Towards Reflectivity profile inversion through Artificial Neural Networks

Juan Manuel Carmona Loaiza

Jülich Centre for Neutron Science (JCNS) at Heinz Maier-Leibnitz Zentrum (MLZ). Lichtenbergstraße 1, 85748 Garching, Germany.

E-mail: j.carmona.loaiza@fz-juelich.de

30 July 2020

Abstract. The goal of Specular Neutron and X-ray Reflectometry is to infer materials Scattering Length Density (SLD) profiles from experimental reflectivity curves. This paper focuses on investigating an original approach to the ill-posed non-invertible problem which involves the use of Artificial Neural Networks (ANN). In particular, the numerical experiments described here deal with large data sets of simulated reflectivity curves and SLD profiles, and aim to assess the applicability of Data Science and Machine Learning technology to the analysis of data generated at large scale facilities. It is demonstrated that, under certain circumstances, properly trained Deep Neural Networks are capable of correctly recovering plausible SLD profiles when presented with never-seen-before simulated reflectivity curves. When the necessary conditions are met, a proper implementation of the described approach would offer two main advantages over traditional fitting methods when dealing with real experiments, namely, 1. no prior assumptions about the sample physical model are required and 2. the times-to-solution are shrank by orders of magnitude, enabling faster batch analyses for large datasets.

Keywords: inverse problems, neutron scattering, x-ray scattering, reflectometry, reflectivity, data science, data analysis, algorithms, artificial intelligence, machine learning, neural networks

Submitted to: *Mach. Learn. Sci. Technol.*

1. Introduction

Neutron and X-ray Specular Reflectometry are established experimental techniques whose aim is to investigate interfacial structures at the sub-nanometer scale through the measurement and analysis of reflectivity curves [1, 2, 3].

In a typical specular reflectometry experiment, a collimated Neutron or X-ray beam of wavelength λ impinges on the surface of a flat sample at an incident angle θ. The incident angle is varied and the specular reflectivity is measured as the ratio between the reflected and the incident beam intensities, $R(\theta) = I_R(\theta)/I_0(\theta)$.

Theoretically, in the absence of significant non-specular scattering from in-plane variations of the SLD, neutron specular reflectivity is accurately described by a one-dimensional Schrödinger wave equation‡,

$$-\frac{\partial^2\psi(k_{0z},z)}{\partial z^2} + 4\pi\,\rho(z)\psi(k_{0z},z) = k_{0z}^2\,\psi(k_{0z},z), \qquad (1)$$

where ψ is the wave function, ρ is the SLD profile of a given sample, k_{0z} is the wave vector z component and z is the depth inside the sample, perpendicular to the sample interfaces (For more details see e.g. [4] and references therein).

In terms of the solution to equation 1, the amplitude of the reflected wave can be represented by the integtral

$$r(Q) = \frac{4\pi}{iQ}\int_0^L \psi(k_{0z},z)\,\rho(z)\,\mathrm{e}^{ik_{0z}\,z}\mathrm{d}z, \qquad (2)$$

where $r(Q)$ is the complex-valued reflection amplitude as function of the wave vector transfer perpendicular to the surface, $Q = 2\,k_{0z} = 4\pi\sin(\theta)/\lambda$, and L is the thickness of the SLD profile. However, the reflection amplitude is not accessible to measurements, but only the reflectivity which, in terms of the reflection amplitude, reads $R(Q) \equiv r^*r$.

1.1. The phase problem and fitting

The measured reflectivity, $R(Q)$, does not carry any information regarding the phase, making the inference of an SLD profile from a reflectivity curve a non-invertible inverse problem: at a theoretical level, there are families of SLD profiles which produce exactly the same reflectivity curve. In particular, this applies to any anti-periodic SLD profile that is reflected at the mid point (See Figure 1). To an experimenter measuring reflectivities, $R(Q) = r^*r$, both SLD profiles are indistinguishable.

‡ Throughout this work, derivations and discussions focus mainly on neutron reflectometry, however, the same derivations and approach apply to the X-ray case with little or no change.

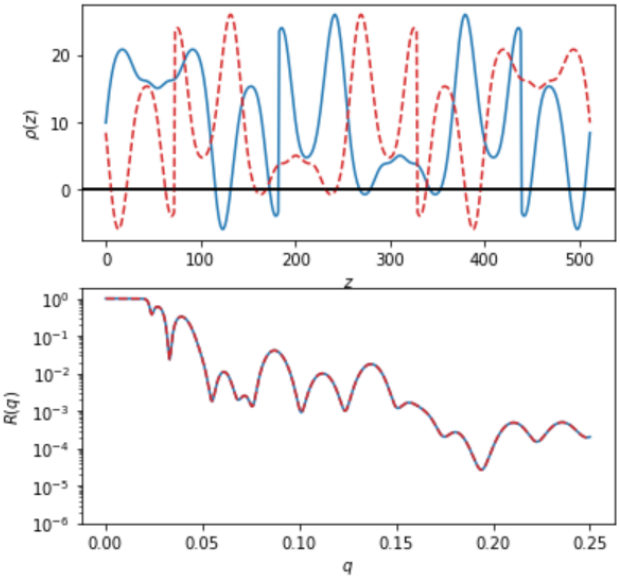

Figure 1. By applying a reflection transformation to an anti-periodic SLD profile, the exact same reflectivity curve is reproduced.

The degeneracy causing different SLD profiles to produce identical reflectivity curves is known as the phase problem, and is accentuated by the truncation of the reflectivity data at a maximum value of Q and the statistical uncertainty associated with noisy data points (e.g. See [5]).

