

USING DATA ANALYSIS TECHNIQUES TO CLASSIFY COUNTIES FROM ROMANIA IN TERMS OF EDUCATION, ACCORDING TO INDEXES FROM ECONOMIC AND SOCIAL FIELDS

*Alexandra-Andreea Mironescu
Grigorescu Ioana Gabriela
Iordache Ana Maria Mihaela**

Abstract

The level of development of a community is closely related to the level of development of the existing education system in that region. In this article we performed an analysis of the education system in Romania from economic and social points of view. We chose indicators that show movement, birth rate, integration into a form of education and population gains, but also the degree of use of modern computing techniques in the education process. The classification of the counties was performed with the help of a neural network trained in the SAS and several scenarios were developed with immediate applicability.

Keywords: *educational, economic development, efficiency, school choice, backpropagation, data analysis*

JEL Classification: C19, C45, C89, I25, I21, C87, I21

1. Introduction

The education system is an important factor that underlies the good development of any community, whether we are discussing at the microeconomic level like regions, county, and others or at the macroeconomic level like country, macro-region or continent. In general, the education system consists of all units and educational institutions with state or private capital, of various types, levels, and forms of organization of training and education.

For a more efficient organization, an education system is structured on educational levels, this ensuring a coherence of the training of the school age population, according to the age and individual characteristics of the pupils and students and in close connection with the requirements of contemporary society. [1]

Alexandra-Andreea Mironescu is PhD Student in Management, The Bucharest University of Economic Studies, Bucharest, E-mail address: alexandramironescu84@yahoo.ro

Grigorescu Ioana Gabriela is Assistant Professor PhD at the Romanian-American University, Bucharest. E-mail address: ioana.gabriela.grigorescu@rau.ro

* Iordache Ana Maria Mihaela is Associate Professor PhD at the Romanian-American University, Bucharest. E-mail address: mihaela.iordache@rau.ro

The education system, both in Romania and in any country in the world can be seen as a cybernetic system which has inputs, outputs and processes or feedback functions that ensure its balance. [2] From the point of view of the inputs, we can identify the social and economic exigencies of the contemporary society, the available financial resources, but also the human resources represented mainly by the teachers. The exits from this system are the future specialists from various fields, specialists who through their accumulated level of training are able to meet the requirements of the society at that moment. Feedback processes or functions are represented by policies approached at the macroeconomic level. Each element of these policies has a greater or lesser influence, in direct or indirect manner, on the education system. Being a system in which time plays an important role, policies at the level of a country must be adopted in the medium and long term so as to create as few shocks as possible, and return to equilibrium to occur as soon as possible, without having significant repercussions in the future.

Thus, the current education system in Romania is characterized by openness and dynamism to the new, operating in accordance with economic and social requirements, scientific and cultural progress, the aspirations of the Romanian people in the conditions of civil society and the rule of law, democratic. The structure of the education system in Romania includes: preschool education, primary education, lower secondary education, compulsory general education, upper secondary education, arts and crafts schools, apprenticeship schools, post-secondary education and higher education.

In the education system, any disturbance on the input can lead to the destabilization of the entire system and the point of balance to be reached again by adopting the correct policies. But in order to reach the adoption of those measures, studies are carried out in various aspects in order to identify possible vulnerabilities and to propose solutions in time to minimize the effect of noise.

The theory confirmed that a system can be analyzed in relation to other systems, components of a whole, but every component can also be analyzed as a system too. In the literature, the education system has been studied in connection with possible factors that can destabilize it. Thus, a study was analyzed the education systems in various countries and proposed solutions to avoid labor migration. [3] Also, it was studied the impact of the education system on the development of the national economy, the solutions found related to the decentralization of education, the intensification of adult education and the establishment of public-private partnerships. [4] Reference [5] showed how artificial intelligence influences the next generation, both teachers and students in their work. Artificial intelligence and especially neural networks have been successfully applied in other studies too [6-11]. Reference [12] presents an algorithm which trains a neural network that allows the choice of courses by university students in the e-learning system.

2. Methodology and datasets used in analysis

The problem studied in this article is the analysis of the education system in Romania, at the level of each county, from an economic-social point of view. Thus, we chose indicators that show movement, birth rate, integration into a form of education and population gains, but also the degree of use of modern computing techniques in the

education process, in order to have a more accurate analysis of existing reality. Also, out of the desire to evolve for the betterment of any larger or smaller community, several scenarios were developed with immediate applicability, but the effects were visible in the medium term, being known that any change in the education system has no immediate effects.

The methodology used in the application has the next steps:

- transforming the primary data into percentages in order to be able to compare more easily the values of the indicators with each other, for the same county. This is usually used due to the large intervals in which the initial data are contained.
- using the SAS Enterprise Guide software to classify the counties in Romania into distinct classes
- using the SAS Enterprise Miner program to train a multilayer perceptron neural network. The input data set for the neural network consists of both the initial data transformed into percentages and the belonging of each county in a distinct class.
- based on the trained neural network, we presented some scenarios regarding the future evolution of the educational system in Romania.

The primary data that formed the basis of the application were registered from the website of the National Institute of Statistics (NIS), for the year 2019 [13]. The chosen indicators were selected so as to best reflect both the education system in each county and the economic system too. Thus, social aspects were taken into account, such as the migration of able-bodied persons, the number of children enrolled in a form of education and unemployment. From economic point of view we took into account the indicators such as the number of unemployed people in each county and related to the IT&C field we chose the total number of computers intended for education.

