PROCEEDINGS OF SCIENCE

Regularization methods vs large training sets

Jaime Vega[1]
Departamento de Aceleradores, Instituto Nacional de Investigaciones Nucleares
P.O. Box 18-1027, Ciudad de México 11801, México
E-mail: jaime.vega@inin.gob.mx

Humberto Carrillo-Calvet
Laboratorio de Dinámica no Lineal, Facultad de Ciencias, Universidad Nacional Autónoma de México
Ciudad de México 04510, México
E-mail: carr@unam.mx

José Luis Jiménez Andrade
Laboratorio de Dinámica no Lineal, Facultad de Ciencias, Universidad Nacional Autónoma de México
Ciudad de México 04510, México
E-mail: jija@ciencias.unam.mx

Digital pulse shape analysis (DPSA) is becoming an essential tool to extract relevant information from waveforms arising from different source. For instance, in the nuclear particle detector field, digital techniques are competing very favorable against the traditional analog way to extract the information contained in the pulses coming from particle detectors. Nevertheless, the extraction of the information contained in these digitized pulses requires powerful methods. One can visualize this extracting procedure as a pattern recognition problem. To approach this problem one can use different alternatives. One very popular alternative is to use an artificial neural network (ANN) as a pattern identifier. When using an ANN, it is common to introduce a regularization method in order to get rid or at least to reduce the effects of overfitting and overtraining. In addition, another option that helps to solve these problems is to use a large training dataset to train the ANN. In this paper, we make an intercomparison of the advantage of regularization methods vs large training datasets when used as methods to reduce the overtraining and overfitting effects when training an ANN.

Artificial Intelligence for Science, Industry and Society, AISIS2019
October 21-25, 2019
Universidad Nacional Autónoma de México, Mexico City, México

[1]Speaker

© *Copyright owned by the author(s) under the terms of the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License (CC BY-NC-ND 4.0).*https://pos.sissa.it/

PoS(AISIS2019)028

1. Introduction

Hardware development of data acquisition systems has allowed the development of powerful digital pulse shape analysis (DPSA) techniques applied to multiple detection systems, see Refs. [1-8]. Additionally, ANNs represent an interesting alternative for the implementation of DPSA. ANNs are adaptive systems that exhibit some advantages for this task. One can use several ANN models to carry out DPSA, for example, different ANN architectures, as well as, different learning laws. On the other hand, two related ever-present problems in the diverse applications of ANNs as pattern recognizers (signal pulses) are overfitting and overtraining. Of course, there are several ways to deal with these two problems when using an ANN. In this paper, we will focus ourselves on the study of three common methods that people use to address the problem of overfitting-overtraining. This will be accomplished in relation to the problem of identifying the signals (pulses) coming from a Bragg curve spectrometer.

2. Bragg curve spectroscopy

Bragg curve spectroscopy (BCS) is a nuclear analytical technique that has been used, for several decades, to identify the ions coming out from nuclear reactions. In this technique, one uses an ionization chamber to measure two parameters: the total energy of the ion E^{Tot}, and the Bragg peak amplitude *BP*, which represents the maximum amplitude of the specific stopping power curve ($S(E) = dE/dx \equiv$ Bragg curve BC) of an ion, when it traverses the gaseous medium within the ionization chamber, losing all its energy. Traditionally, one obtains these two parameters by feeding the anode output signal from a BCS to two electronic amplification branches, one with a large integration time, providing the E^{Tot} signal, and the other with a short integration time, providing the *BP* signal [9-12]. A detailed description of a DPSA alternative appears in [8].

3. Digital pulse shape analysis and Bragg curve spectroscopy

For the identification of a Bragg curve, we will use the DPSA procedure followed in Ref. [8]. For that purpose, we will analyze digital synthetic BCs. These curves consist of 81 values, $\{S(i)\}_{i=0,\,80}$. In Fig. 1, it is shown an ideal BC (red curve) along with a synthetic BC (dotted black curve) which contains a fast-changing component simulating any source of experimental noise.

In order to appreciate how well an ANN is learning its assigned task, it is convenient to plot the evolution of the sum-of-squares error (SSE) functions (over the training and validation datasets) vs the number of ANN training epochs. One epoch means presenting once each one of the curves in the training dataset during the ANN training. Therefore, accordingly, we define these two error curves as:

$$E^T(\rho) = \sum_{p\in D^T} \left|\vec{y}^t - f\left[\vec{x}_p;\, \vec{\omega}(\rho)\right]\right|^2 \quad \text{and} \quad E^V(\rho) = \sum_{p\in D^V} \left|\vec{y}^t - f\left[\vec{x}_p; \vec{\omega}(\rho)\right]\right|^2,$$

where $\vec{\omega}(\rho)$ represents the weight array of the ANN after ρ training epochs, $f\left[\vec{x}_p;\, \vec{\omega}(\rho)\right]$ is the ANN output for pattern p after ρ training epochs, and the sum is carried over all patterns p in the training D^T and validation D^V datasets respectively.