Typically, the data obtained from Reflectometry experiments is analyzed in terms of physical models, trusting their ability to reproduce measured experimental reflectivity curves. Using specialized software (e.g.[6, 7]), an iterative process of parameter optimization is thus established in which, at each iteration, 1. certain parameters of the physical model are set, 2. theoretical reflectivity curves are calculated and 3. a comparison is made between the theoretical and experimental reflectivity curves. The latter comparison is quantified by a Figure of Merit (FOM) and the goal of the iterative process is reached when the FOM reaches its minimum value, namely, when the experimental and the theoretical reflectivity curves are as close as allowed by the physical model and the experimental resolution. Such an iterative *fitting* process is by far immediate, requiring experimenters to try out several FOMs, several minimization algorithms, several sets of model parameters and even several physical models.

In recent years, BornAgain [6] –a well established code for simulating and fitting neutron and X-ray grazing-incidence small-angle scattering (GISAS) data, has started to support fitting and simulation capabilities for reflectometry data as well. The present work aims at exploring the possibilities that Machine

Learning has to offer towards the development of reflectivity data-driven software.

2. The expressive power of artificial neural networks

Deep neural networks can be thought of as compositions of multiple simple functions (called layers) that can approximate rather complicated functions. In fact, the celebrated universal approximation theorem states that depth-2 networks with suitable activation functions can approximate any continuous function on a compact domain to any desired accuracy [8, 9, 10, 11]. However, the size of such a neural network could be exponential in the input dimension, which means that the depth-2 network may have a very large width [12, 13, 14]. In fact, part of the recent renaissance in Artificial Neural Networks (ANN), lies not on enabling wider networks to be trained, but on the empirical observation that *deep neural networks* tend to achieve greater expressive power per parameter than their shallow counterparts.

It has been shown that any Lebesgue-integrable function from $\mathbb{R}^N \to \mathbb{R}$ can be approximated by a fully-connected `ReLU` deep neural network of width $N+4$ to arbitrary accuracy with respect to L_1 distance and, except for a negligible set, all functions from $\mathbb{R}^N$ to $\mathbb{R}$ cannot be approximated by any `ReLU` network whose width is no more than N [15]. [16] show that any continuous function $f : [0,1]^{d_{in}} \to \mathbb{R}^{d_{out}}$ can be approximated by a net of width $d_{in} + d_{out}$, obtaining also quantitative depth estimates for such an approximation in terms of the modulus of continuity of f. At the same time, they claim that there are no conclusive results regarding the depth such a network should have, and, even in the case that a precise ANN architecture to achieve a given precision can be defined, nothing can be said regarding the success of the training process.

In contrast to these apparent theoretical drawbacks, one of the first architectures that is taught when studying ANNs –a single hidden layer of width 128, is able to classify 28×28 pixel images (i.e. points in $\mathbb{R}^{784}$) into 10 discrete categories, passing in the process through a mapping into the real unit interval (e.g. `TensorFlow` tutorials [17]). These achievements, in apparent contradiction with the theoretical results described above, are possible because of the fact that the images under classification do not sample the whole $\mathbb{R}^{784}$ but are drawn from only a very limited subspace of it. In fact, when the trained network is presented with images that do not belong to that subspace, the ANN fails –it may even classify apparently random noise as some of the 10 digits with almost 100% certainty.

In the present work, focus is made on training simple and small ANNs targeted to specialized kinds of SLD profiles, in contrast to larger and general-purpose neural networks, as ANN attempts to learn a general pseudo-inverse function, for the time being, are almost certainly doomed to fail.

3. Related work

Non invertible inverse problems are not unique to X-ray and Neutron reflectometry, and several other scientific communities have already started to investigate the usefulness of Artificial Neural Networks for tackling them, showing astonishing performance for applications like low-dose computed tomography or various sparse data problems. While there are few theoretical results, some well-posedness results and quantitative error estimates have been found for some problems [18]. For instance, in Electrical Impedance Tomography (EIT), which represents the typical nonlinear ill-posed problem, the electrical properties of tissues are determined by injecting a small amount of current and measuring the resulting electric potential, which must be transformed into a tomographic image by some reconstructing algorithm. Many artificial intelligence approaches to tackle EIT have been taken in the past few years (e.g. [19] and references therein) with outstanding results.

In the realm of X-ray reflectivity, a recent work shows that properly trained ANNs with simple fully connected architectures can be used to characterize thin film properties (thickness, roughness and density) from XRR data within milliseconds and minimal a priori knowledge. Their results differ from traditional least mean squares fitting by less than 20% [20]. Such an approach could benefit the study of the growth behavior of thin films.

4. Data simulation, preprocessing, network architecture, training and implementation.

4.1. Scaling of the problem

The only physical quantities involved in the calculation of a reflectivity curve, assuming a perfect instrument able to measure up to $Q \to \infty$ and an SLD profile extending up to $z \to \infty$, are the wave transfer vector Q and the SLD profile $\rho(z)$. These quantities can be further reduced by using dimensional analysis. In fact, the number of dimensionless groups that define the problem, which equals the total number of physical quantities (Q and ρ) minus the fundamental dimensions (length), is only one ($= 2 - 1$). By choosing an arbitrary SLD scale ρ_0 and defining the dimensionless parameter $p = \rho/\rho_0$, equation (2) can

be re-casted in the following form:

$$r_0(Q) = \sqrt{\rho_0} \times \frac{4\pi}{iQ} \int_0^\infty \psi(\eta, \xi)\, p(\xi)\, \mathrm{e}^{i\eta\,\xi} \mathrm{d}\xi, \tag{3}$$

where $\xi = z\sqrt{\rho_0}$, $\eta = k_{0z}/\sqrt{\rho_0}$, and $p(\xi) = \rho(z)/\rho_0$ are the dimensionless depth, wave vector and SLD profile respectively. Thus, to solve for a different SLD scale, ρ_*, it is enough to solve for $r_0(Q)$ and rescale afterwards by $\sqrt{\rho_*/\rho_0}$, i.e.

$$r_*(Q) = \sqrt{\frac{\rho_*}{\rho_0}} \times r_0(Q). \tag{4}$$

In the following, the SLD scale of the problem is chosen to be that of the substrate, $\rho_0 = \rho_{subs}$, i.e., $p_{subs} \equiv 1$.