The indicators used in the application are: the total school units, the number of live births, the average monthly net nominal earnings, the number of computers used for education, the number of permanent emigrants, the number of temporary emigrants (at least 12 months), the number of children enrolled in the pre-university education system and the total number of unemployed. In the application, each indicator has been assigned a code in order to work more easily. Table 1 shows the coding of the indicators.

Table 1. Initial values for the data sample for 2019

Crt. No.	Code of indicator	Full name of the indicator
1	I_1	Total school units
2	I_2	Number of children born alive in 2019
3	I_3	Average monthly net nominal earnings
4	I_4	Total number of computers intended for education
5	I_5	Definitive number of emigrants
6	I_6	Number of temporary emigrants (at least 12 months)
7	I_7	Number of children enrolled in the pre-university system
8	I_8	Total number of unemployed people

The indicator I_1, the total school units, counts the totality of the administrative education units with legal personality, registered in the National Nomenclature SIRUES, in which one or more educational levels function, having a unique management. Thus, the following units are included in this indicator: nurseries, kindergartens, primary and secondary schools, special primary and secondary schools, high schools, school groups within which several levels of education operate, arts and crafts schools (professional), foremen, post-secondary, university education institutions.

The indicator I_2, the number of children born alive in 2019, represents the social basin in which the other studied indicators have their roots, directly or indirectly. Thus, the school units numbers depend mainly on the number of students, the number of students depends in time on the number of children born alive.

The monthly net nominal earnings represent the amount of money remaining after deducting taxes and duties, such as: tax, social security contribution due by employees and social health insurance contribution due by employees. Indicator I_3, the average monthly net nominal earnings, represents the ratio between the net amounts paid to employees by economic agents in the reference month, regardless of the period for which they are due, and the average number of employees.

Indicator I_4, the total number of computers intended for education, counts the total number of electronic computing devices of the PC or laptop type included in the educational process. This indicator is strongly correlated with I_1, but also to a lesser extent with the other indicators used in the analysis.

Indicator I_5, the number of definitive emigrants, refers to persons who emigrate abroad and who give up their domicile in Romania, thus establishing their domicile on the territory of another state. Temporary emigrants (see I_6 indicator from the table 1) are people who no longer have their habitual residence in Romania for a period of at least 12 months. The habitual residence is represented by the place where a person spends his daily rest period, excluding temporary absences such as visits for various purposes such like holidays, delegations, medical treatments, religious pilgrimages, and others. Indicator I_7, the number of children enrolled in the pre-university system has in its component the whole number of the children from kindergartens, nurseries and students included in the pre-university training and education system from a school year, regardless of the form of education attend, the program studies and age.

The indicator of the total number of unemployed people includes the number of people who are looking for a job from the age of 16 until retirement, the number of people who do not earn income or earn less than the value of the reference social indicator of unemployment insurance and employment stimulation in force, graduates of educational institutions and graduates of special schools for people with disabilities, aged at least 16 years, who, within 60 days of graduation, failed to be employed according to professional training and others.

2. Results and discussions

After we chose the indicators and expressed them in percentages in order to be able to compare the counties with each other, we will further perform a first classification. For starters, we will choose the SAS Enterprise Guide program to obtain classes that are as homogeneous as possible on the inside, but as different as possible on the outside. Furthermore, using the results from this step as input data for the next

point in the proposed methodology, we will apply specific algorithms, implemented in the SAS Enterprise Miner software which will train a neural network. The purposes of training the neural network are trying to improve the initial classification and the reusability of the already trained network to another study. So, once trained the network, it can be used at a later date, without going through the whole algorithm presented in this research paper.

The initial data were recorded from NIS and were transformed into percentages (see table 2), so that the indicators can be comparable with each other and for a better reflection of reality.

Table 2. Initial values for indicators

County	I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8
Alba	2.01	1.57	2.39	1.65	1.4	1.71	1.65	2.02
Arad	2.23	2.14	2.34	1.83	2.08	2.16	2.06	1.01
Arges	3.06	2.81	2.6	2.58	1.63	2.98	3.08	3.01
Bacau	2.69	3.07	2.33	2.58	4.38	3.06	3.15	4.27
Bihor	2.91	3.06	2.14	2.93	1.46	2.86	3.03	1.43
Bistrita-Nasaud	1.59	1.72	2.15	1.17	1.38	1.44	1.58	1.29
Botosani	1.91	1.99	2.24	1.38	1.77	2.03	2.16	1.38
Brasov	2.66	3.21	2.17	3.24	1.5	2.67	2.94	2.03
Braila	1.51	1.24	2.61	1.02	3.85	1.56	1.44	1.68
Buzau	2.11	1.87	2.19	1.73	1.35	2.28	2.12	4.38
Caras-Severin	1.74	1.09	2.27	0.98	2.36	1.53	1.27	1.3
Calarasi	1.36	1.34	2.2	0.98	1.12	1.59	1.35	1.26
Constanta	3.26	3.41	2.37	3.26	3.6	3.51	3.71	2.9
Covasna	1.14	1.11	2.13	0.82	0.45	1.08	1.1	1.11
Dambovita	2.23	2.28	2.25	1.9	1.8	2.61	2.31	2.57
Dolj	3.09	2.97	2.48	2.95	1.94	3.38	3.12	6.96
Galati	2.67	2.46	2.33	2.5	4.26	2.65	2.59	4.1
Giurgiu	1.24	1.17	2.39	0.8	0.62	1.46	1.13	0.64
Gorj	1.66	1.36	2.38	1.3	0.76	1.71	1.68	1.84
Harghita	1.84	1.67	2.1	1.52	0.39	1.62	1.69	2
Hunedoara	1.59	1.63	2.17	1.72	2.13	1.99	1.83	2
Ialomita	1.29	1.23	2.18	0.92	0.74	1.44	1.24	1.64
Ifov	1.89	2.47	2.83	1.05	1.19	1.99	1.77	0.3
Maramures	2.67	2.35	2.18	2.1	1.67	2.42	2.4	2.01
Mehedinti	1.53	1.09	2.22	0.89	0.8	1.34	1.22	2.69
Mures	2.69	2.88	2.45	2.76	1.69	2.82	2.85	2.47
Neamt	1.97	2.27	2.14	1.91	2.88	2.44	2.35	2.85
Olt	2.3	1.68	2.4	1.24	1.08	2.23	1.98	3.43
Prahova	3.31	3.15	2.52	2.81	2.49	3.81	3.54	2.76
Satu Mare	1.87	1.73	2.28	1.3	1.86	1.77	1.21	1.41
Salaj	1.59	1.31	2.25	1.02	0.57	1.11	1.77	1.68
Sibiu	2.56	2.25	2.66	3.12	2.72	1.94	2.17	1.34
Suceava	3.19	4.13	2.19	3.46	2.58	3.16	3.83	4.28
Teleorman	1.99	1.31	2.11	1.01	1.55	1.96	1.41	3.42
Timis	3.44	3.79	2.96	5.18	5.33	3.42	3.31	1.08

