Contents lists available at ScienceDirect

Radiation Physics and Chemistry

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

Performance of machine learning algorithms on neutron activations for Germanium isotopes

Rihab Gargouri [a,*], Serkan Akkoyun [b], Ramzi Maalej [a], Kamel Damak [a]

[a] LaMaCoP, Faculty of Sciences of Sfax, University of Sfax, 3018, Sfax, Tunisia
[b] Department of Physics, Faculty of Sciences, Sivas Cumhuriyet University, Sivas, Turkey

ARTICLE INFO

Handling Editor: Dr. Chris Chantler

Keywords:
Neutron-induced reaction
Cross-section
Supervised machine learning
Artificial Neural Networks
K-Nearest Neighbors
Support Vector Machines

ABSTRACT

In the studies of nuclear physics, one of the important parameters for nuclear reactions is the reaction cross-section. It can be obtained from experimental data or by different theoretical models. In this study, we implement machine learning (ML) algorithms to the regression analysis of the nuclear cross-section of neutron-induced nuclear reactions of Germanium isotopes. The data for the training of the machine was borrowed from the TENDL-2019 library for the total cross-section data of possible nuclear reactions after the bombardment of the different target materials by neutrons. Three ML models, Artificial Neural Networks (ANN), K-Nearest Neighbors (KNN) and, Support Vector Machines (SVM), were developed to fit nuclear data from the TENDL-2019 database in order to predict neutron induce reaction cross sections. The performance of each algorithm is determined and compared by evaluating the mean square error (MSE) and the correlation coefficient (R^2). According to the results obtained, we demonstrate that cross-section information can be obtained safely with ML techniques and the regression curve generated by our models is in good agreement with the evaluated nuclear data library. From our study, ANN and KNN are found to be better compared to SVM algorithm. ML models can enhance classical physics-guided models and play a role in nuclear data analyses. They can be used as an alternative to the estimation of cross-sections for neutron energies of an unknown energy value.

1. Introduction

The neutron-induced reactions at low and medium energies have great importance in nuclear physics studies such as fusion and medical applications (Hamid et al., 2021). Because of the easy production of a high-intensity beam, the low stopping compared to heavier particles, relatively high cross-sections, and the production of high-intensity neutrons via break-up, one of the most important particles is neutron (Tárkányi et al., 2020). The accurate knowledge of the neutron-induced reaction cross-sections is important for selecting and validating the best structural materials for fusion studies. The accuracy of neutron modeling and theoretical techniques strongly depends on the quality of the nuclear data. These data are generated in the form of libraries such as the Evaluated Nuclear Data File (ENDF) (Chadwick, n.d.), the Joint Evaluated Fission and Fusion (JEFF) (Plompen, 2020), or the Japanese Evaluated Nuclear Data Libraries (JENDLs) (Shibata et al., 2011) combining data for fission product yields, energy differential cross-sections, covariance for neutron cross section, product energy-angle distributions, etc. However, in certain energy ranges, experimental nuclear cross-sections for some nuclear reactions are not available due to experimental difficulties. Compared to the database for proton-induced reactions, the neutron-induced reactions database was relatively poor (Tárkányi et al., 2021). In recent years, Machine Learning (ML) and Artificial Intelligence (AI) have been very popular spanning a wide range of scientific applications. The main idea behind ML algorithm is to learn the pattern from training a model with a pre-existing dataset and successfully make accurate and powerful predictions.

The investigation of neutron interaction with Ge yields important information for the study of nuclear forces, nuclear structure, and nuclear reaction. Therefore, it is important to measure the cross-sections of Ge isotopes induced by neutrons. Few experimental data for Ge isotopes exist, but a survey of available cross-section data showed that existing data cover a wide energy range but they also show many discrepancies (Haque et al., 2009). Lederer-Woods et al., measured the ^{73}Ge (n, γ) reaction cross-sections experimentally (Lederer-Woods et al., 2019). Iwamoto et al., evaluated the neutron cross-sections for ^{70,72,73,74,76}Ge isotopes (Iwamoto et al., 2005). Meierhofer et al.,., experimentally

[*] Corresponding author.
E-mail address: rihab.gargouri.etud@fss.usf.tn (R. Gargouri).

https://doi.org/10.1016/j.radphyschem.2023.110860
Received 9 December 2022; Received in revised form 16 February 2023; Accepted 20 February 2023
Available online 25 February 2023

measured thermal neutron capture cross-sections of ^{74}Ge and ^{76}Ge isotopes (Meierhofer et al., 2010). In this study, the reaction cross-sections of the isotopes ^{70,72,73,74,76}Ge with different neutron energies were estimated using machine learning technique. Pedro Vicente-Valdez et al., used a variety of ML such as K-Nearest Neighbors (KNN) and Decision Trees (DT) to predict total neutron-induced reactions cross-section for ^{233}U and ^{35}Cl (Vicente-Valdez et al., 2021). Z.M.Niu et al., predicted the nuclear mass using Bayesian neural network (BNN) approach (Niu and Liang, 2018). In our work, we applied three different algorithms which are the artificial neural network (ANN), K-Nearest Neighbors (KNN) and, Support Vector Machines (SVM). We tried to reach our goal by using ML methods and a small number of input parameters. According to the results obtained from the ML calculations for five Ge isotopes performed using the dataset obtained from the TENDL-2019 database (Koning et al., 2019), it was concluded that these techniques are alternative tools for this purpose. This shows potential in building a ML model for generating nuclear cross-section data for both well-studied and understudied nuclear reactions.

2. Materials and method

The accuracy of neutron modeling depends on the quality of the nuclear data. Data libraries are generated by evaluators combining physics-based model codes and experimental data. Here, we define the computational cross-section datasets from TENDL-2019 (Koning et al., 2019). TENDL-2019 is one of the most complete nuclear data libraries used to simulate the nuclear reaction that involves protons, photons, neutrons, deuterons, and alpha particles with incident energy up to 200 MeV. In this study, although data filtering or splitting data ranges can improve performance in the range where there is a change of 10^7 factors in cross-sections from 0 to 200 MeV energy for the reactions, we performed our calculations without preferring this restriction in our study. Calculations were made to include all Ge isotopes in the data set used, and separate operations were not performed for the isotopes. That is, when applying ML methods, both the training and test datasets contain data for ^{70}Ge, ^{72}Ge, ^{73}Ge, ^{74}Ge and, ^{76}Ge isotopes.