4.2. Data simulation

The phase problem implies that a single input (e.g. a reflectivity curve) may be consistent with two or more different outputs (e.g. SLD profiles). If one were to train an ANN to find a pseudo-inverse transformation, using data containing different output targets corresponding to the same input (different branches), the training process would not be successful, as different branches would cause the weights of the ANN to drift in inconsistent directions. In order to avoid such a situation, it must be ensured that either $*1$.- the solution space has no branches, or $*2$.- the training targets lie all in the same branch of the solution space. For the last scenario, it must also be kept in mind that ANNs trained in such a way will only be useful as long as the expected solutions are consistent with the branch to which the training targets belong. To fulfill $*1$, a set of artificial SLD profiles is generated which offer a 1-1 correspondence to their associated reflecticity curves –SLD profiles odd with respect to the middle of the depth. To try to fulfill $*2$ to some extent, two sets of SLD profiles are generated, each set corresponding to a physically relevant typology of samples, namely, single films and lamellar structures.

All simulated SLD profiles used as training targets in this work have an overall thickness $L = 512$Å, and are sampled by 512 equally spaced points within the semi-closed interval $z = (0, L]$, between two semi-infinite fronting and backing mediums of constant SLD, $\rho_{-\infty} = 0$ and $\rho_{+\infty} = 10^{-6}$Å^{-2} respectively. Each SLD profile was thus modeled as a `Multilayer` composed of 512 slices, 1 Å thick each, with no interfacial roughness. In this way, smooth SLD profiles were mimicked by quasi-continuous small variations of the SLD between contiguous layers throughout the whole interfacial structure. Such SLD profiles are then used to simulate the corresponding reflectivity curves for which the wave vector transfer is limited to an interval $0 \leq Q \leq Q_{\max} = 0.25$, sampled by 129 equally spaced points. Imperfections in the reflectivity curves are only characterized by a background of 10^{-6} and a Q-resolution of 5%.

In order to bring the so-called *features* (in the case at hand, the reflectivity curves) into a small dynamic range, the average reflectivity curve needs to be subtracted from all reflectivity curves in the set. Thus, the new set of curves is a zero-mean set of curves. After that, each of the resulting zero-mean curves is rescaled by dividing it by the standard-deviation curve. Finally, the obtained zero-mean, unit-standard-deviation set of curves is ready to be used for training. In the following, we refer to this set of curves as the *training features*. In contrast to the reflectivity curves, the SLD profiles, i.e. the *training targets*, are left unchanged.

Figure 2 gives an overview of the three different data sets used and their preprocessing, and a more detailed discussion of each is carried out in section 5.

4.3. Network architecture

The network architecture is given to some extent by the problem constraints: The number of input neurons must be the same as the input reflectivity curve lengths, which is chosen to be 129. The number of output neurons must be the same as the expected SLD profile resolution points, which are 512. To allow for enough expressive power of the network, four hidden layers are created with 2048 neurons –a size four times larger than that of the output layer and 16 times larger than that of the input neurons. The network is kept relatively shallow to allow fast trainings and computation of SLD profiles. To reproduce non-linearities, `ReLU` functions are used after each hidden layer, except the one before the output layer. In order to prevent *overfitting*, a single dropout layer with a rate of 0.5 is added as a regularizer before the output layer. The architecture of the ANNs, as displayed by the `keras` method `Model.Summary()` is the following:

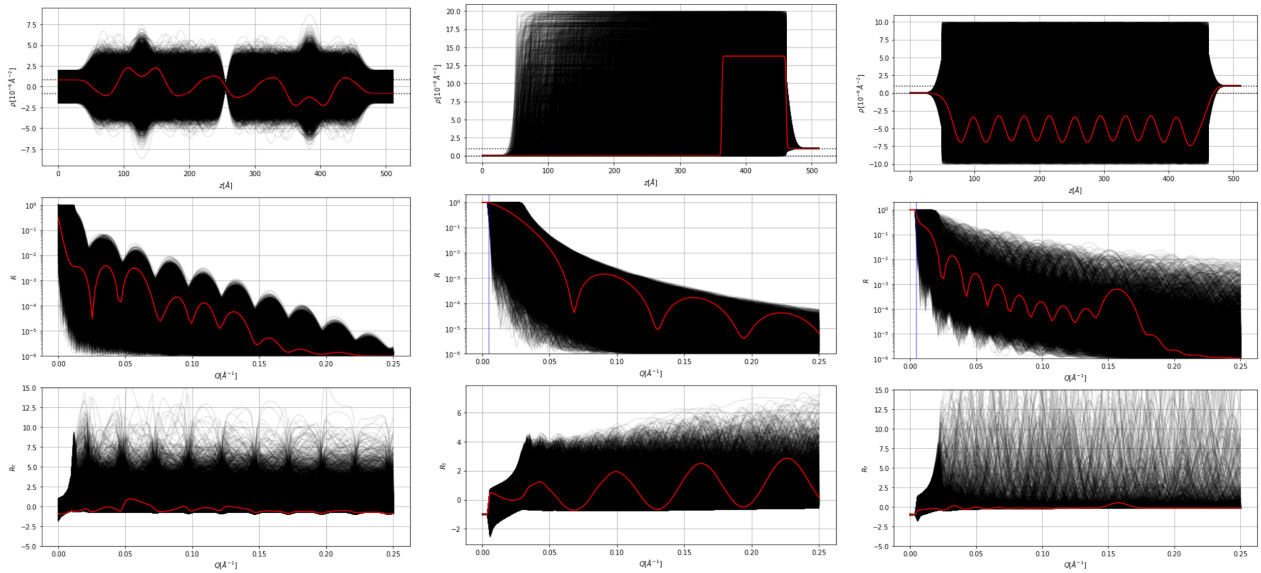

Figure 2. In the **top row**, three different families of SLD profiles are shown: Odd functions, single films and lamellar structures; the **middle row** shows the simulated reflectivity curves obtained from the SLDs in the top panels; The **bottom row** show the rescaling of the reflectivity curves which have zero mean and unit standard deviation. A single random curve is highlighted in red to give a qualitative impression of the features defining each family of curves.

```
_________________________________________________
Model: "sequential"
_________________________________________________
Layer (type)                 Output Shape              Param #
=================================================
reshape (Reshape)            (None, 129)               0
_________________________________________________
dense (Dense)                (None, 2048)              266240
_________________________________________________
dense_1 (Dense)              (None, 2048)              4196352
_________________________________________________
dense_2 (Dense)              (None, 2048)              4196352
_________________________________________________
dropout (Dropout)            (None, 2048)              0
_________________________________________________
dense_3 (Dense)              (None, 512)               1049088
=================================================
Total params: 9,708,032
Trainable params: 9,708,032
Non-trainable params: 0
_________________________________________________
```

4.4. Training

At training time, the optimizer of choice is the Adaptive Moment Estimation algorithm (ADAM [21]), together with a mean-squared-error (MSE) loss function. All models are set to train for 500 epochs and an early stopping callback with a `patience` parameter value of 10 epochs is also provided. This callback prevents overfitting by stopping the ANNs training whenever the error in the validation set does not decrease anymore through the epochs.

5. Results

Three different datasets have been used to train three different neural networks: i) SLD profiles possessing odd symmetry, ii) single films and iii) lamellar structures. Thus, while all of the networks are architecturally equal, their learned weights are different. The trained ANNs were tested by feeding them 5K never-seen-before reflectivity curves and calculating the Mean Absolute Error,

$$\mathrm{MAE}_j = \mathrm{MAE}(\rho_j, \tilde{\rho_j}) = \frac{1}{\mathrm{N_{RES}}} \sum_k \rho_{jk} - \tilde{\rho}_{jk}, \qquad (5)$$

between the ANN predicted SLD profile and the SLD target profile.

5.1. Odd SLD profiles

If an SLD profile possesses the symmetry $\rho(z + L/2) = -\rho(z) + \mathrm{const}$, $z \in [0, L/2]$, the same reflectivity curve it produces is recovered by reflecting it at its mid-point (c.f. Section 1.1). Thus, by defining a dataset composed only by odd SLD profiles, it is ensured that there is a one to one correspondence between an SLD profile and its associated reflectivity curve. Figure 3 shows five test SLD profiles recovered after a neural network is trained using such a dataset. The overall Mean Absolute Error for the test set, composed of 5K samples, lies around 0.35.

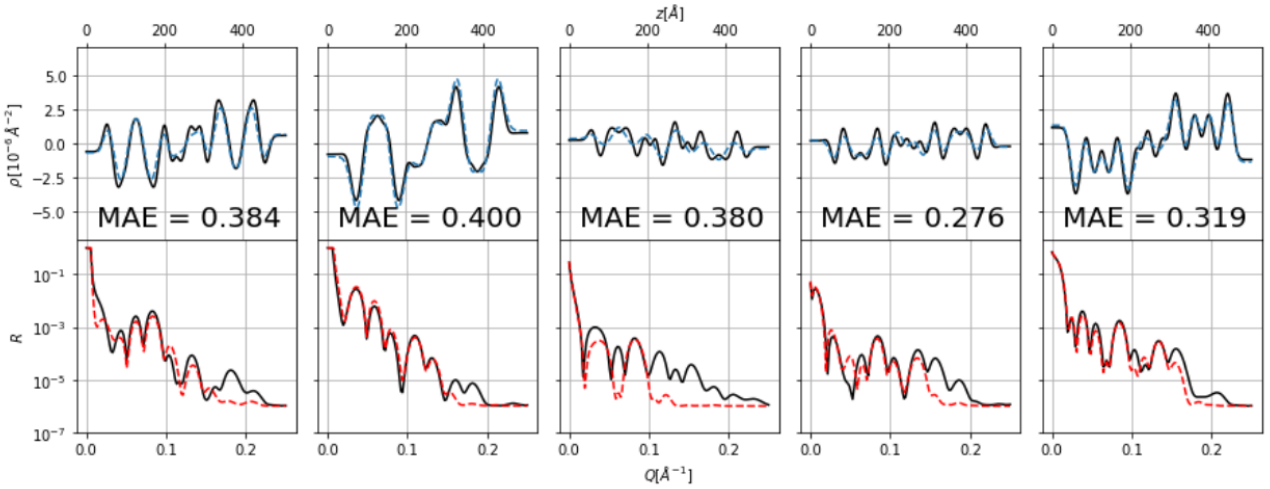

Figure 3. By training an ANN using a set of 100K random odd SLD profiles, the phase problem is not present and there is a one to one correspondence between an SLD profile and its associated reflectivity curve. Five test reflectivity curves (black lines; bottom panels) were shown to the trained network for the first time; from those curves, the network guesses the SLD profiles that produce them (blue dashed lines; top panels). In the figure, the Mean Absolute Errors (MAE) with respect to the true SLD profiles (black lines; top panels) are also shown. Additional reflectivity curves are shown (red dashed lines, bottom panels), generated from the SLD profiles predicted by the ANN.