PoS(AISIS2019)028

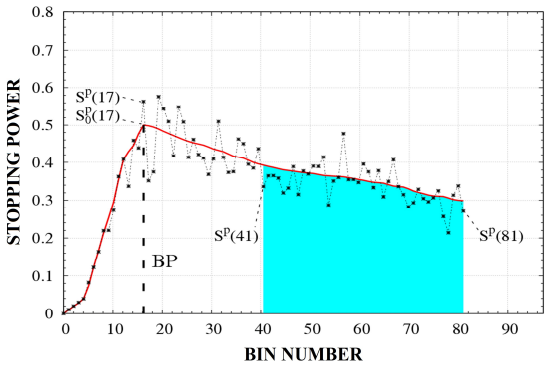


Fig. 1. Plot of an ideal BC (red solid curve) along with a synthetic BC (black dotted curve), which includes a zero mean fast component simulating any possible source of experimental noise. The fluctuating amplitude of the fast component (standard deviation) is equal to 10% of the corresponding ideal BC amplitude at every point; we call this quantity the noise to signal ratio N/S. Using this kind of BCs we construct the training dataset D^T as well as the validation dataset D^V.

4. ANN training

We used three different options to train the ANNs. In all three cases, we used the back-propagation training learning law [13]. In the first and the second cases we used a learning rate η = 0.3, a momentum term α = 0.15, and, as regularization method, we used early stopping. In the third case, we used a learning rate η = 0.3, and weight decay as regularization method as in Refs. [14, 15]. In the first and third cases, the datasets D^T and D^V consists of 45,100 BCs. In the second case, these two datasets consist of 451,000 BCs. According to this, in all cases we employed the stochastic gradient descent algorithm to update the weight array using, in the first and the second cases:

$$\Delta\omega_{ij}(\rho+1) = -\eta\frac{\partial E^T(\varrho)}{\partial\omega_{ij}} + \alpha\Delta\omega_{ij}(\rho).$$

In the third case, the weight array was updated without using a momentum term (α) but, instead, including a weight decay term (λ):

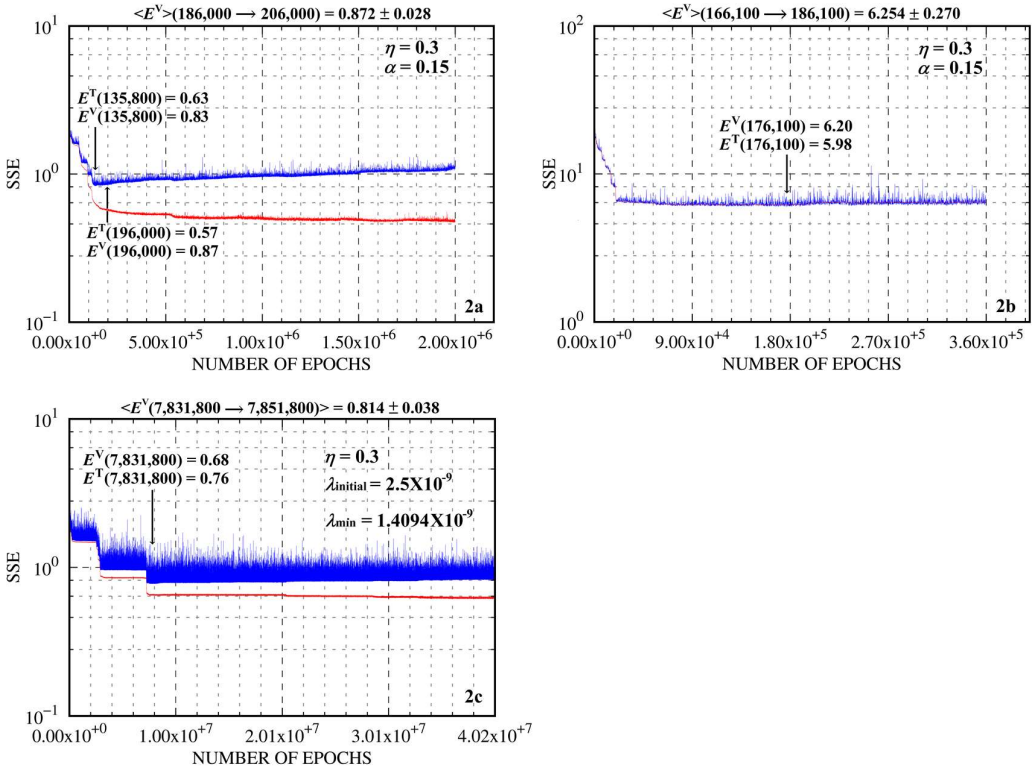
$$\omega_{ij}(\rho+1) = -\eta\frac{\partial E^T(\varrho)}{\partial\omega_{ij}} - \lambda\omega_{ij}(\rho).$$

5. Results

In Figs. 2a-c, we show the $E^T(\rho)$ (red curve) and $E^V(\rho)$ (blue curve) error curves for each one of the three studied cases. In Figs. 2a and 2b we used a back-propagation learning law with η = 0.3 and α = 0.15. In Fig. 2a, the datasets D^T and D^V consists of 45,100 BCs. In Fig 2b, these two datasets consist of 451,000 BCs. In Fig 2c, the learning law was back-propagation with weight decay, using a learning rate η = 0.3, an initial weight decay term, $\lambda = 2.5\text{x}10^{-9}$, and the datasets consists of 45,100 BCs. In each one of these three figures, we indicate the values of the training and validation error curves evaluated at the epoch number where the validation error curve reaches its minimum value. Additionally, on the top of the figures, we also present the corresponding average value around the validation minimum alone with its standard deviation.