The enormous amount of experimental data gathered over the years provides a logical opportunity for ML applications to inform cross sections in regions of uncertainty, iterate quicker through the assessment processes, and ultimately decrease, if not remove, human bias from the process. For this proof-of-concept work, three simple yet well-proven ML algorithms were used for the prediction of the total neutron-induced reaction (n, γ) cross-sections of ^{70}Ge, ^{72}Ge, ^{73}Ge, ^{74}Ge and, ^{76}Ge isotopes. There are two main components to the input to feed into our machine learning algorithms, which are the incident energy of the neutron (E), and the neutron number (N) of the isotopes. The output is the cross-section (XS) for the neutron-induced reaction. The properties related to the reactions on Ge isotopes are in given in Table 1.

To construct a reliable model to predict the neutron-induced reaction cross-sections of Ge isotopes, Pearson's correlation coefficient (r) is calculated for all possible combinations of our input descriptors (Benesty et al., 2009). The correlation between the variables is seen in Fig. 1. As expected, there is almost no correlation among descriptors (N and E) of the Ge isotopes (r = 0.014) which lowers the cost of training and reduces the possibility of an overfitting or under fitting problems (Dieterich, 1995).

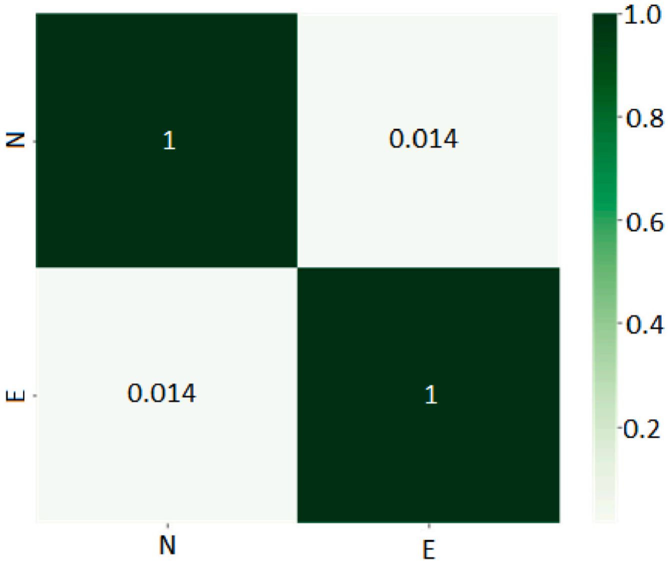

Fig. 1. Correlation between descriptors N and E used as inputs for the ML model.

Table 1
The properties of the isotopes and the related reactions (A, Z and N are the mass, proton and neutron numbers of the Ge isotopes).

Isotope			Number of Data	
A, Z, N	Abundance (%)	Reaction	Training	Test
70, 32, 38	20.52	^{70}Ge + n -> ^{71}Ge+γ	94	28
72, 32, 40	27.45	^{72}Ge + n -> ^{73}Ge+γ	85	34
73, 32, 41	7.76	^{73}Ge + n -> ^{74}Ge+γ	89	32
74, 32, 42	36.52	^{74}Ge + n -> ^{75}Ge+γ	88	32
76, 32, 44	7.75	^{76}Ge + n -> ^{77}Ge+γ	93	27

To avoid overfitting, cross-validation method was used in the current work, aiming to make optimal use of each sample and to obtain more justified models. Therefore, the datasets are divided using k-fold cross-validation. Then, it is partitioned into k randomly selected subsets of equal size, with one of the subsets utilized for testing. The technique is then continued until at least one of the k subsets is used as the testing dataset. In this paper, 5-fold cross-validation is used, in which the dataset is randomized and divided into 5 subsets, the machine is trained using 4 of the subsets (80%), after that, the testing part is done using the remaining subset (20%), these steps are repeated until each subset is used for testing once, and the root-mean square error for each cross-validation is estimated. The whole data (602 data points) belonging to the problem is divided into two randomly chosen subsets of unequal size set. One of them will be used for training the machine learning model using 75% of the dataset and the testing is done using the remaining (25%). The performance for each algorithm is analysed and compared using mean square error (MSE) (Chai and Draxler, 2014). In addition, the correlation coefficient (R^2) as a quality criteria were used to describe the proportion of variability in a dataset that can be explained by the model.

$$MSE = \frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{n} \tag{1}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{n} (y_i - < y >)^2} \tag{2}$$

where n is the number of observations, y_i is the actual value, $\hat{y_i}$ is the predicted value, and $<y>$ is the overall mean.

3. Model description and numerical algorithm

In this supervised regression ML problem, we apply three types of learning techniques, K-Nearest Neighbors (KNN), Support Vector Machines (SVM), and Artificial Neural Network (ANN). This Section contains a basic description of these algorithms. Python 3 with the Scikit-learn package was used for building all machine learning models (Pedregosa et al., 2011).

3.1. K-Nearest Neighbors (KNN)

The KNN method, a supervised learning algorithm, is a basic regression and classification method (Goldberger et al., n.d.; Tong et al., 2021). The KNN regression technique determines the value of a new data point by first searching the K nearest data points in the training dataset and classifying them by increasing distance, thus the model's name. K is a hyper-parameter that must be set before the training step. There are several distance metrics available, including Manhattan and Euclidean, that may be used to calculate the distance between points in the i dimensional space, where i is the number of trained variables.

In the Euclidean example (Eq. (3)), the distance between points x and y is computed by calculating the square root of the squared difference between $(x_1, x_2, \ldots, x_n)$ and $(y_1, y_2, \ldots, y_n)$. On the other hand, the Manhattan distance (Eq. (4)) is determined by taking the absolute distance between points x and p. The distance function of KNN algorithms take the form:

$$d(x,y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \tag{3}$$

$$d(x,y) = \sum_{i=1}^{n} |x_i - y_i| \tag{4}$$

The value of the new data point (Eq. (5)) is equal to the mean of the K nearest values if the weights are uniform. As an alternative, the K closest points can be weighted according to distance, which means that closest points have a greater impact on the value of the new data point (they are weighted by the inverse of their distance).

$$\text{newpoint} = \frac{1}{k} \sum_{i}^{k} k_i \tag{5}$$

The accuracy and performance of the model usually depend on the choice of hyper-parameters. Bayesian optimization of the hyper-parameters is performed, choosing the ones that minimize cross-validation error. These parameters need to be adjusted for the model to generate a good function that fits and explains the cross section data accurately although, these parameters can give too much flexibility consequently under fitting or overfitting the data. For KNN, Bayesian optimization is performed using cross validation to find the optimal value of the parameter K (number of nearest neighbors). Fig. 2 represents the RMSE of the cross-validation (CV) and test set as a function of the parameter K. Larger values of K result in models with higher bias and lower variance, whereas smaller values of K generate models with high variance and small bias. We can see that the best performance of our model is when the value of K is equal to 1.