Tulcea	1.16	0.83	2.32	0.88	1.17	1.1	0.99	1.13
Vaslui	2.11	2.21	2.22	1.44	3.4	1.98	2.12	3.89
Valcea	1.84	1.41	2.16	1.22	1.02	1.83	1.71	1.77
Vrancea	1.71	1.53	2.14	1.19	1.43	1.76	1.66	1.92
Cluj	3.89	3.77	3.09	9.59	3.11	3.4	3.41	1.74
Iasi	3.73	4.97	2.8	5.08	7.19	3.8	4.41	3.12
Bucuresti	8.8	10.49	3.64	12.96	15.28	8.36	9.67	5.91

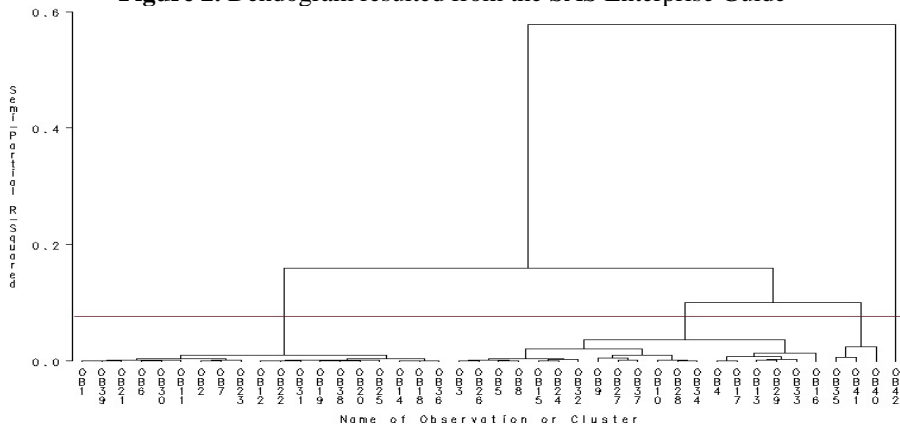
A first problem of the values of the indicators registered in table 2 is the identification of those of them that do not bring additional information in the model. Thus, after calculating eigenvalues it can be seen in Figure 1 that after the first seven indicators the amount of information has a maximum input. However, even if the amount of information brought in addition by indicator 8 is very small, we will keep it in the analysis for better reflection of reality.

Figure 1. Eigenvalues of the covariance matrix

Eigenvalues of the Covariance Matrix				
	Eigenvalue	Difference	Proportion	Cumulative
1	21.0956972	19.4663176	0.8839	0.8839
2	1.6293795	0.7544661	0.0683	0.9521
3	0.8749134	0.7409718	0.0367	0.9888
4	0.1339416	0.0754420	0.0056	0.9944
5	0.0584996	0.0269391	0.0025	0.9968
6	0.0315605	0.0063008	0.0013	0.9982
7	0.0252597	0.0067386	0.0011	0.9992
8	0.0185210		0.0008	1.0000

Following the application of the data clustering techniques in the SAS Enterprise Guide program, the dendrogram described in Figure 2 resulted. In the SAS Enterprise Guide program, we chose the Ward method as the county classification method. This involves minimizing the sum of the squares of the errors from an object to the center of its cluster. [14], [15] In other words, the Ward method divides the data set into classes that are as homogeneous as possible inside and as far apart as possible. [16], [17].

Figure 2. Dendrogram resulted from the SAS Enterprise Guide



From the dendrogram presented in Figure 2 results the classification method of the counties, coded as Ob1, Ob2 and so on until Ob42. We chose the distance between the clusters to be closer to level 0 than to 0.2, so that we have three classes for dividing the counties. If we had chosen as the cut-off point above, then two fairly similar classes would have resulted, meaning that the distance between them would have been smaller than in the first case. [18] From the point of view of the socio-economic reality, it was not true. The reasoning is similar for choosing a single class. Even if the two classes were very different, having a very large distance between them, the economic reality would not have allowed such a thing. From the graph presented in figure 2, observation 42, for example Bucharest is an outlier and will be eliminated further from future analyzes.