PoS(AISIS2019)028

An important issue when using weight decay as a regularization method has to do with the way used to reduce the weight decay term, λ. As the ANN training progresses, it gets more complex, that means the absolute value of the weight array will grow, consequently one has to diminish the value of λ, allowing an effective control of the growth rate of $|\Delta\vec{\omega}(\rho)|$. The initial λ value is 2.5x10^{-9}. We started checking $|\Delta\vec{\omega}(\rho)|$ every 1,000 epochs, beginning at 2,000 training epochs, and comparing its current value with its value 1,000 epochs before. If $|\Delta\vec{\omega}(\rho)|$ changes less than 1%, we reduce the size of the weight decay term using a common ratio r = 0.984034. This common ratio r would reduce the initial λ value from 2.5X10^{-9} to 1X10^{-9} in ~57 steps. In Fig. 2c, the validation error curve minimum corresponds to λ_{min} = 1.4094X10^{-9} at 7,831,000 epochs.



Figs. 2a, 2b and 2c. We show the training $E^T(\rho)$ (red curve) and the validation $E^V(\rho)$ (blue curve) SSE funtions for each one of the three studied cases. We indicate the epoch numbers where the minimum of the validation error curve lies, and the corresponding values of the training and validation error curves. On the top of the figures, we show the average value of the validation error curve around its minimum value together with the standard deviation of the 21 values used to obtain the $E^V(\rho)$ average value. In Figs. 2a and 2b we report the value of the learning rate (η) and the momentum term (α). In Fig. 2c we report the value of the learning rate (η), the initial weight decay term ($\lambda_{initial}$), and the weight decay term at the minimum (λ_{min}).

In Fig. 2a, we present the values of the error curves at two different epochs. In this figure, one can easily see that the minimum lies at 135,800 epochs. The problem is, at that point, the quality of the corresponding scatter plot is not yet acceptable, because the ANN has not been able already to learn well the 41 classes belonging to the smallest BP value. It happens that, after training the ANN some additional epochs, we still get a reasonable small validation error and,

PoS(AISIS2019)028

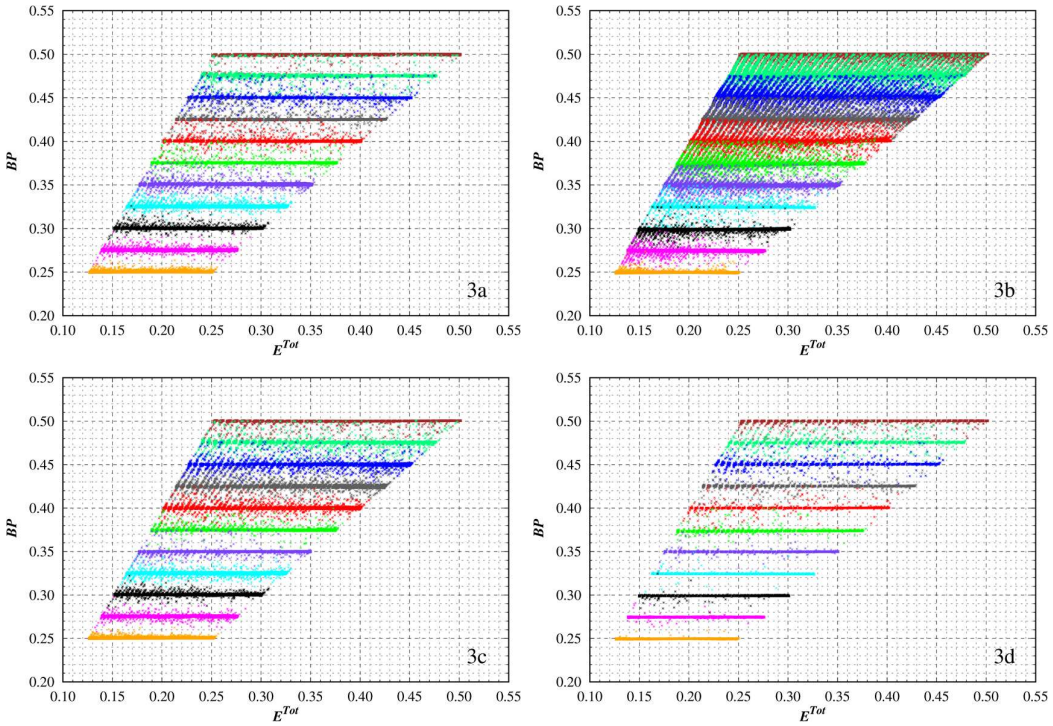
now, the quality of the scatter plot is acceptable, see Ref. [8]. From now on, we will use this second epoch number, 196,000, as the best option for the first case.

In Figs. 2a, 2b and 2c one can see that the option that reaches the smallest minimum value in its validation curve is Fig. 2b. In order to get to this conclusion, one has to realize that in the second case we used 451,000 BCs. That means, 10 time more BCs than in the other two cases and, since we are dealing with the total sum-of-squares errors, then we ought to rescale down the minimum value of the second case by a factor of 10 before comparing it with the other two minimum values. Therefore, when we rescale down the value of the second case, we get an error of 0.6254337 ± 0.02702884.

In order to get rid of local fluctuations present in the E^V error curves, we report average minimum E^V values. We obtained these average values using an averaging window equal to 20,000 epochs, containing 21 equidistant points 1,000 epochs apart and centered on the minimum value. See Table 1.

Table 1
Average minimum values of the E^V error curves for each one of the three studied cases, together with the intervals we used to evaluate the three E^V average values.