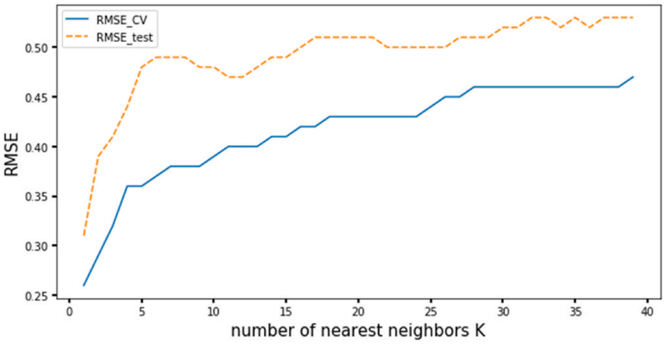

Fig. 2. Test and cross validation RMSE as a function of the number of neighbors (k).

3.2. Support Vector Machines (SVM)

Support Vector Machine (SVM), was first developed by Vapnik. It has become a hot topic of intensive study due to its successful application in classification tasks and regression tasks. SVM was first formulated as supervised learning that utilizes hyperplanes for the separation of classes (Chang and Lin, 2011). Essentially, SVM converts a non-linear regression into a linear regression by projecting input space to new feature space by using kernel functions. After employing hyperplanes to classify data points into clusters, the algorithms create a line or hyperplanes that decrease the error or loss function. The SVM approach was more recently extended to regression problems, a domain in which it was SVR. In SVM regression, a fixed (non-linear) mapping is used to first map the input x onto an N-dimensional feature space, and then a linear model is built in this feature space (Cherkassky and Ma, 2004; Dewi and Chen, 2019; Yang et al., 2002). The SVR gives prediction solutions based on the functional form equation:

$$f(x,w) = \sum_{n=1}^{N} w_n K(x, x_n) + w_0 \tag{6}$$

where w_n and x_n are respectively the model weight and the position of ach SVs. In addition n is the number of SVs, w_0 is the bias and K (x, x_n) is a kernel function corresponding to x_n.

Now the question is to determine w_n and w_0 from the training data by minimizing the empirical risk, $R_{emp}(w)$, based on the empirical risk,

$$R_{emp}(w) = \frac{1}{n} \sum_{i=1}^{n} L_\varepsilon(y_i, f(x_i, w)) \tag{7}$$

Absolute value error and squared error are two common loss functions used for minimizing empirical risk. SVM regression employs a novel loss function known as ε-insensitive loss.

SVM regression uses ε -insensitive loss to perform linear regression in the high-dimensional feature space while simultaneously attempting to minimize model complexity by minimizing $\|w\|^2$.This may be characterized by inserting (non-negative) slack variables ξ_i, ξ_i^* $i = 1, \ldots n,$ to assess the deviation of training samples beyond the 1-insensitive zone. As a result, SVM regression is defined as the minimizing of the following functional:

$$\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n} (\xi_i + \xi_i^*) \tag{8}$$

Where C is a positive constant (regularization parameter).

A single kernel function is employed in the conventional method, and its form is determined by a set of parameters. The quality of the regression, like other kernel-based techniques, depends on the selection of the kernel function and its parameters, which must be appropriate for the present data. It is well known that SVM generalization performance (estimation accuracy) depends on a good setting of hyper-parameters 'C', 'ε' and kernel function. In the current work, the hyper-parameters for SVM model were determined using the grid search function. Search range for C and ε was in between 10 to 100 and 1 to 0.01 respectively. Kernel function considered is linear or 'rbf'. Based on the grid search results, the value of C and ε had little effect on the model, and they were finally set to 100 and 0.1 respectively. Kernel function was fixed as 'rbf'. Changes in parameters other than ε and C had little effect on the model accuracy and were therefore left at the default values implemented in the Scikit-learn package.

3.3. Artificial Neural Network

As a nonlinear mathematical method, ANN is a powerful and versatile method that mimics the human brain functionality and consists of several processing units called neurons (Hornik et al., 1989; Otchere et al., 2021). It is a network composed of three types of layers: input, hidden, and output layers, each has a number of neurons linked to those in nearby levels (Fig. 3). The basic components of an ANN are the neurons which are organized in a different layered structure and connected via adaptive synaptic weights. The weights are calculated beforehand during a training phase. The first layer forwards the input vector to be identified to the first neurons. Input layer neurons receive the data from the environment and the output layer neurons give the result as close as to the desired ones. A neuron receives the outputs of the previous neurons as input, performing weighted addition of them, evaluating the result on an activation function, and forwarding the outcome to the next layer. Furthermore, there is no rule for the number of hidden layers and neurons between input and output layers. In the application of the ANN method, different numbers of hidden neurons have been considered. One of the structures chosen is composed of two hidden layers with ten hidden neurons along with the input and output layers (with two and one neurons, respectively) as it has also been enough to achieve high accuracy with the target dataset. In the training stage, the adaptive weights between neurons are adjusted to construct ANN. The weights play an important role in solving the problem. If weight is adjusted well, it works for all similar types of data that have never been seen in the training stage. Therefore, this training stage continues until the acceptable error level. Parameter optimization is focused on the activation function, and on early stopping of training to limit overfitting. In this application, Levenberg-Marquardt's back-propagation algorithm for the training and tangent hyperbolic activation function for the hidden neurons were used.

4. Results and discussion

Since cross-section values corresponding to different energies take numerical values in a very wide range, the data range needs to be narrowed. One of the methods used for this purpose is to use the logarithms of the values. In the results presented in this study, the cross section is meant by the logarithms of the total cross-section values. Therefore, in the ML calculations performed, the logarithms of the total cross-section values of the neutron induced reactions taken from the TENDL-2019 database were used. The datasets for ^{70}Ge, ^{72}Ge, ^{73}Ge, ^{74}Ge, and ^{76}Ge total cross-sections (Koning et al., 2019a, 2019b, 2019c, 2019d, 2019e) were fed into the algorithm of feed-forward multilayer ANN, SVM and KNN to build a machine learning model with a nuclear cross-section library as the target output.