The resulting classes are as follows:

- first class (c1) formed by the counties: Alba, Arad, Bacau, Bistrita-Nasaud, Botosani, Buzau, Caras-Severin, Calarasi, Giurgiu, Gorj, Harghita, Hunedoara, Ialomita, Ilfov, Mehedinti, Salaj, Tulcea, Valcea, Vrancea;
- second class (c2) formed by the counties: Arges, Bihor, Brasov, Braila, Constanta, Covasna, Dambovita, Dolj, Galati, Maramures, Mures, Neamt, Olt, Prahova, Satu Mare, Sibiu, Suceava, Teleorman, Vaslui;
- third class (c3) formed by the counties: Timis, Cluj, Iasi.

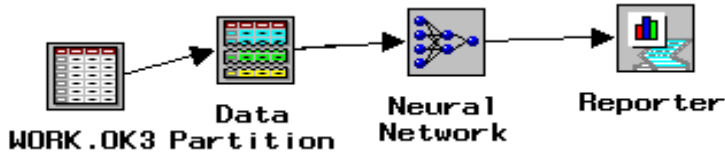
Furthermore, a neural network of the multilayer perceptron type was associated. [19-22] The matrix with the initial data for the neural network consisted of the data that formed the basis of the classification of the counties (table 2), and in addition we added three columns with the class where each county is (table 3). Thus, if a county is for example in the first class, in the column corresponding to c1 it will be written the number 1, and the other columns, c2 and c3 will contain the number 0. As we wrote above, from the data matrix initials for the neural network will be eliminated outliers, in our case the outlier being Bucharest.

Table 3. Matrix with initial data for the neural network

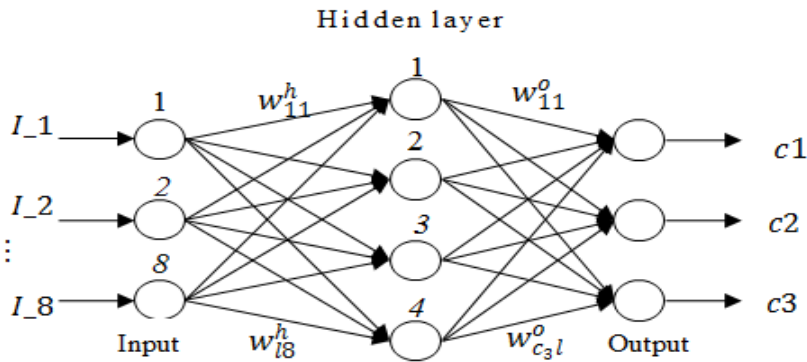
County	I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8	class_1	class_2	class_3
Alba	2.01	1.57	2.39	1.65	1.4	1.71	1.65	2.02	1	0	0
Arad	2.23	2.14	2.34	1.83	2.08	2.16	2.06	1.01	1	0	0
Arges	3.06	2.81	2.6	2.58	1.63	2.98	3.08	3.01	0	1	0
Bacau	2.69	3.07	2.33	2.58	4.38	3.06	3.15	4.27	1	0	0
Bihor	2.91	3.06	2.14	2.93	1.46	2.86	3.03	1.43	0	1	0
Bistrita-Nasaud	1.59	1.72	2.15	1.17	1.38	1.44	1.58	1.29	1	0	0
Botosani	1.91	1.99	2.24	1.38	1.77	2.03	2.16	1.38	1	0	0
Cluj	3.89	3.77	3.09	9.59	3.11	3.4	3.41	1.74	0	0	1
Brasov	2.66	3.21	2.17	3.24	1.5	2.67	2.94	2.03	0	1	0
Timis	3.44	3.79	2.96	5.18	5.33	3.42	3.31	1.08	0	0	1
Braila	1.51	1.24	2.61	1.02	3.85	1.56	1.44	1.68	0	1	0