Case	$<E^V>$ rescaled minimum value	$<E^V>$ averaging interval (epochs)	% above 2nd case
First	0.8723767 ± 0.02778259	186,000 - 206,000	39.5
Second	0.6254337 ± 0.02702884	166,100 - 186,100	
Third	0.8142579 ± 0.03826261	7,831,800 - 7,851,800	30.2

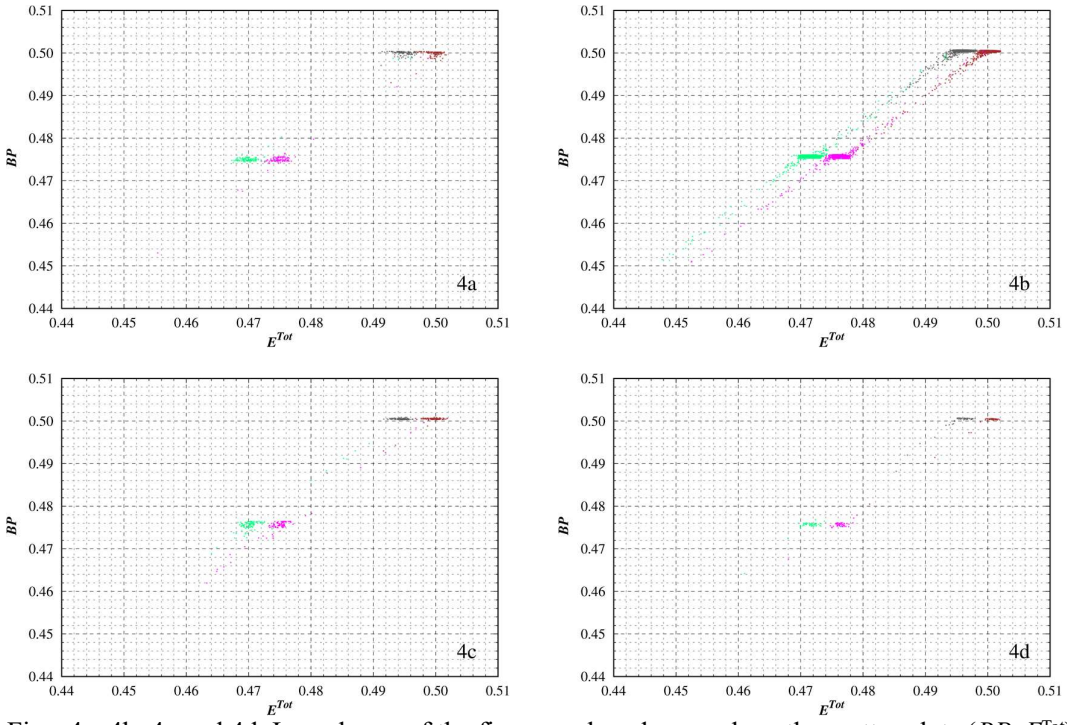


Figs. 3a, 3b, 3c and 3d. In each one of the scatter plots a, b and c, we show all the data points (*BP*, E^{Tot}) belonging to the three cases. Figure 3d belongs to the second studied case where the dataset contains 451,000 BCs, although, for comparison reasons, we are just plotting one tenth of the total number of points, i.e., 45,100. In this way, we can compare this scatter plot with the other two

PoS(AISIS2019)028

scatter plots exactly on the same footing. Once we take into account this fact, we can clearly see that the scatter plot of the second studied case is the one with the best quality, because there is less interclass overlapping of the scatter points.

In Figs. 3a, 3b and 3c, we show the scatter plots, Bragg peak amplitude vs Total energy (*BP* vs E^{Tot}), predicted by the trained ANNs for the three studied cases. We obtained these scatter plots at the number of epochs where each one of the E^V curves reaches its minimum value. Apparently, the scatter plot displayed in Fig. 3b, belonging to what we consider the best option, seems to be of an inferior quality as compared with Figs. 3a and 3c. However, as before, this first impression is due to the fact that scatter plot 3b contains 1,000 patterns per class, i.e., there are 451,000 BCs in its validation dataset. The way to get around this situation is to display the same number of points in the three scatter plots, i.e., 45,100 points. Therefore, when we do that for our second studied case, we obtain the scatter plot displayed in Fig. 3d. Now, one can easily see that indeed the second case is our best option. When comparing Fig. 3d with Figs. 3a and 3c, one can see that interclass overlapping is reduced in the former case. To make this assertion more obvious, in Figs. 4a, 4b, 4c, and 4d, we present these same data as in Figs. 3a-d scatter plots, but now by zooming in on the scatter plots, and including only four classes. The four classes presented belong to the two largest BP values and to the two largest total energy values. One can clearly see that in Fig. 4d (second studied case) there is less interclass overlapping, what reflects itself in its validation error curve $E^V(\rho)$ having a smaller minimum value.



Figs. 4a, 4b, 4c and 4d. In each one of the figures a, b and c, we show the scatter plots (*BP*, E^{Tot}) including only four classes, corresponding to the two largest *BP* values and to the two largest total energy values E^{Tot}. Again, to put the comparison on the same footing, in Fig. 4d, for the second studied case, we display the scatter plot belonging to the same four considered classes, although, now, including only 100 scatter points per class. One can easily see, in Fig. 4d, that the points in the four depicted classes pack themselves more closely together than in Figs. 4a and 4b.