5.2. Training on films with positive SLD

The simplest family of SLD profiles is that of single layers on top of a substrate. In this family, each SLD profile is made up of only three regions: The superstrate, $\rho_{sup} = 0$ extending from $z = -\infty$ to $z = \Delta_-$; the substrate, $\rho_{sub} = 1$ extending from $z = 512 - \Delta_+$ to $z = \infty$; and a film of constant SLD ρ_f extending from $z = \Delta_-$ to $z = 512 - \Delta_+$. $\Delta_{+/-}$ are buffer regions defined to smoothen the transition between the SLD of the film and that of the super- and substrates. $\Delta_+ = 50$ is fixed for all SLD profiles of this data set, while $\Delta_- \in [50, \Delta_+)$ is randomly chosen for each generated profile and effectively defines the thickness of the film. The SLD of each film is randomly chosen $\rho_f \in [0, 20]$.

Figure 4 shows five test SLD profiles recovered after a neural network is trained using such a dataset. The overall Mean Absolute Error for the test set, composed of 5K samples, lies around 0.08.

5.3. Training on Lamellar structures

The family of SLD profiles for this data set is defined by lamellar structures, each one having n_r equally spaced regions of alternating SLDs between ρ_1 and ρ_2. Both SLD values were chosen randomly between -10 and 10, and the number of regions was randomly chosen between 1 and 64, extending from $z = \Delta$ to $z = 512 - \Delta$, where $\Delta = 50$ defines a buffer to smoothen the transition between the lamellar sample and its surroundings. The obtained profiles were smoothened using a Gaussian filter of a width randomly chosen between 0 and 10.

Figure 4 shows five test SLD profiles recovered after a neural network is trained using such a dataset. The overall Mean Absolute Error for the test set, composed of 5K samples, lies around 0.98.

The SLD profiles predicted by the ANN are not always consistent with the target SLD profiles. However, when calculating the reflectivity curves produced by such predicted profiles, the curve obtained is quite similar to the original curve presented to the ANN. For this family of SLD profiles, the trained network is a good example of a pseudo-inverse that, due to the degeneracy of the problem, is recovering a plausible solution but not necessarily the correct one.

6. Discussion

The time required to generate each of the training data sets and to train each of the ANNs is rather short: around a couple of hours for data generation and a similar time for ANN training (provided a GPU is available). For unfortunate cases in which the data set is not suited for ANN training (because of the phase problem, for example), the overfitting regime may be reached rather soon and the training stops even in less than 30 minutes. The trained ANNs require only around 100 MB of space and the training data is at least around 500 MB large. Once trained, the ANNs are able to recover plausible SLD profiles from 5K reflectivity curves in around 0.5 sec. What these metrics tell, is that the ANNs are abstracting in an efficient way the information contained in the training data.

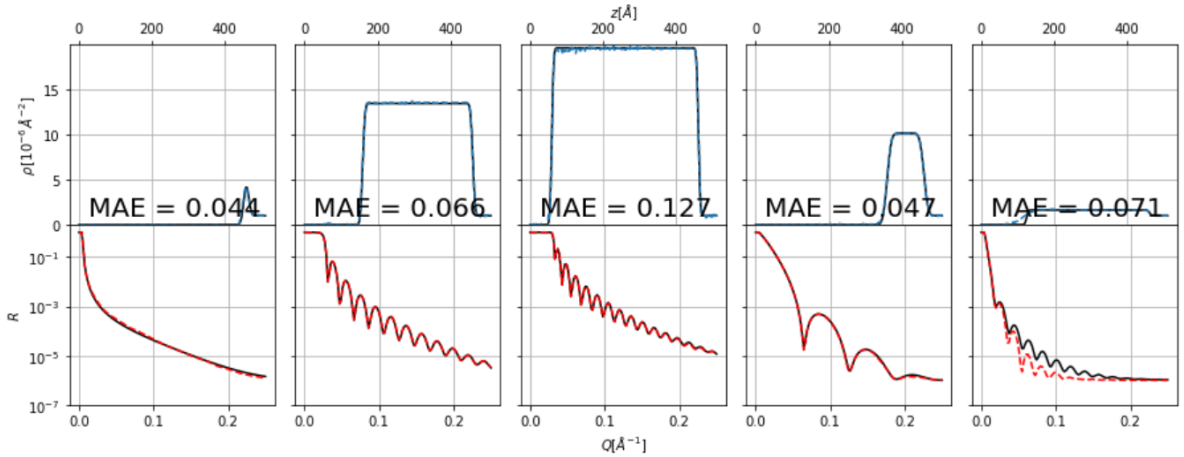

Figure 4. Single films are simple SLD profiles for which the trained network is able to correctly predict the associated SLD profiles. Five test reflectivity curves (black lines; bottom panels) were shown to the trained network for the first time; from those curves, the network guesses the SLD profiles that produce them (blue dashed lines; top panels). In the figure, the Mean Absolute Errors (MAE) with respect to the true SLD profiles (black lines; top panels) are also shown. Additional reflectivity curves are shown (red dashed lines, bottom panels), calculated from the SLD profiles predicted by the ANN.

