
- Ramsundar, B and Zadeh, RB.** 2018. *TensorFlow for deep learning: from linear regression to reinforcement learning*. “O'Reilly Media, Inc.”.
- RDA COVID-19 Working Group.** 2020. Recommendations and guidelines on data sharing. *Research Data Alliance*, 10. DOI: <https://doi.org/10.15497/rda00052>
- Ruder, S.** 2016. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*.
- Singh, S.** 2001. Quantifying structural time varying changes in helical data. *Neural Computing & Applications*, 10(2): 148–154.

Suganya, S and Charles, EYA. September 2019. Speech Emotion Recognition Using Deep Learning on audio recordings. In *2019 19th International Conference on Advances in ICT for Emerging Regions (ICTer)*, 250: 1–6. IEEE. DOI: <https://doi.org/10.1109/ICTer48817.2019.9023737>

Sun and Brockhauser
Data Science Journal
DOI: 10.5334/dsj-2022-015

18

Tschentscher, T, Bressler, C, Gruenert, J, Madsen, A, Mancuso, A, Meyer, M, Scherz, A, Sinn, H and Zastrau, U. 2017. Photon Beam Transport and Scientific Instruments at the European XFEL. *Applied Sciences*, 7: 592. DOI: <https://doi.org/10.3390/app7060592>

Vaswani, A, Shazeer, N, Parmar, N, Uszkoreit, J, Jones, L, Gomez, AN, Kaiser, L and Polosukhin, I. 2017. Attention is all you need. *arXiv preprint arXiv:1706.03762*.

Wilkinson, MD, Dumontier, M, Aalbersberg, IJ, Appleton, G, Axton, M, Baak, A, Blomberg, N, Boiten, JW, da Silva Santos, LB, Bourne, PE and Bouwman, J. 2016. The FAIR Guiding Principles for scientific data management and stewardship. *Scientific data*, 3(1): 1–9. DOI: <https://doi.org/10.1038/sdata.2016.18>

Yang, J, Xu, J, Zhang, X, Wu, C, Lin, T and Ying, Y. 2019. Deep learning for vibrational spectral analysis: Recent progress and a practical guide. *Analytica Chimica Acta*, 1081: 6–17. DOI: <https://doi.org/10.1016/j.aca.2019.06.012>

Zalden, P, Quirin, F, Schumacher, M, Siegel, J, Wei, S, Koc, A, Nicoul, M, Trigo, M, Andreasson, P, Enquist, H, Shu, MJ, Pardini, T, Chollet, M, Zhu, D, Lemke, H, Ronneberger, I, Larsson, J, Lindenberg, AM, Fischer, HE, Hau-Riege, S, Reis, DA, Mazzarello, R, Wuttig, M and Sokolowski-Tinten, K. 2019. ‘Femtosecond x-ray diffraction reveals a liquid–liquid phase transition in phase-change materials’, *Science*, 364(6445): 1062–1067. DOI: <https://doi.org/10.1126/science.aaw1773>

Zeiler, MD. 2012. ADADELTA: An adaptive learning rate method. *arXiv 2012. arXiv preprint arXiv:1212.5701*, 1212.

TO CITE THIS ARTICLE:

Sun, Y and Brockhauser, S. 2022. Machine Learning Applied for Spectra Classification in X-ray Free Electron Laser Sciences. *Data Science Journal*, 21: 15, pp. 1–18. DOI: <https://doi.org/10.5334/dsj-2022-015>

Submitted: 28 December 2020

Accepted: 11 April 2022

Published: 09 August 2022

COPYRIGHT:

© 2022 The Author(s). This is an open-access article distributed under the terms of the Creative Commons Attribution 4.0 International License (CC-BY 4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited. See <http://creativecommons.org/licenses/by/4.0/>.

Data Science Journal is a peer-reviewed open access journal published by Ubiquity Press.