Buzau	2.11	1.87	2.19	1.73	1.35	2.28	2.12	4.38	1	0	0
Suceava	3.19	4.13	2.19	3.46	2.58	3.16	3.83	4.28	0	1	0
Caras-Severin	1.74	1.09	2.27	0.98	2.36	1.53	1.27	1.3	1	0	0
Iasi	3.73	4.97	2.8	5.08	7.19	3.8	4.41	3.12	0	0	1
Calarasi	1.36	1.34	2.2	0.98	1.12	1.59	1.35	1.26	1	0	0
Constanta	3.26	3.41	2.37	3.26	3.6	3.51	3.71	2.9	0	1	0
Covasna	1.14	1.11	2.13	0.82	0.45	1.08	1.1	1.11	0	1	0
Dambovita	2.23	2.28	2.25	1.9	1.8	2.61	2.31	2.57	0	1	0
Dolj	3.09	2.97	2.48	2.95	1.94	3.38	3.12	6.96	0	1	0
Galati	2.67	2.46	2.33	2.5	4.26	2.65	2.59	4.1	0	1	0
Giurgiu	1.24	1.17	2.39	0.8	0.62	1.46	1.13	0.64	1	0	0
Gorj	1.66	1.36	2.38	1.3	0.76	1.71	1.68	1.84	1	0	0
Harghita	1.84	1.67	2.1	1.52	0.39	1.62	1.69	2	1	0	0
Hunedoara	1.59	1.63	2.17	1.72	2.13	1.99	1.83	2	1	0	0
Ialomita	1.29	1.23	2.18	0.92	0.74	1.44	1.24	1.64	1	0	0
Vaslui	2.11	2.21	2.22	1.44	3.4	1.98	2.12	3.89	0	1	0
Ifov	1.89	2.47	2.83	1.05	1.19	1.99	1.77	0.3	1	0	0
Maramures	2.67	2.35	2.18	2.1	1.67	2.42	2.4	2.01	0	1	0
Mehedinti	1.53	1.09	2.22	0.89	0.8	1.34	1.22	2.69	1	0	0
Mures	2.69	2.88	2.45	2.76	1.69	2.82	2.85	2.47	0	1	0
Neamt	1.97	2.27	2.14	1.91	2.88	2.44	2.35	2.85	0	1	0
Olt	2.3	1.68	2.4	1.24	1.08	2.23	1.98	3.43	0	1	0
Prahova	3.31	3.15	2.52	2.81	2.49	3.81	3.54	2.76	0	1	0
Satu Mare	1.87	1.73	2.28	1.3	1.86	1.77	1.21	1.41	0	1	0
Salaj	1.59	1.31	2.25	1.02	0.57	1.11	1.77	1.68	1	0	0
Sibiu	2.56	2.25	2.66	3.12	2.72	1.94	2.17	1.34	0	1	0
Teleorman	1.99	1.31	2.11	1.01	1.55	1.96	1.41	3.42	0	1	0
Tulcea	1.16	0.83	2.32	0.88	1.17	1.1	0.99	1.13	1	0	0
Valcea	1.84	1.41	2.16	1.22	1.02	1.83	1.71	1.77	1	0	0
Vrancea	1.71	1.53	2.14	1.19	1.43	1.76	1.66	1.92	1	0	0

In SAS Enterprise Miner the initial data were imported into the work.ok3 file (Figure 3), then the data partition module was called and after that we will associate the module for neural network. Most of the initial data were used to train the network, and 10% was used to verify the results. After the neural network was trained, we called the Reporter module to retain in a report as a web page the main output information about the studied problem.

Figure 3. The work diagram resulted from SAS Enterprise Miner

In SAS Enterprise Miner the initial variables were divided into two categories: input type variables containing the indicators from I_1 to I_8 described in table 1 and target type variables, being represented by the classes found after performing the cluster analysis (Figure 4).

Figure 4. Multilayer perceptron neural network with a hidden layer of neurons

The input data have values within a well-established range, and the target data are binary with two values: 1 or 0. No data processing will be performed in the input layer, this being done in the intermediate and output layer. At the level of each neuron in the intermediate and final layers there are two types of processing: linear processing (neuron activation) and nonlinear processing (neuron output). [23], [24]

For any neuron from the hidden layer it can be defined the activation function as in equation (1) when the set x is applied to the input x^m . [25]

$$net_j^h = \sum_{i=0}^8 w_{ji}^h \cdot I_{-i}, j=1..4 \quad (1)$$

The vector $\{w_{ji}^h\}, j=1..4, i=1..8$ from the equation (1) represents the set of weights of the intermediate layer (Figure 4) and $I^m = (I_{-1}, I_{-2}, I_{-3}, I_{-4}, I_{-5}, I_{-6}, I_{-7}, I_{-8})^m$ is the input set and the index h comes from *hidden*. The output values (noted with i^j) are calculated according to equation (2).

$$i_j = f_j^h(net_j^h), j=1..4 \quad (2)$$

The function f_j^h is hyperbolic tangent, $f: R \rightarrow R, f(x) = th(x)$.

The neural network will have a single output parameter called belonging to a class and will be studied in depending on the classes initially chosen in the analysis: c1, c2 and c3 (Figure 4). For a k neuron in the output layer activation function can be calculated when applying vector I to the input as in formula (3). [26]

$$net_k^o = \sum_{j=1}^4 w_{kj}^o \cdot I_j, k=1,2,3 \quad (3)$$

The values $\{w_{kj}^o\}, k=1,2,3; j=1...4$ represent the weights of the output layer. The output value of the output neuron k is determined by the activation net_k^o and is expressed by the formula (4).

$$y_k^m = f_k^o(net_k^o), k=1,2,3 \quad (4)$$

The function f_k^o is hyperbolic tangent, a nonlinear function.

The training algorithm of the multilayer perceptron network has as main objective the minimization of the error on the training set (5).

$$E = \frac{1}{K} \sum_{p=1}^K E_p \quad (5)$$

K is the number of training sets and the error of the training set p is defined in formula (6).

$$E_p = \frac{1}{2} \sum_{k=1}^M (y_k^p - y_k^m)^2 \quad (6)$$

The objective function is deviance, which involves calculating the difference between the current probability of the neural network and the probability for a saturated model. This function cannot be negative, the value of zero reaches only for a data set without noise in other words when the network is fully trained. The data selection criterion is average error, this indicates the average error of assigning a county to a class, and the neural network entrainment algorithm was standard backpropagation.

The backpropagation algorithm involves performing the next steps [25], [27], [28]:

1. The first set is chosen from the training matrix of the neural network. We have the input $I^m = (I_1^m, I_2^m \dots I_8^m)$ for which the ideal output is known as $c^m = (c1^m, c2^m, c3^m)$. The set of weights is randomly generated in the interval [-1,1], both the weights of the intermediate layer ($\{w_{ji}^h\}, j=1...4, i=1...8$) and the weights of the output layer ($\{w_{kj}^o\}, k=1,2,3; j=1...4$).