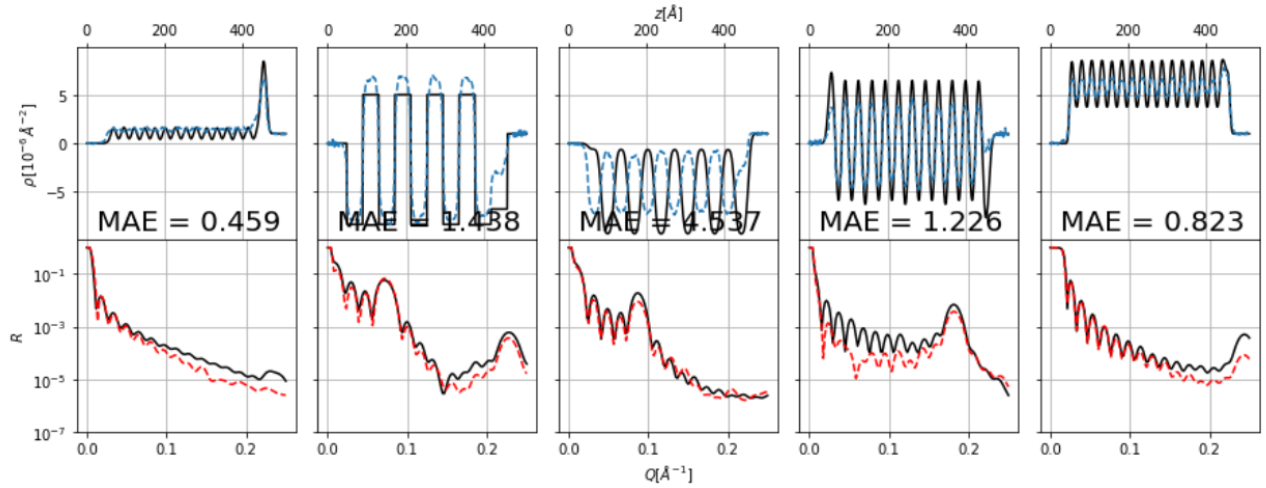

Figure 5. In the case of periodic lamellar SLD profiles, the phase problem is very present: even if most of the predictions recover to a close approximation the original reflectivity curves, the predicted SLD profiles differ from the actual targets. Five test reflectivity curves (black lines; bottom panels) were shown to the trained network for the first time; from those curves, the network guesses the SLD profiles that produce them (blue dashed lines; top panels). In the figure, the Mean Absolute Errors (MAE) with respect to the true SLD profiles (black lines; top panels) are also shown. Additional reflectivity curves are shown (red dashed lines, bottom panels), calculated from the SLD profiles predicted by the ANN.

A good question to ask now is whether the ANNs are actually learning the transformations or are only memorizing the associations between particular SLD profiles and reflectivity curves. While the evolution of the metrics during the training and the performance on the test set suggest that actual learning is taking place (See Appendix A.2), it must be stressed that the ANNs are certainly not learning a full general transformation, but are rather optimized for interpolating within the family of SLD profiles for which they are trained. Thus, in order to take advantage of the proposed approach in real experiments, the simulated training data should incorporate the instrument specifications with as much detail as possible. Additionally, in order to train an ANN robust against noise, for each target SLD profile, several noisy reflectivity curves with varying levels of noise should be simulated, effectively allowing the network to give higher priority to data points lying at lower Q values.

Due to the quick response of the trained ANNs, they could be incorporated in a data pipeline able to potentially provide preliminary analyses of real time phenomena like the swelling or drying of thin films

(See also [20]), or used in large batch analysis of huge data sets coming from large facilities. In fact, the time required for an ANN to invert 10K reflectivity curves is only around one second. However, as seen in Section 5.3, interpreting reflectivity curves in such a way, may offer plausible SLD profiles that nevertheless do not correspond to the actual sample under the beam, thus, a careful examination at a later stage will be always required and, more likely than not, the ANN-inferred SLD profiles should be instead used as starting models to feed traditional fitting methods.

The ANNs described in this work are certainly not useful resources for interpreting real reflectivity curves in terms of SLD profiles. In order to bring the proposed approach to the arena of real experiments, additional information about the instrument resolution, the actual Q range and some information about the sample would also be needed. However, the numerical experiments carried out here, show that the short training times and the small space required, could make it practical to train several ANNs with richer architectures, specialized in different instruments and selected families of samples. The results presented in this work should be taken as a baseline with which to compare future developments aimed at tackling the analysis of data coming from real experiments.

Acknowledgments

I would like to thank the BornAgain team at JCNS-MLZ, especially Joachim Wuttke for his healthy skepticism and criticism; and Alexander Schober for his optimistic encouragement. I would also like to thank, for the many fruitful discussions, Wojciech Potrzebowski from the ESS; Miguel González from the ILL; Alexandros Koutsioumpas, Jean F. Moulin, Gaetano Mangiapia, and Martin Haese from the MLZ. Their feedback and comments have certainly had a positive impact on the development and presentation of this work.