PoS(AISIS2019)028

Another remarkable aspect that one can see when comparing Fig. 4a with Figs. 4c and 4d, is that Fig. 4a presents the largest interclass overlapping. This observation is consistent with the fact that Fig. 4a belongs to the validation error curve with largest minimum value; see Table 1. In a similar fashion, Fig. 4c, shows and intermediate class overlapping, and possesses an intermediate minimum value in its validation error curve, see Table 1. This observation clearly exemplifies that the size of the minimum value of the validation error curve correlates to the overlapping extent of the different considered pattern classes. In addition, in Fig. 4d, one can see the points of the four displayed classes more closely packed as compared with the points in Figs. 4a and 4b. This packing effect also helps the validation error curve to reach a smaller minimum value.

In reference [16], it was concluded that the concomitant experimental noise that accompanies and distorts the BCs, sets limits to the learning capability of the ANN used as a pattern identifier. In other words, the minimum value reachable by the validation error curve grows with the extent of the noise size coming along with the BCs. It happens that way, because once the ANN gets to its minimum validation value, during the learning process, it starts learning the noise embodied in training dataset signals rather than the remaining still unlearned small and subtle real signal features, overshadowed by a larger noise. Since the noise component present in the training dataset is different to the one present in the validation dataset, it normally happens that the training error curve $E^T(\rho)$ keeps on decreasing after the minimum of the validation error curve $E^V(\rho)$ is reached, while this later starts growing larger. This is precisely the onset of overtraining. In brief, this is the reason why the size of the noise to signal ratio, N/S, limits the amount of feature extraction that the ANN may learn from a training dataset.

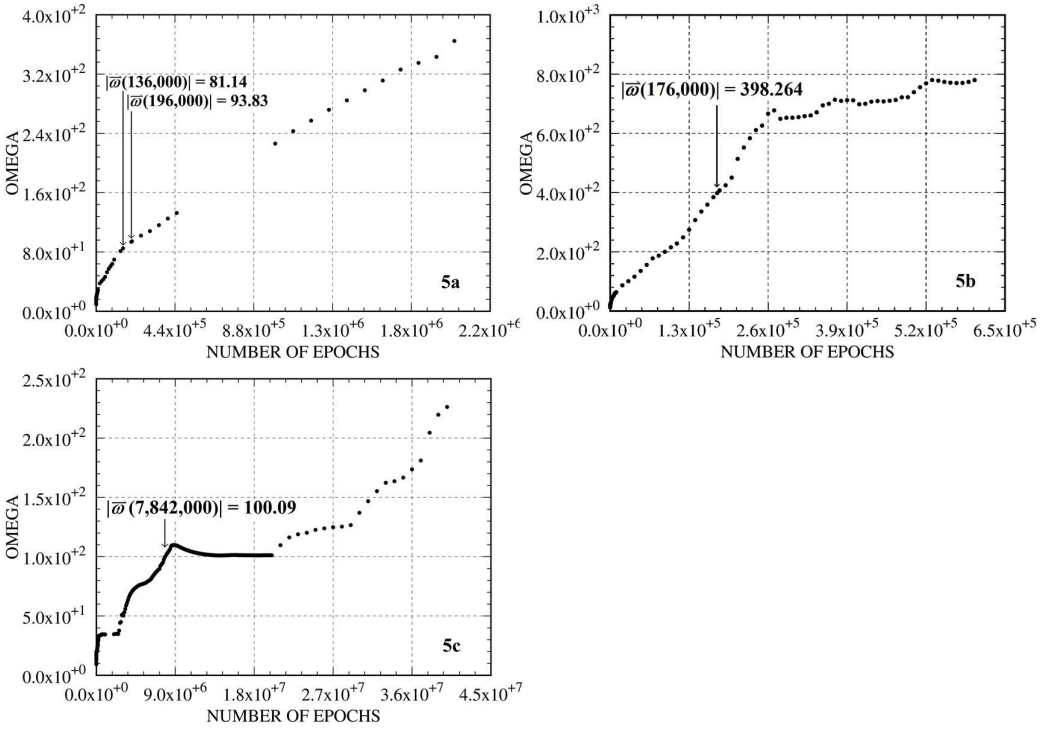
As has been mentioned in Refs. [14, 15 and 17], in order that an ANN can learn more thoroughly complex patterns (the subtle features), it requires letting the absolute value of its weight array to grow during its learning or adaptation process; allowing the ANN to become a more complex model. On the other side, it is also true that the noise learning present in the BCs, also requires a more complex model, i.e., an ANN with larger weight values. The question is, will the ANN use the additional complexity to learn either subtle real signal features or the signal noise? In order to help to better understand the consequences of a more complex system, in Figs. 5a, 5b, and 5c, we plot the absolute value of the weight array $|\vec{w}(\rho)|$ vs the number of training epochs, ρ, for the three studied cases.

Table 2.
Absolute values of the weight array $|\vec{w}(\rho)|$ at the minimum of the validation error curve $E^V(\rho)$ for the three studied cases.