2. The values of the activation and output functions of the neurons in the intermediate layer must be calculated according to formula (7).

$$net_j^h = \sum_{i=1}^8 w_{ji}^h \cdot I_i^m, j=1...4 \quad i_j = f_j^h(net_j^h), j=1...4 \quad (7)$$

3. Next it should be calculated the values of the activation and output functions of the neurons in the output layer as is presented in formula (8).

$$net_k^o = \sum_{j=1}^4 w_{kj}^o \cdot I_j \quad c_k^m = f_k^o(net_k^o), k=1,2,3. \quad (8)$$

4. We will calculate the error for the output layer is presented in equation (9).

$$\delta_k^o = (ck - ck^m) \cdot f_k^o \cdot \dot{f}_k^o(net_k^o), k=1,2,3 \quad (9)$$

The error for the intermediate layer is calculated too in this step (10).

$$\delta_j^h = \dot{f}_j^h(net_j^h) \cdot \sum_{k=1}^3 \delta_k^o \cdot w_{kj}^o, j=1...4. \quad (10)$$

5. We adjust the weights in the output layer after formulas (11) and (12) where $\eta, 0 < \eta < 1$ is the learning rate.

$$w_{kj}^o(t+1) = w_{kj}^o(t) - \eta \cdot \frac{\partial E}{\partial w_{kj}^o}, k=1...3, j=1...4 \quad (11)$$

$$\frac{\partial E}{\partial w_{kj}^o} = (ck - ck^m) \cdot \frac{\partial f_k^o}{\partial w_{kj}^o} = -(ck - ck^m) \cdot \frac{\partial f_k^o}{\partial (net_k^o)} \cdot \frac{\partial (net_k^o)}{\partial w_{kj}^o} = (ck - ck^m) \cdot \frac{\partial f_k^o}{\partial (net_k^o)} \cdot \frac{\partial}{\partial w_{kj}^o} \cdot \sum_{j=1}^4 w_{kj}^o \cdot I_j = (ck - ck^m) \cdot \dot{f}_k^o(net_k^o) \cdot I_j = -\delta_k^o \cdot I_j \quad (12)$$

The formula for adjusting the weights related to the output layer is (13).

$$w_{kj}^o(t+1) = w_{kj}^o(t) - \eta \cdot \delta_k^o \cdot I_j, k=1...3, j=1...4. \quad (13)$$

6. We will adjust the weights of the intermediate layer after the formula (14), having the learning rate 0.2.

$$w_{ji}^h(t+1) = w_{ji}^h(t) - \eta \cdot \frac{\partial E}{\partial w_{ji}^h} = w_{ji}^h(t) + \eta \cdot \delta_j^h \cdot I_i, j=1...4, i=1...8. \quad (14)$$

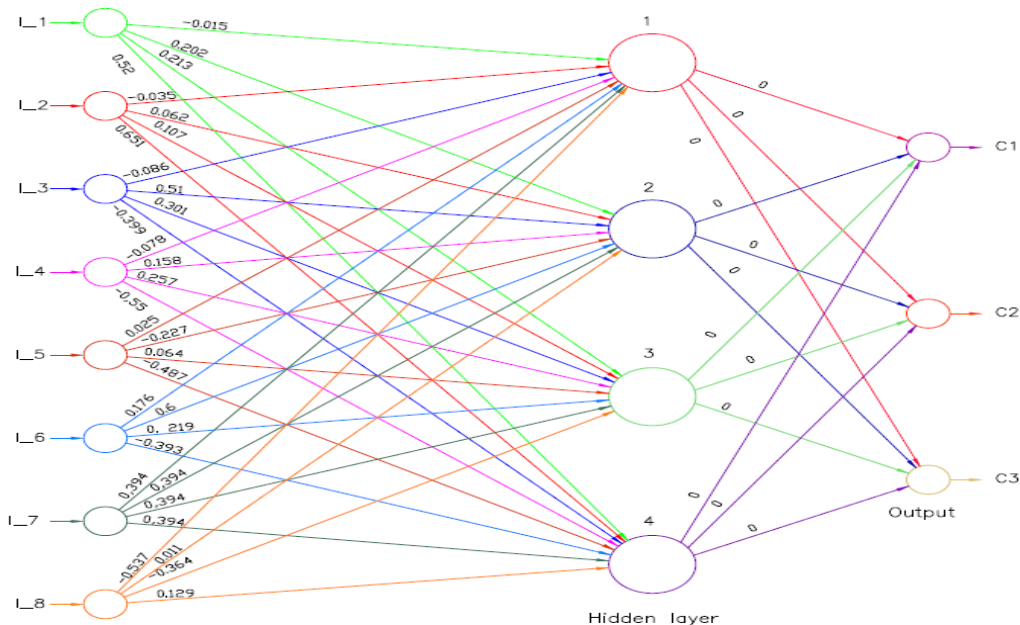
7. The error of the training vector is calculated as in equation (15).

$$E = \frac{1}{2} \sum_{k=1}^M (ck - ck^m)^2 \quad (15)$$

The error is propagated in the opposite direction, backwards, thus the weights will be adjusted in turn from the intermediate layer to the output layer and from the input layer to the intermediate layer.

8. If there are more items in the training set, we will resume the algorithm for a new item. Finally, the error corresponding to an epoch is calculated as the simple arithmetic mean of the errors due to each element in the training set. In general, the algorithm stops either when the value of the error has reached the minimum level, or when the maximum number of training epochs set by the user has been reached.