References

[1] Parratt L G 1954 *Physical review* **95** 359
[2] Penfold J and Thomas R K 1990 *Journal of Physics: Condensed Matter* **2** 1369–1412
[3] Tanner B 2018 *Handbook of Advanced Nondestructive Evaluation; Ida, N., Meyendorf, N., Eds* 1–34
[4] Majkrzak C, Berk N and Perez-Salas U 2003 *Langmuir* **19** 7796–7810
[5] Majkrzak C and Berk N 2003 *Physica B: Condensed Matter* **336** 27–38
[6] Pospelov G, Van Herck W, Burle J, Carmona Loaiza J M, Durniak C, Fisher J M, Ganeva M, Yurov D and Wuttke J 2020 *Journal of Applied Crystallography* **53** 262–276
[7] Nelson A R and Prescott S W 2019 *Journal of applied crystallography* **52** 193–200
[8] Cybenko G 1989 *Mathematics of Control, Signals, and Systems (MCSS)* **2** 303–314
[9] Funahashi K I 1989 *Neural networks* **2** 183–192
[10] Hornik K, Stinchcombe M, White H *et al.* 1989 *Neural networks* **2** 359–366
[11] Barron A R 1994 *Machine Learning* **14** 115–133
[12] Cohen N, Sharir O and Shashua A 2016 On the expressive power of deep learning: A tensor analysis *Conference on learning theory* pp 698–728
[13] Eldan R and Shamir O 2016 The power of depth for feedforward neural networks *Conference on learning theory* pp 907–940
[14] Telgarsky M 2016 Benefits of depth in neural networks *29th Annual Conference on Learning Theory* (*Proceedings of Machine Learning Research* vol 49) (PMLR) pp 1517–1539
[15] Lu Z, Pu H, Wang F, Hu Z and Wang L 2017 The expressive power of neural networks: A view from the width *Advances in neural information processing systems* pp 6231–6239
[16] Hanin B 2019 *Mathematics* **7** 992
[17] Tensorflow classification tutorial URL `https://www.tensorflow.org/tutorials/keras/classification`
[18] Li H, Schwab J, Antholzer S and Haltmeier M 2020 *Inverse Problems*
[19] Khan T A and Ling S H 2019 *Algorithms* **12** 88
[20] Greco A, Starostin V, Karapanagiotis C, Hinderhofer A, Gerlach A, Pithan L, Liehr S, Schreiber F and Kowarik S 2019 *Journal of applied crystallography* **52**
[21] Kingma D P and Ba J 2014 *arXiv preprint arXiv:1412.6980*

Appendix A. Appendix

Appendix A.1. Training on random SLD profiles

Throughout this work, we have argued that the ANN training process is sure to fail in a majority of cases, as the solution space usually has branches and many of the training targets may lie in inconsistent branches of the solution space, causing the weights of the ANN to drift in inconsistent directions during the training. To prove this point, an additional ANN was trained using a set of SLD profiles composed of a random number of layers of random thicknesses and random heights. Figure A1 shows five test SLD profiles recovered after a neural network is trained using such a dataset. None of the predicted SLD profiles is consistent with the targets and none of the recovered reflectivity curves is consistent with the input data. The Mean Absolute Error for the test set, composed of 5K samples, lies around 2.4.

Appendix A.2. Metrics Evolution

After the ANN architecture is defined, the ANN is trained using the training set to modify its weights and the validation set to asses the performance of the ANN; after each epoch, the MAE is computed. Its evolution is shown in Figure A2, which shows that, except for the family of random SLD profiles, the MAE continuously decreases throughout the epochs for both training and validation sets, and training stops at the moment at which the validation error seems not to decrease further. In contrast, for the family of random SLD profiles, overfitting starts already at epoch 10.

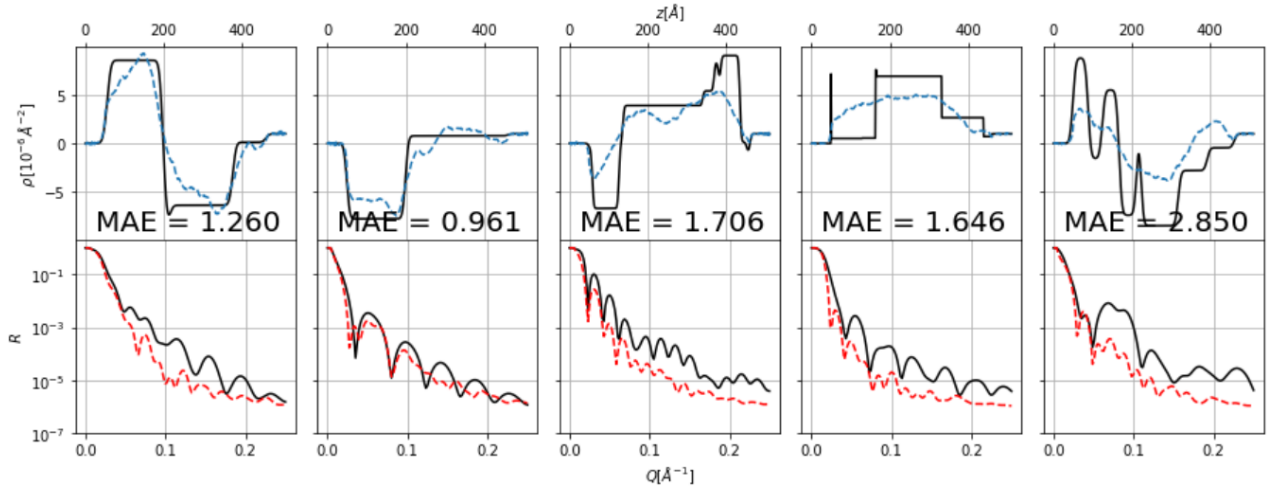

Figure A1. The training of the ANN fails when using a set of 100K random SLD profiles of between 1 and 16 slabs of random thicknesses and heights between -10 and 10. Five test reflectivity curves (black lines; bottom panels) were shown to the trained network for the first time; from those curves, the network guesses the SLD profiles that produce them (blue dashed lines; top panels). In the figure, the Mean Absolute Errors (MAE) with respect to the true SLD profiles (black lines; top panels) are also shown. Additional reflectivity curves are shown (red dashed lines, bottom panels), calculated from the SLD profiles predicted by the ANN.