Case	ρ	$\lvert\vec{\omega}(\rho)\rvert$
1	196,000	93.8343
2	176,000	398.264
3	7,842,000	100.088

When, realizing that a large $|\vec{\omega}(\rho)|$ implies a rather complex system, one is tempted to conclude that the large observed $|\vec{\omega}(\rho)|$ value, in the second case, might spring from the fact that the ANN is adapting to the noise present in the BC, rather than to the still unlearned subtle pattern features. However, the reason why that does not occur in this case, owes to the fact that the training

PoS(AISIS2019)028



Figs. 5a, 5b, and 5c. Plots of the absolute value of the weight array $|\vec{w}(\rho)|$ vs number of training epochs, ρ, for the three studied cases. We indicate the $|\vec{w}(\rho)|$ values at the corresponding minimum of the validation error curve.

dataset is huge (451,000 BCs), thus making quite difficult for the ANN to learn or to adapt its weight array value to the noise present in all the patterns. Under these circumstances, the ANN has the chance to grow its $\vec{\omega}$ array, becoming a more complex model, capable of learning or extracting small and subtle real features present in the patterns or BCs, instead of learning or adapting to the concomitant noise. Here, it is good to remark that the same small and subtle real pattern features are present in all patterns, allowing the ANN to learn them rather than the noise with a similar amplitude but changing from pattern to pattern. That explains why, in this case, the ANN can reach a smaller minimum value for the validation error curve (after rescaling down the value by a factor of 10), allowing it to still learn the subtle features present in the patterns instead of leaning the noise present in them. Noise learning for this huge training dataset would require an even larger $|\vec{\omega}(\rho)|$ value, a more complex model. In fact, that does not seem to be happening, in the second case, even at 360,000 epochs, see Fig. 2b. One can see that the $|\vec{\omega}(\rho)|$ value keeps on growing after 176,000 epochs, see Fig. 5b, although the training and validation error curves remain close together, see Fig. 2b, indicating the absence of data overfitting and overtraining. One can explain this observation in terms of the system complexity as follows: when one moves from a 45,100 BCs to a 451,000 BCs dataset, we are increasing the system complexity due to the concomitant noise that accompanies the BCs, the noise is different in all BCs. Then, when dealing with a large dataset, the model complexity of the ANN will be devoted to extract the true features present in the BCs, because their complexity is smaller than the complexity associated with the noise. In this way overfitting and overtraining will be prevented.

PoS(AISIS2019)028

When comparing the first and the third cases, we see that the values of $|\vec{\omega}(\rho)|$, calculated at the minimum of their validation error curves, is larger for the third case. That means that, inasmuch as the weight decay term has controlled the growth of $|\vec{\omega}(\rho)|$ at a very small pace, its associated ANN has had the chance to adapt more smoothly to the data, implying a better search for the minimum of the $E^V(\rho)$ error curve in the third case, i.e., the additional model complexity of the third case (larger absolute value of the weight array) allowed it to extract more real signal from the dataset when compared to the fisrt case. In other word, the weight decay term helps the model complexity of the ANN to adapt itself a little bit better to the small and subtle real features rather than to the noise, preventing or at least diminishing the occurrence of overfitting, i.e., learning noise. In these two cases, the consequence of preventing or diminishing the occurrence of overfitting is clearly observe when comparing Fig. 2a and Fig. 2c.

If one looks at Fig. 5a (first case), one can see that, around 196,000 epochs, its $|\vec{\omega}(\rho)|$ value presents evidence of a steady and uncontrolled growth. Due to the regularization effect of the weight decay term, in the third case, this uncontrolled growth is reduced considerably, see Fig. 5c, after it reaches its minimum value. That causes the ANN, in the first case, to start extracting or learning noise from its dataset, preventing its validation error curve reaching a smaller minimum value. One can observe that this uncontrolled growth of the validation error curve is also present in some extent in our best option, second case, Fig. 5b, meaning its ANN might already be learning noise. In the first place, this observed behavior, see Fig. 5b, occurs after the ANN has had the chance to extract or learn smaller and subtle real signal features from its dataset in comparison to Fig. 5a. This allow us to explain the better quality of its scatter plot, Fig. 3d, evaluated at its minimum. On the other hand, the corresponding $E^T(\rho)$ and $E^V(\rho)$ error curves for case two, Fig. 2b, remain quite close together. That means that the ANN, in the second case, is not overtraining after it has reached its minimum value at 176,100 training epochs. Perhaps, this is an indication that the ANN complexity is growing but the noise complexity is so big that the ANN has not been able at all in learning any noise component present in the training dataset, thus explaining why both error curves still remain close together.

6. Discussion

In order to make a characterization of overfitting and overtraining we define two sources of the system complexity or dataset complexity. One source is associated with the intrinsic ideal BC shape (intrinsic complexity) and the other is associated with the concomitant noise accompanying the BCs signals (noise complexity). If, at some point during the training process, it happens that a part of both complexities are comparable, then overfitting will occur eventually. Once the minimum of the $E^V(\rho)$ curve is reached, overtraining will show up if training continues. Since noise complexity grows as the dataset size grows, it happens that when the dataset is quite large (451,000 BCs), i.e., noise complexity is larger than the instrinsic complexity of the BCs, then the ANN, during a considerable number of training epochs, will be able to learn many of the BC features without overfitting the noise component, because the ANN or model complexity is still not big enough to start learning the noise, and by the same token overftraining will not show up. Once the ANN has learned most of the extractable BCs features, if training continues, the ANN or model complexity will increase, growing the absolute value of its weight array, trying to match its complexity to the one required to learn the noise present in the BCs of the training dataset. In relation to this, in case two, the $E^T(\rho)$ and $E^V(\rho)$ error curve remains close together up to 360,000

PoS(AISIS2019)028

training epochs, see Fig. 2b, then one concludes that the ANN has not been able to adapt itself in any extent to the noise present in the training dataset and, consequently, no overfitting has occurred yet.