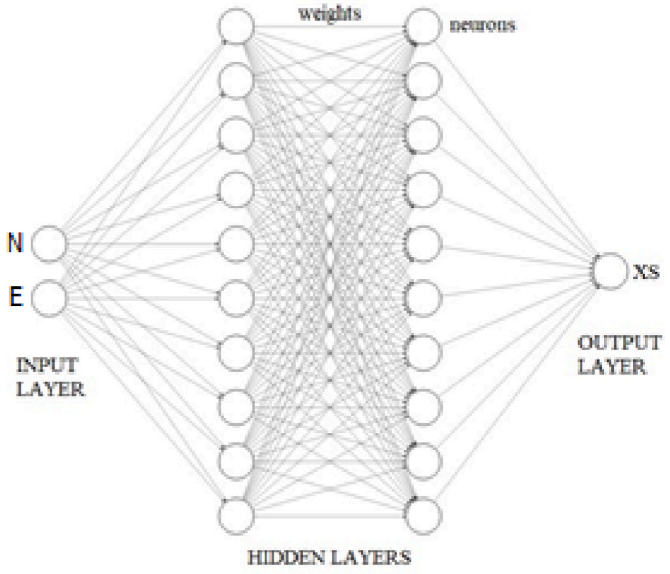

Fig. 3. One of the used ANN structure with two hidden layers of 10 hidden neurons in each.

In the first stage of our study, we estimated the neutron induced reaction cross sections on isotopes using ANNs of different structures. The performance of our ANN model for the ^{70}Ge, ^{72}Ge, ^{73}Ge ^{74}Ge and ^{76}Ge neutron induced total cross-sections are tabulated in Table 2 in terms of R^2 and MSE. Figs. 4 and 5 show the graph of the estimated versus actual values, from which the accuracy of the prediction made by the ANN algorithm can be seen.

As can be seen in Table 2, according to the results obtained by using a single hidden layer and 8 hidden neurons ANN structure, R^2 and MSE values were obtained as 0.96 and 0.14 mb^2, respectively. The increase in the number of neurons in the hidden layer to 20 led to a noticeable improvement in training. R^2 and MSE values were obtained as 0.97 and 0.10 mb^2 by using this structure. This noticeable decrease in MSE meant that increasing the number of neurons in the hidden layer would have positive effects on the results. However, increasing the number of hidden layers to 2 and the inclusion of 10 neurons in each layer did not create a significant improvement in the results of the training data. In Fig. 4, the results obtained from ANN structures with different hidden layers are presented graphically. In this figure, in which the ANN estimates are presented against the theoretical literature data, it is seen that the ANN predictions in the range where the cross section is about −2 to 2 mb are better. At larger values, the deviations are clearly evident. While the maximum deviation in the single hidden layer 8-neuron structure was 3.20 mb, the maximum deviation in the 20-neuron structure was 2.73 mb. Whereas, the maximum deviation in the double hidden layer structure is 2.74 mb.

The figure in which the estimates on the test data of the different ANN structures used are examined is presented in Fig. 5. Again, it is clearly seen that the estimations in the range of about −2 to 2 mb are more successful, but the deviations in the larger cross-section values are higher. It was observed that the maximum deviation of the structure with a single hidden layer and 8 neurons was 3.20 mb, and the deviation in the structure of 20 neurons was 2.76 mb. On the other hand, the maximum deviation of the double hidden layer and 10-neuron structure is 2.72. The increase in the number of neurons in a single-layer structure and over 20 increases the performance of the ANN, which is approaching memorization, on the training data, but it reduces the estimations on the test data below the acceptable limit. Therefore, we did not consider it necessary to present the results we obtained from more than 20 structures in this study. The different ANN structures, also presented in Table 1, are the results in which the results for the training and test data are acceptably balanced. This is the reason why we do not present the results of our ANN calculations in other structures, whether single layers or double layers.

After examining the ANN results using different structures, we also addressed the problem with SVM and KNN machine learning approaches. Figs. 6 and 7 depict the predicted versus actual plot where we can observe the accuracy of the prediction made by KNN and SVM algorithms, respectively. The performance of three proposed machine learning algorithms has been investigated using the correlation coefficient R^2 and MSE as reported in Table 3.

For the training stage, we can evaluate from Table 3 that the ANN algorithm have the highest correlation coefficient with R^2 = 0.97 and MSE = 0.09, whereas the KNN algorithm has the lowest error with MSE = 0.07 mb^2 for the testing stage compared to ANN and SVM algorithms.

Table 2
The performance of our ANN model using TENDL-2019 input. R^2 is the correlation coefficient while MSE is the mean square error.

Training Data			Test Data	
	R^2	**MSE**	**R^2**	**MSE**
ANN (h=8)	0.96	0.14	0.94	0.20
ANN (h=20)	0.97	0.10	0.95	0.17
ANN (h=10–10)	0.97	0.09	0.95	0.15

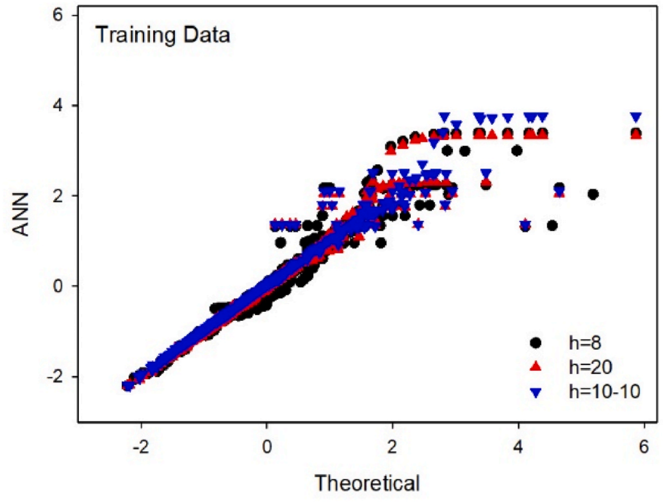

Fig. 4. Predictions of cross-section by using ANN with different hidden neuron structures on the training data set.