One peculiarity that is observed in the three main data sets studied, is that the validation error is smaller than the training error. This counter-intuitive behavior comes from the dropout layer which, being activated only at training time, causes poorer performance on the training set –a condition that is rather perceived as better validation performance. To test this oddity, the dropout layer was removed while training an ANN over the single films data set and the expected relation between training and validation errors was recovered: $\mathrm{MAE}_{\mathrm{valid}} \geq \mathrm{MAE}_{\mathrm{train}}$ (See Figure A3).

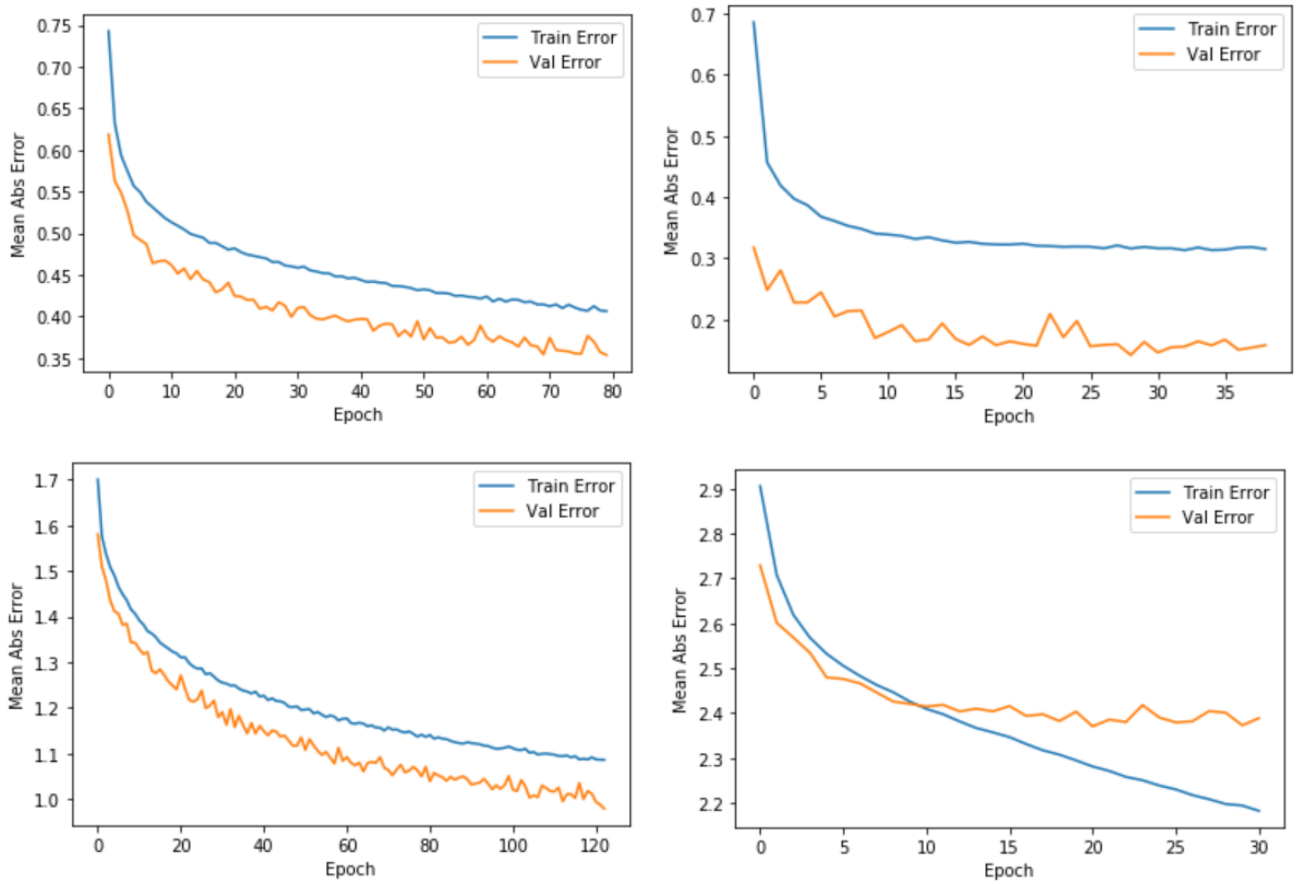

Figure A2. The MAE evolution throughout the epochs as the training proceeds is shown for the different families of SLD profiles studied in this work: Odd functions (top left panel), single films (top right panel), lamellar structures (bottom left panel) and random SLD profiles (bottom right panel). See the text for discussion.

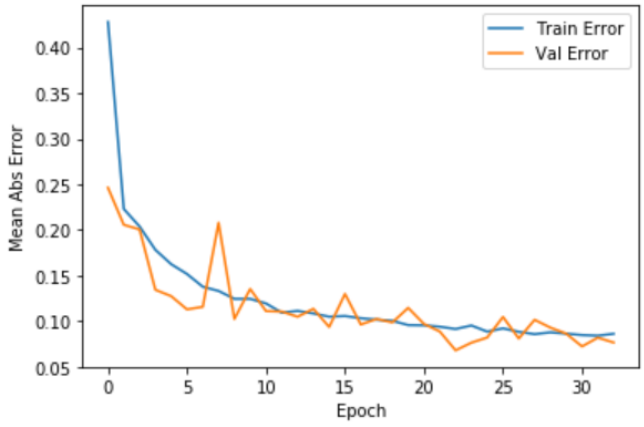

Figure A3. For the case of single films SLD profiles, the MAE evolution throughout the epochs as the training proceeds is shown for a special ANN in which the dropout layer is not present. See the text for discussion.