In regard to weight decay regularization, we realize that the weight decay term provides us the opportunity to search carefully for the validation error curve minimum and, at the same time, controlling the increasing ANN or model complexity very effectively. Nevertheless, there is a limit to this, it is has to do with the fact that learning smaller subtle real signal features requires increasing the model complexity, i.e., the size of the weight array, which is precisely what the weight decay term controls, by letting it to increase slowly. In the end, the weight decay performance exceeds the simple back-propagation with early stopping regularization, but it is not capable of matching the results obtained when using a larger training dataset.

Summarizing, small subtle real signal features requires two things to be learned:

a) To increase the system model complexity carefully allowing the ANN to be able to learn those small and subtle real signal features.

b) To provide the ANN the opportunity of using an increasing model complexity to learn the small and subtle real features rather than the noise that comes along with the signal.

To accomplish restraint a) we used a weight decay term to control the weight array size growth rate. To implement restraint b), we increased the size of the training dataset by a factor of ten making noise learning more difficult.

When using weight decay as a regularization method, definitively we do minimize or delay noise learning. Weight decay effectively controls the growth of the weight array size, allowing the ANN a more careful search for the validation error curve minimum. The problem is that to learn smaller and subtle real signal features will eventually require a more complex model, i.e., one with a larger absolute value of the weight array, and, at that point, noise learning will start overshadowing the small and subtle real feature extraction. As a matter of fact, when using weight decay as a regularization method, long before the validation error curve minimum is reached, overfitting starts. For instance, one can see in Fig. 2c that both error curves, $E^T(\rho)$ and $E^V(\rho)$, begin to split apart at ~ 250,000 training epochs, $E^T(\rho)$ getting smaller than $E^V(\rho)$ at a faster rate, indicating that overfitting is taking place.

Using a larger training dataset has the advantage of allowing the increment of the ANN or model complexity in order to learn the small and subtle real signal features, and, at the same time, preventing noise learning while the ANN learns the small and subtle real features. That is why a larger training dataset becomes a better regularization method than using weight decay.

Finally, we would like to comment on two related issues relevant to the present analysis: overtraining and overfitting. According to Ref. [18], the overfitting problem refers to exceeding some optimal ANN size or model complexity, while overtraining refers to exceeding the number of training epochs required to train an ANN and start destroying the predictive ability of the network. In this context, overfitting relates to the model complexity used to fit the training dataset. In ANN applications, it means using an ANN with more parameters than those justifiable by the dataset. In these circumstances, eventually, the additional complexity will start extracting noise from the data, rather than any possible signal feature not extracted yet. A regularization method like weight decay may help to control the increasing complexity of a large ANN. In such a case, the idea is to find out if this ANN complexity control may help to extract the remaining smaller

PoS(AISIS2019)028

and subtle real signal features rather than noise. When comparing Fig. 2a and Fig. 2c, we see that indeed weight decay regularization effectively controls the ANN complexity, performing a better signal feature extraction, what reflects itself in a smaller minimum value in its validation error curve, Fig. 2c.

Overtraining does not relate to the ANN or model complexity in relation to the training dataset. It relates to the presence of a signal noise that overshadows the smaller and subtle unlearned signal features. Increasing the size of the training dataset demands a larger ANN complexity to be able to learn the noise. At this point, it is opportune to realize the following: i) the signal noise is different for every pattern or BC belonging to the training dataset; ii) the underlying or intrinsic ideal BC shape is always the same for all BC belonging to the same class in the training dataset. Due to these two remarks, it is easy to see that the required model complexity to learn the signal noise increases with the size of the dataset, while the required complexity to learn the intrinsic features of the ideal BC remains the same. From this, we conclude that increasing the size of the training dataset delays the onset of overtraining, allowing the ANN the chance to learn even smaller and subtle real signal features. This is, when using a large training dataset, we are using the growing model complexity (due to the increasing effective number of parameters with training) to learn the smaller real signal features rather than noise more effectively. That is why our best option corresponds to our second studied case, i.e., when the dataset increased from 45,000 BCs to 451,000 BCs, compare Fig. 2b with Figs. 2a and 2c.

7. Conclusions and future work

The results of this study, following Refs. [14, 15 and 17], emphasizes the relevance to optimize the model complexity in order to achieve the best generalization. In this sense, one has to carefully decide the best option to optimize the ANN complexity used to solve a specific task. But the other relevant aspect that one should keep in mind is that another way to warrant a better generalization capability of the ANN is to use a large training dataset. In fact, the model complexity (ANN size) and the size of the training dataset (system complexity) ought to be selected in a mutually constrained way, trying to get a reasonable generalization capability.

In this study, we used BCs datasets with a noise to signal ratio, N/S, equal to 10%. For a future work, it would be interesting to perform new comparative studies, using different values of the N/S ratio, to see if the observed noise effect prevails or modifies itself in some extent. Also, for a future work, one could increase the ANN complexity by increasing the number of units or the number of layers that constitute the ANN (structural stabilization). The question is "will the additional gain of ANN complexity be used to learn the smaller and subtle real signal features yet unlearned, or it will be used to learn signal noise?"