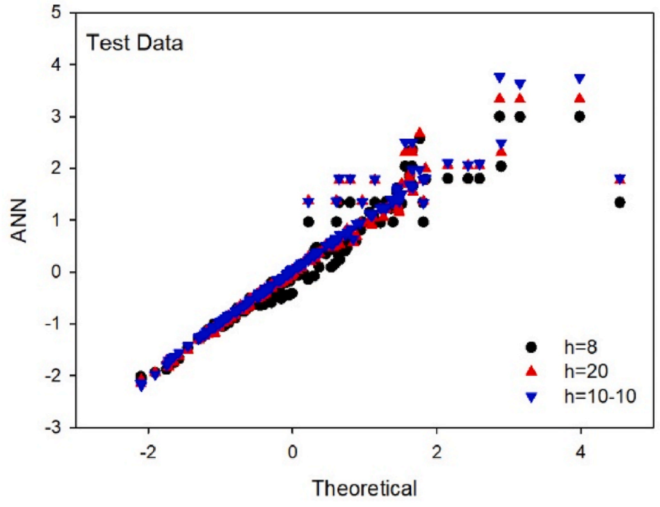

Fig. 5. Predictions of cross-section by using ANN with different hidden neuron structures on the test data set.

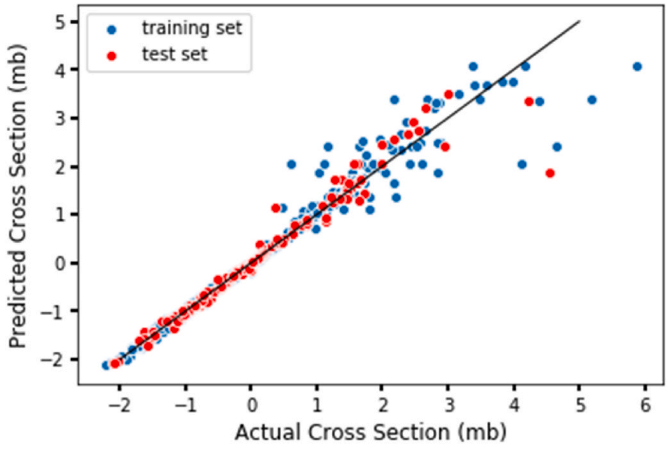

Fig. 6. Predictions of cross-section by using KNN on the test data set.

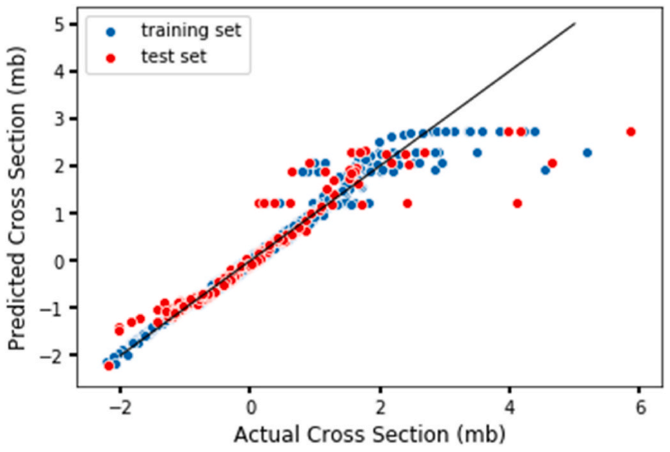

Fig. 7. Predictions of cross-section by using SVM on the test data set.

Table 3
Evaluated ML models predictions error on the TENDL-2019 library Measurements for Ge Reactions.

Training Data			Test Data	
	R^2	MSE	R^2	MSE
KNN	0.95	0.07	0.94	0.07
SVM	0.94	0.09	0.84	0.27
ANN (h=10–10)	0.97	0.09	0.95	0.15

From Figs. 5–7, we observed the comparison of the performances of our machine learning models. We can see that ANN and SVM algorithms exhibit the same trends were, the KNN algorithm performs better in term of MSE. For KNN, the predicted nuclear cross-section shows a good fit with TENDL-2019 library data points when being fed input sets.

In Figs. 8 and 9, energy values versus total reaction cross-section graphs of (n, γ) reactions performed with neutrons sent on five different Ge isotopes are given. The results of the double hidden layer (10-10) ANN structure, which is the machine learning with the highest performance in this study, are presented in comparison with the TENDL-2019 data. In Fig. 8, the ANN estimates on the training data set are given and it is seen that the ML results are in agreement with the literature. In Fig. 9, the results of the ANN predictions on the test data set are presented comparatively. It is seen that a few ANN estimates at very small values, where the energy value is close to zero, are lower than the literature on both training and test data. This may be due to the fact that the energy value is close to zero, so it is treated as zero in the input neuron in the ANN calculations. It is worth reminding again that the logarithms of the cross-section values were used in the calculations for the estimations.

5. Conclusion

In this study, we proposed an alternative approach to the existing theoretical models in generating nuclear reaction cross-section data using machine learning techniques. The predictions of total neutron-induced reactions on ^{70}Ge, ^{72}Ge, ^{73}Ge, ^{74}Ge, and ^{76}Ge isotopes with neutron energy from 0.5 to 200 MeV have been performed by using artificial intelligence methods. The data for the cross-section of the different reactions by different energy values is highly non-linear. Therefore, it is a very difficult task to predict cross-section values with great accuracy. We tried to reach our goal by using three different machine learning algorithms, which are ANN, SVM, and KNN. We have used ANN in different structures with single and fence layers. After many trials, we concluded that using two hidden layers with ten hidden neurons, and tangent hyperbolic activation function gives good results. We also examined the effects of increasing the number of layers on the