Figure 5. The neural network before training



The specific indicators before the training of the neural network with four neurons on the intermediate layer have the following values (Figure 5):

1. Weights from the input layer to the intermediate layer of neurons:

- the weights from the input layer to neuron 1 from the intermediate layer are: -0.015, -0.035, -0.086, -0.078, 0.025, 0.176, -0.537, 0.279;
- the weights from the input layer to neuron 2 from the intermediate layer are: -0.202, 0.062, -0.51, 0.158, -0.227, 0.6, 0.011, 0.34;
- the weights from the input layer to neuron 3 from the intermediate layer are: -0.213, 0.107, 0.301, 0.257, 0.064, 0.219, -0.364, 0.173; and
- the weights from the input layer to neuron 4 from the intermediate layer are: -0.52, 0.651, -0.399, -0.55, -0.487, -0.393, 0.129, -0.394.

2. The weights from the intermediate layer of neurons to the initial layer (consisting of classes) are initially equal to 0.

3. Activation thresholds of neurons in the intermediate and final layers are: for the intermediate layer the thresholds are: 1,878, -383.862, -1.484, 1.422; and for the final layer the thresholds have the following values: -0.154, -0.15, -2.484.

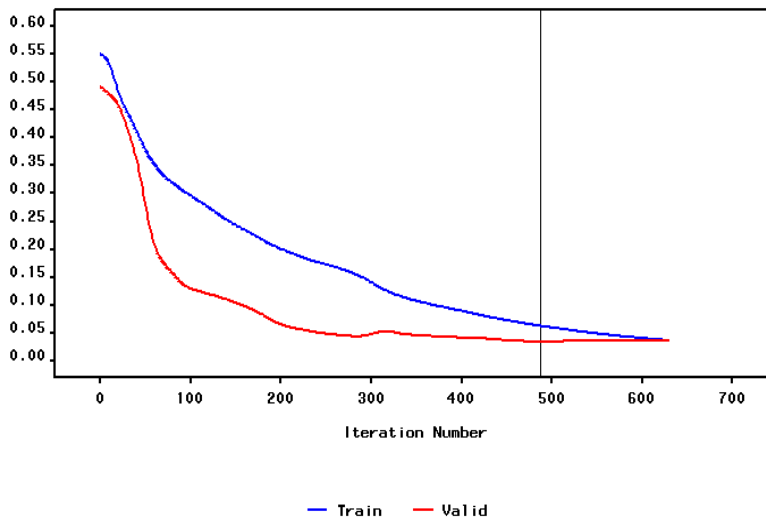
4. The value of the objective function is equal to 0.55.

Table 4. Initial settings for applying the Standard Backpropagation algorithm

Minimum Iterations	0
Maximum Iterations	630
Maximum Function Calls	2147483647
Maximum CPU Time	604800
ABSGCONV Gradient Criterion	0,00001
GCONV Gradient Criterion	1E-8
ABSFCONV Function Criterion	0
FCONV Function Criterion	0,0001
FCONV2 Function Criterion	0
FSIZE Parameter	0
ABSXCONV Parameter Change Criterion	0
XSIZE Parameter	0
ABSCONV Function Criterion	-1.34078
Singularity Tolerance (SINGULAR)	1E-8
Prop_learn	0.2
Prop_momentum	0

After completing the process of training the neural network with four neurons in the intermediate layer, there were 630 iterations until the desired result was reached (meeting the convergence criterion, variable ABSGCONV) and successful training of the network. Both the objective function and the gradient function were activated by 630 times, and the last value of the objective function was 0.0136.

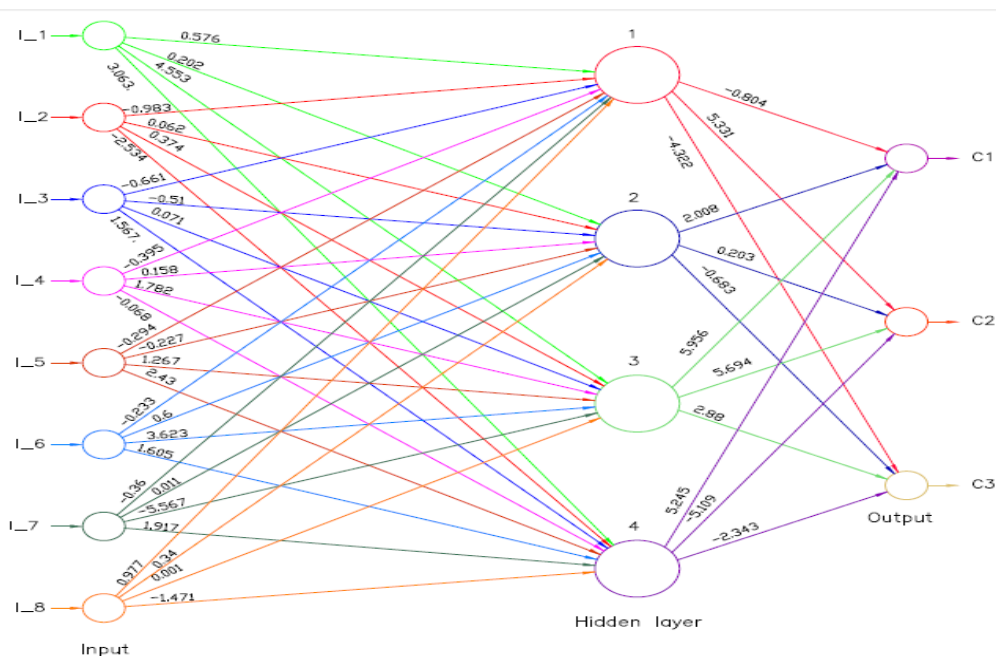
Figure 6. The evolution of the average error for the neural network during the training epochs



If at the first iterations the graphs made for the average drive error and the average validation error have similar evolutions (Figure 6), the errors being somewhat with similar values, after iteration 100 it can be seen that the average validation error decreases suddenly and remains on a slightly descending slope until after iteration 600, when the two errors have approximately equal values. The average training error of the

neural network has a constantly decreasing evolution, having a steeper slope up to iteration 150. At the end of the training cycle, respectively validation of the neural network, the two errors have values close to zero, which was desirable, network training ending successfully.