Acknowledgements

We gratefully acknowledge the valuable computational support of ABACUS-Centro de Matemática Aplicada y Cómputo de Alto Rendimiento of Cinvestav-IPN for making this work possible.

References

[1] G. Benato, V. D'Andrea, C. Cattadori and S. Riboldi, *Improvement of the GERDA Ge Detectors Energy Resolution by an Optimized Digital Signal Processing*, *Physics Procedia* **61** (2015) 673.

PoS(AISIS2019)028

[2] E. Calore, D. Bazzacco and F. Recchia, *Pulse shape analysis for segmented germanium detectors implemented in graphics processing units, Nuclear Instruments and Methods in Physics Research. A* **719** (2013) 1.

[3] J. L. Flores, I. Martel, R. Jiménez, J. Galán and P. Salmerón, *Application of neural networks to digital pulse shape analysis for an array of silicon strip detectors, Nuclear Instruments and Methods in Physics Research A* **830** (2016) 287.

[4] A. B. Garnsworthy, C. J. Pearson, D. Bishop, B. Shaw, J. K. Smith, M. Bowry, V. Bildstein, G. Hackman, P. E. Garrett, Y. Linn, J. P. Martin, W. J. Mills and C. E. Svensson, *The GRIFFIN data acquisition system, Nuclear Instruments and Methods in Physics Research A* **853** (2017) 85.

[5] R. Jiménez, M. Sánchez-Raya, J. A. Gómez-Galán, J. L. Flores, J. A. Dueñas and I. Martel, *Implementation of a neural network for digital pulse shape analysis on a FPGA for on-line identification of heavy ions, Nuclear Instruments and Methods in Physics Research A* **674** ((2012) 99.

[6] I. Mateu, P. Medina, J. P. Roques and E. Jourdain, *Simulation of the charge collection and signal response of a HPGe double sided strip detector using MGS, Nuclear Instruments and Methods in Physics Research A* **735** (2014) 574.

[7] U. Rizwan, A. B. Garnsworthy, C. Andreoiu, G. C. Ball, A. Chester, T. Domingo, R. Dunlop, G. Hackman, E. T. Rand, J. K. Smith, K. Starosta, C. E. Svensson, P. Voss and J. Williams, *Characteristics of GRIFFIN high-purity germanium clover detectors, Nuclear Instruments and Methods in Physics Research A* **820** (2016) 126.

[8] J. J. Vega and R. Reynoso, *Application of neural networks to pulse-shape analysis of Bragg curves, Nuclear Instruments and Methods in Physics Research B* **243** (2006) no. 1 232.

[9] C. R. Gruhn, M. Bini, R. Legrain, R. Loveman, W. Pang, M. Roach, D. K. Scott, A. Shotter, T. J. Simons, J. Wouters, M. Zisman, R. Devries, Y. C. Peng and W. Sondheim, *Bragg Curve Spectroscopy, Nuclear Instruments and Methods in Physics Research* **196** (1982) no. 1 33.

[10] A. Moroni, I. Iori, L. Z. Yu, G. Prete, G. Viesti, F. Gramegna and A. Dainelli, *Position Sensitive and Bragg Curve Spectroscopy Detector System, Nuclear Instruments and Methods in Physics Research* **225** (1984) no. 1 57.

[11] K. E. Rhem and F. L. Wolfs, *A Focal Plane Detector for Reactions with Medium Weight Projectiles, Nuclear Instruments and Methods in Physics Research A* **273** (1988) 262.

[12] M. F. Vineyard, B. D. Wilkins, D. J. Henderson, D. G. Kovar, C. Beck, C. N. Davids and J. J. Kolata, *Performance of a Large Bragg-Curve Spectrometer", Nuclear Instruments and Methods in Physics Research A* **255** (1987) no. 3 507.

[13] A. Zell, G. Mamier, M. Vogt, N. Mache, R. Hübner, S. Döring, K. Hermann, T. Soyez, M. Schmalzl, T. Sommer, A. Hatziegeorgious, D. Posselt, T. Schreiner, B. Kett, G. Clemente, J. Wieland, J. Gatter, M. Reckzo, M. Riedmiller, M. Seemann, M. Ritt, J. DeCoster, J. Biedermann, J. Danz, C. Wehrfritz, R. Werner, M. Berthold and B. Orsier, *SNNS-Stuttgart Neural Network Simulator, Version 4.2*, University of Stuttgart, Institute for Parallel and Distributed High Performance Systems, and University of Tübingen, Wilhem-Schickard-Intitute for Computer Science 1998.

[14] C. M. Bishop, *Neural Networks for Pattern Recognition,* Oxford University Press 1995.

[15] C. M. Bishop, *Pattern Recognition and Machine Learning,* Springer 2006.

[16] J. J. Vega, R. Reynoso and H. Carrillo-Calvet, *Learning limits of an artificial neural network, Revista Mexicana de Física* **S54** (2008) no. 1 22.

[17] J. E. Moody, *The effective number of parameters: An analysis of generalization and regularization in nonlinear learning systems,* in Proceedings of *NIPS4, San Mateo, CA,* (1992) 847.

PoS(AISIS2019)028

[18] I. V. Tetko, D. J. Livingstone and I. Luik, *Neural networks studies. 1. Comparison of overtraining and overfitting, Journal of Chemical Information and Computer Sciences* **35** (1995) 826.

PoS(AISIS2019)028