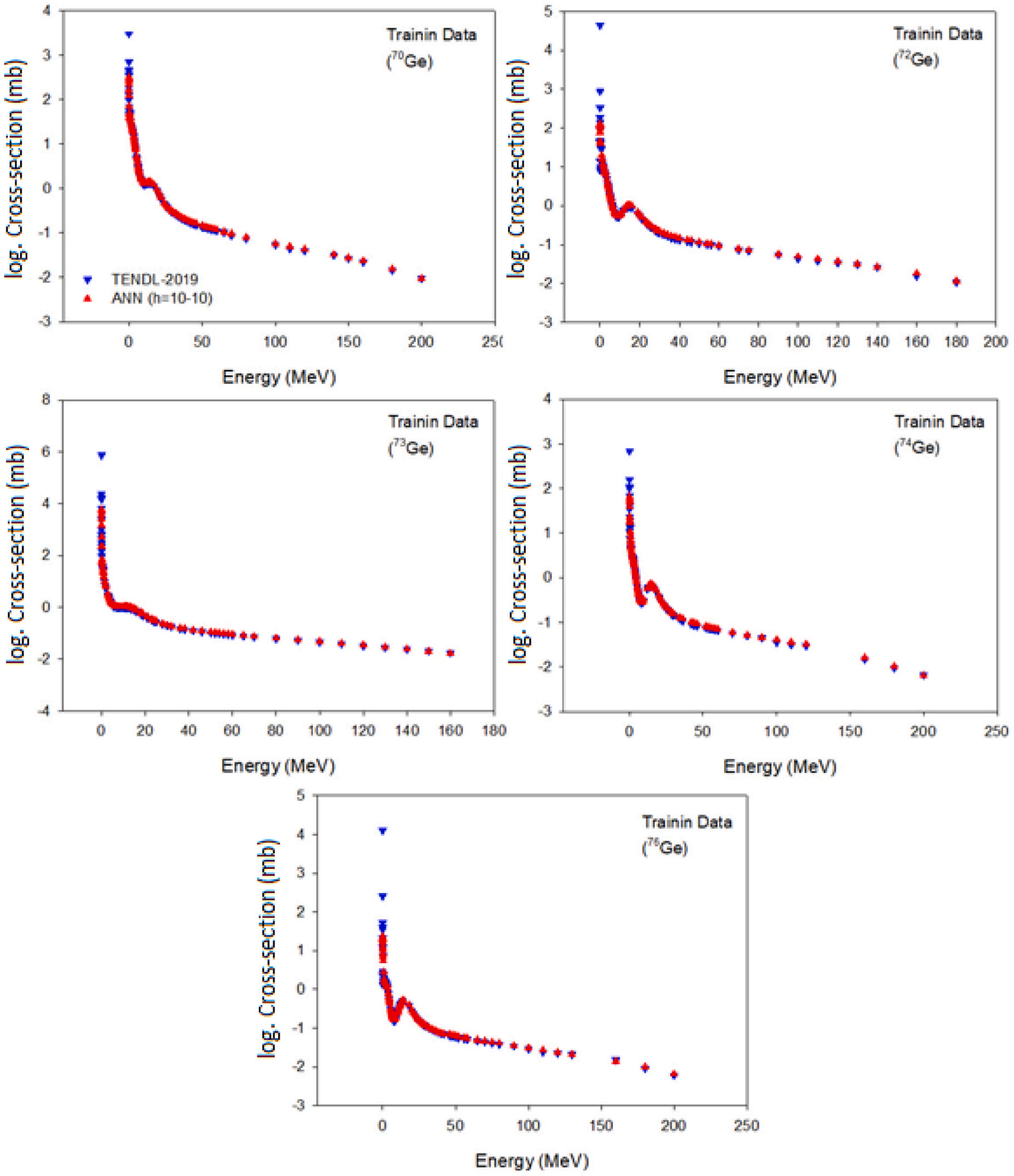

Fig. 8. Excitation function for neutron activation reaction from the ANN calculation on training data in comparison with TENDL-2019.

results and performed our calculations with structures with 2 hidden layers. Among them, we also presented the results of the bilayer structure with 10 neurons each. However, we cannot say that the use of double layers causes significant improvements in the results. According to the results, ANN and KNN are found to be better compared to SVM algorithm. Another contribution of this study to the literature from a different perspective is that it is a cross-validation method for calculations based on theoretical calculations. In addition, although it is concluded that the ML results are more compatible with the experimental data than the other theoretical results, the theories can be improved by updating them. Moreover, one can also apply these approaches to improve and predict other nuclear properties with many experimental data, such as β-decay half-lives, nuclear masses, nuclear charge radii, and so on.

Authorship statement

Conception and design of study: Serkan Akkoyun, Rihab Gargouri, Ramzi Maalej, Kamel Damak acquisition of data: analysis and/or interpretation of data: Rihab Gargouri, Serkan Akkoyun, Ramzi Maalej, Kamel Damak.

Drafting the manuscript: Rihab Gargouri, Serkan Akkoyun, Ramzi Maalej, Kamel Damak revising the manuscript critically for important intellectual content: Serkan Akkoyun, Rihab.

Gargouri, Ramzi Maalej, Kamel Damak.

Approval of the version of the manuscript to be published (the names of all authors must belisted): Rihab Gargouri, Serkan Akkoyun, Ramzi Maalej, Kamel Damak.

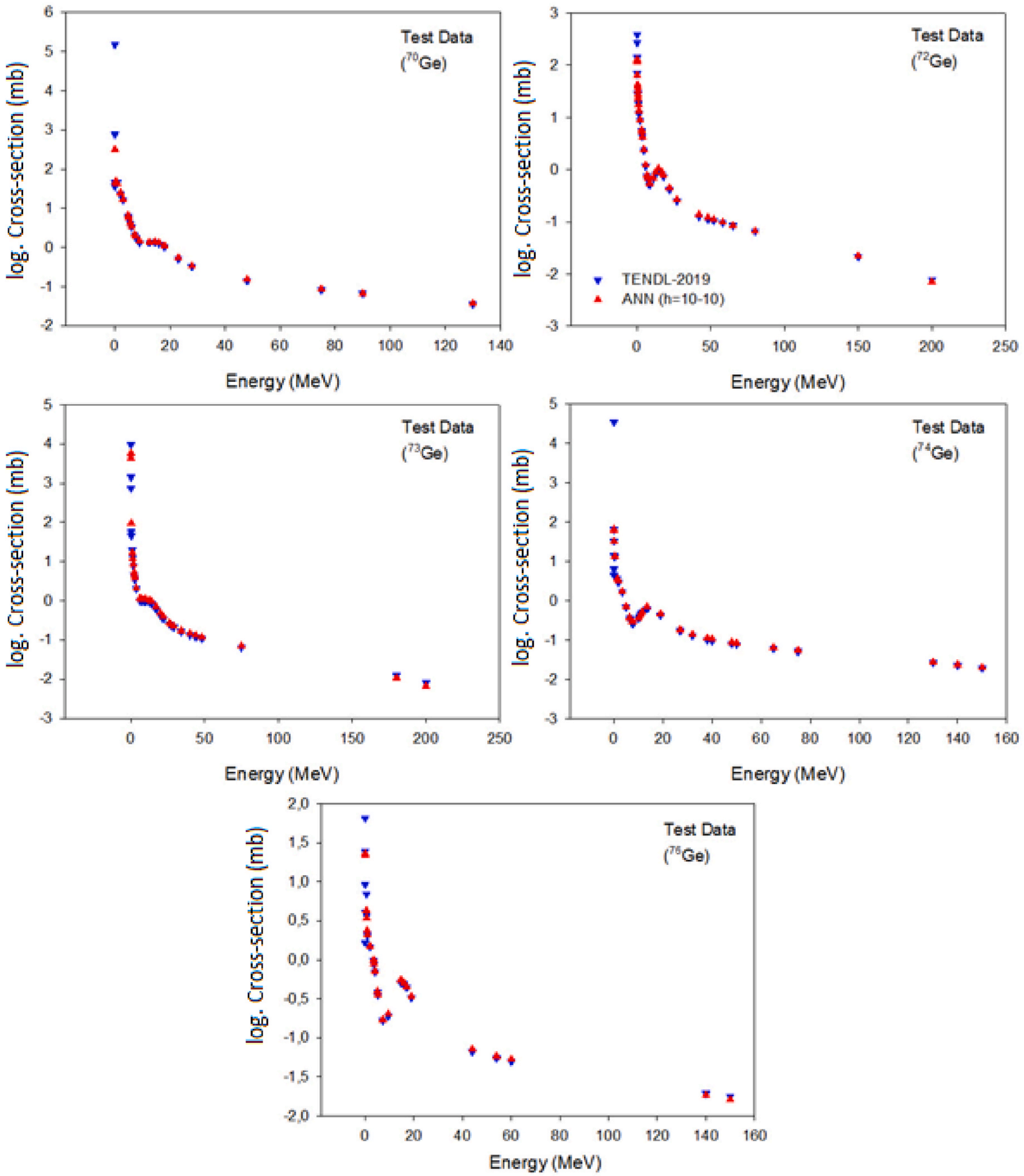

Fig. 9. Excitation function for neutron activation reaction from the ANN calculation on test data in comparison with TENDL-2019.

Declaration of competing interest

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

Data availability

Data will be made available on request.

Acknowledgements

This work is supported by the Scientific Research Project Fund of Sivas Cumhuriyet University under the project number F-2022-667.

References

Benesty, J., Chen, J., Huang, Y., Cohen, I., 2009. Pearson correlation coefficient. In: Noise Reduction in Speech Processing, Springer Topics in Signal Processing. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 1–4. https://doi.org/10.1007/978-3-642-00296-0_5.

Chadwick, M.B., n.d. ENDF/B-VII.1 Nuclear Data for Science and Technology: Cross Sections, Covariances, Fission Product Yields and Decay Data. NUCLEAR DATA SHEETS vol. 110.

Chai, T., Draxler, R.R., 2014. Root mean square error (RMSE) or mean absolute error (MAE)? – Arguments against avoiding RMSE in the literature. Geosci. Model Dev. (GMD) 7, 1247–1250. https://doi.org/10.5194/gmd-7-1247-2014.

Chang, C.-C., Lin, C.-J., 2011. LIBSVM: a library for support vector machines. ACM Trans. Intell. Syst. Technol. 2, 1–27. https://doi.org/10.1145/1961189.1961199.

Cherkassky, V., Ma, Y., 2004. Practical selection of SVM parameters and noise estimation for SVM regression. Neural Network. 17, 113–126. https://doi.org/10.1016/S0893-6080(03)00169-2.

Dewi, C., Chen, R.-C., 2019. Random forest and support vector machine on features selection for regression analysis. https://doi.org/10.24507/ijicic.15.06.2027.

Dietterich, T., 1995. Overfitting and undercomputing in machine learning. ACM Comput. Surv. 27, 326–327. https://doi.org/10.1145/212094.212114.

Goldberger, J., Hinton, G.E., Roweis, S.T., Salakhutdinov, R.R., n.d. Neighbourhood Components Analysis 8.

Hamid, M.A.B., Beh, H.G., Oluwatobi, Y.A., Chew, X.Y., Ayub, S., 2021. Neutron-induced nuclear cross-sections study for plasma facing materials via machine learning: molybdenum isotopes. Appl. Sci. 11, 7359. https://doi.org/10.3390/app11167359.

Haque, M.M., Islam, M.T., Hafiz, M.A., Miah, R.U., Uddin, M.S., 2009. 14.8 MeV neutron activation cross section Measurements for Ge isotopes. J. Sci. Res. 1, 173–181. https://doi.org/10.3329/jsr.v1i2.1532.

Hornik, K., Stinchcombe, M., White, H., 1989. Multilayer feedforward networks are universal approximators. Neural Network. 2, 359–366. https://doi.org/10.1016/0893-6080(89)90020-8.

Iwamoto, O., Herman, M., Mughabghab, S.F., Obložinský, P., Trkov, A., 2005. Neutron cross-section evaluations for 70, 72, 73, 74, 76Ge. In: In AIP Conference Proceedings (Vol. 769, No. 1, pp. 434-437). American Institute of Physics.

Koning, A.J., Rochman, D., Sublet, J.-Ch, Dzysiuk, N., Fleming, M., van der Marck, S., 2019. TENDL: complete nuclear data library for innovative nuclear science and technology. Nucl. Data Sheets 155, 1–55. https://doi.org/10.1016/j.nds.2019.01.002.

Koning, A.J., Rochman, D., Sublet, J.-Ch, Dzysiuk, N., Fleming, M., van der Marck, S., 2019a. https://tendl.web.psi.ch/tendl_2019/neutron_file/Ge/Ge070/tables/xs/ng.tot.

Koning, A.J., Rochman, D., Sublet, J.-Ch, Dzysiuk, N., Fleming, M., van der Marck, S., 2019b. https://tendl.web.psi.ch/tendl_2019/neutron_file/Ge/Ge072/tables/xs/ng.tot.

Koning, A.J., Rochman, D., Sublet, J.-Ch, Dzysiuk, N., Fleming, M., van der Marck, S., 2019c. https://tendl.web.psi.ch/tendl_2019/neutron_file/Ge/Ge073/tables/xs/ng.tot.

Koning, A.J., Rochman, D., Sublet, J.-Ch, Dzysiuk, N., Fleming, M., van der Marck, S., 2019d. https://tendl.web.psi.ch/tendl_2019/neutron_file/Ge/Ge074/tables/xs/ng.tot.

Koning, A.J., Rochman, D., Sublet, J.-Ch, Dzysiuk, N., Fleming, M., van der Marck, S., 2019e. https://tendl.web.psi.ch/tendl_2019/neutron_file/Ge/Ge076/tables/xs/ng.tot.

Lederer-Woods, C., Battino, U., Ferreira, P., Gawlik, A., Guerrero, C., Gunsing, F., Heinitz, S., Lerendegui-Marco, J., Mengoni, A., Reifarth, R., Tattersall, A., Valenta, S., Weiss, C., Aberle, O., Andrzejewski, J., Audouin, L., Bécares, V., Bacak, M., Balibrea, J., Barbagallo, M., Barros, S., Bečvář, F., Beinrucker, C., Belloni, F., Berthoumieux, E., Billowes, J., Bosnar, D., Brugger, M., Caamaño, M., Calviño, F., Calviani, M., Cano-Ott, D., Cerutti, F., Chiaveri, E., Colonna, N., Cortés, G., Cortés-Giraldo, M.A., Cosentino, L., Damone, L.A., Deo, K., Diakaki, M., Dietz, M., Domingo-Pardo, C., Dressler, R., Dupont, E., Durán, I., Fernández-Domínguez, B., Ferrari, A., Finocchiaro, P., Frost, R.J.W., Furman, V., Göbel, K., García, A.R., Gheorghe, I., Glodariu, T., Gonçalves, I.F., González-Romero, E., Goverdovski, A., Griesmayer, E., Harada, H., Heftrich, T., Hernández-Prieto, A., Heyse, J., Jenkins, D.G., Jericha, E., Käppeler, F., Kadi, Y., Katabuchi, T., Kavrigin, P., Ketlerov, V., Khryachkov, V., Kimura, A., Kivel, N., Knapova, I., Kokkoris, M., Krtička, M., Leal-Cidoncha, E., Leeb, H., Licata, M., Lo Meo, S., Losito, R., Macina, D., Marganiec, J., Martínez, T., Massimi, C., Mastinu, P., Mastromarco, M., Matteucci, F., Mendoza, E., Milazzo, P.M., Mingrone, F., Mirea, M., Montesano, S., Musumarra, A., Nolte, R., Palomo-Pinto, F.R., Paradela, C., Patronis, N., Pavlik, A., Perkowski, J., Porras, J.I., Praena, J., Quesada, J.M., Rauscher, T., Riego-Perez, A., Robles, M., Rubbia, C., Ryan, J.A., Sabaté-Gilarte, M., Saxena, A., Schillebeeckx, P., Schmidt, S., Schumann, D., Sedyshev, P., Smith, A.G., Stamatopoulos, A., Suryanarayana, S.V., Tagliente, G., Tain, J.L., Tarifeño-Saldivia, A., Tassan-Got, L., Tsinganis, A., Vannini, G., Variale, V., Vaz, P., Ventura, A., Vlachoudis, V., Vlastou, R., Wallner, A., Warren, S., Weigand, M., Wright, T., Žugec, P., 2019. Measurement of 73Ge(n,γ) cross sections and implications for stellar nucleosynthesis. Phys. Lett. B 790, 458–465. https://doi.org/10.1016/j.physletb.2019.01.045.

Meierhofer, G., Grabmayr, P., Jochum, J., Kudejova, P., Canella, L., Jolie, J., 2010. Thermal neutron capture cross section of Ge 74. Phys. Rev. C 81, 027603. https://doi.org/10.1103/PhysRevC.81.027603.

Meierhofer, G., Kudejova, P., Canella, L., Grabmayr, P., Jochum, J., Jolie, J., n.d. Thermal neutron capture cross-section of 76Ge. The Eur. Phys. J. A 4.

Niu, Z.M., Liang, H.Z., 2018. Nuclear mass predictions based on Bayesian neural network approach with pairing and shell effects. Phys. Lett. B 778, 48–53. https://doi.org/10.1016/j.physletb.2018.01.002.

Otchere, D.A., Arbi Ganat, T.O., Gholami, R., Ridha, S., 2021. Application of supervised machine learning paradigms in the prediction of petroleum reservoir properties: comparative analysis of ANN and SVM models. J. Petrol. Sci. Eng. 200, 108182 https://doi.org/10.1016/j.petrol.2020.108182.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Duchesnay, E., 2011. Scikit-learn: Machine learning in Python. J. Mach. Learn. Res. 12, 2825–2830.

Plompen, A.J.M., 2020. The joint evaluated fission and fusion nuclear data library, JEFF-3.3. Eur. Phys. J. A 108.

Shibata, K., Iwamoto, O., Nakagawa, T., Iwamoto, N., Ichihara, A., Kunieda, S., Chiba, S., Furutaka, K., Otuka, N., Ohsawa, T., Murata, T., Matsunobu, H., Zukeran, A., Kamada, S., Katakura, J., 2011. JENDL-4.0: a new library for nuclear science and engineering. J. Nucl. Sci. Technol. 48, 1–30. https://doi.org/10.1080/18811248.2011.9711675.

Tárkányi, F., Hermanne, A., Ditrói, F., Takács, S., Ignatyuk, A.V., Spahn, I., Spellerberg, S., 2021. Activation cross section data of deuteron induced nuclear reactions on rubidium up to 50 MeV. Eur. Phys. J. A 57, 21. https://doi.org/10.1140/epja/s10050-020-00327-1.

Tárkányi, F., Takács, S., Ditrói, F., Hermanne, A., Adam-Rebeles, R., Ignatyuk, A.V., 2020. Investigation of the deuteron induced nuclear reaction cross sections on lutetium up to 50 MeV: review of production routes for 177Lu, 175Hf and 172Hf via charged particle activation. J. Radioanal. Nucl. Chem. 324, 1405–1421. https://doi.org/10.1007/s10967-020-07182-w.

Tong, L., He, R., Yan, S., 2021. Prediction of neutron-induced fission product yields by a straightforward k -nearest-neighbor algorithm. Phys. Rev. C 104, 064617. https://doi.org/10.1103/PhysRevC.104.064617.

Vicente-Valdez, P., Bernstein, L., Fratoni, M., 2021. Nuclear data evaluation augmented by machine learning. Ann. Nucl. Energy 163, 108596. https://doi.org/10.1016/j.anucene.2021.108596.

Yang, H., Chan, L., King, I., 2002. Support vector machine regression for volatile stock market prediction. In: Yin, H., Allinson, N., Freeman, R., Keane, J., Hubbard, S. (Eds.), Intelligent Data Engineering and Automated Learning — IDEAL 2002, Lecture Notes in Computer Science. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 391–396. https://doi.org/10.1007/3-540-45675-9_58.