Figure 7. The trained neural network



The indicators specific to the neural modeling of the network with four neurons in the intermediate layer have the following values (Figure 7):

1. weights from the input layer to the intermediate layer of neurons:

- the weights from the input layer to the first neuron from the intermediate layer are: 0.576, -0.983, -0.661, -0.395, -0.294, -0.233, -0.36, 0.977;
- the weights from the input layer to the second neuron from the intermediate layer are: 0.202, 0.062, -0.510, 0.158, -0.227, 0.6, 0.011, 0.34;
- the weights from the input layer to the third neuron from the intermediate layer are: 4.553, 0.374, 0.071, 1.782, -1.267, 3.623, -5.567, 0.001; and
- the weights from the input layer to the fourth neuron from the intermediate layer are: 3.063, -2.534, 1.567, -0.068, -2.433, 1.605, 1.917, -1.471.

2. the weights from the intermediate layer of neurons to the final layer, consisting of classes are: -0.804, 2.008, 5.956, 5.245 (for c1); 5.331, 0.203, 5.694, -5.109 (for c2); -4.322, -0.683, 2.88, -2.343 (for c3).

3. the activation thresholds of neurons in the intermediate and final layers: for the intermediate layer the values of these thresholds are: 3.249, -383.862, -0.582 and for the final layer the thresholds have the following values: -2.162, -0.358, -1.801.

4. the value of the objective function is 0.036.

In order to be able to use the already trained neural network in the future, the program SAS Enterprise Miner offers the user the network transfer functions, as follows:

1. For the hidden layer of neurons (H1, H2, H3 and H4) the transfer functions together with the related scores are presented in equations (20)-(24).

$$H1=0.539 \cdot S_{I_1}-0.953 \cdot S_{I_2}-0.652 \cdot S_{I_3}-0.357 \cdot S_{I_4}-0.26 \cdot S_{I_5}-0.236 \cdot S_{I_6}-0.388 \cdot S_{I_7}+0.96 \cdot S_{I_8}; \quad (20)$$

$$H2=0.202 \cdot S_{I_1}+0.062 \cdot S_{I_2}-0.51 \cdot S_{I_3}+0.158 \cdot S_{I_4}-0.227 \cdot S_{I_5}+0.6 \cdot S_{I_6}+0.011 \cdot S_{I_7}+0.34 \cdot S_{I_8}; \quad (21)$$

$$H3=4.411 \cdot S_{I_1}+0.253 \cdot S_{I_2}+0.023 \cdot S_{I_3}+1.574 \cdot S_{I_4}-1.287 \cdot S_{I_5}+3.233 \cdot S_{I_6}-4.989 \cdot S_{I_7}-0.038 \cdot S_{I_8}; \quad (22)$$

$$H4=2.342 \cdot S_{I_1}-1.958 \cdot S_{I_2}+1.215 \cdot S_{I_3}-1.018 \cdot S_{I_4}-2.007 \cdot S_{I_5}+1.563 \cdot S_{I_6}+1.563 \cdot S_{I_7}-0.989 \cdot S_{I_8}; \quad (23)$$

$$\begin{aligned} H1 &= 3.146 + H1; & H2 &= -383.862 + H2; \\ H3 &= -0.583 + H3; & H4 &= 2.417 + H4. \end{aligned} \quad (24)$$

2. For the nodes in the output layer of the neural network (c1, c2 and c3) the software SAS Enterprise Miner has the functions and with their corresponding weights presented in equations (25)-(27).

$$Pc1=-0.749 \cdot H1+1.64 \cdot H2-5.353 \cdot H3+4.325 \cdot H4; Pc1=-1.794 + Pc1; \quad (25)$$

$$Pc2=5.009 \cdot H1+0.453 \cdot H2+5.049 \cdot H3-4.098 \cdot H4; Pc2=-0.607+Pc2; \quad (26)$$

$$Pc3=-3.8 \cdot H1-0.662 \cdot H2+2.789 \cdot H3-2.405 \cdot H4; Pc3=-1.822+Pc3. \quad (27)$$

After the training of the neural network, it can be seen that the groups of counties remained approximately the same as those resulting from the cluster analysis. The advantage of using neural networks is that once trained and, as a result, found the transfer functions and those that calculate the scores, they can be used in the future, without needed to resume the whole algorithm.

The classes resulting from the analysis can be characterized as follows:

1. First class (c1) which has counties with low economic and social development, in the sense that the average net earnings are low, emigration is high, the number of unemployed is high and, implicitly, the number of schools is small;

2. Second class (c2) is characterized by an average economic and social development. The values of the growth indicators used in the analysis are higher than in the previous class. Thus, in the counties that are in this class we can also find

independent higher education units or branches of large universities in large cities. The unemployment rate, permanent and temporary emigration levels are lower than the values of the same indicators found in class 1; and

3. Third class (c3) is made up of counties with large urban agglomerations, with strong university centers. As a consequence of the urban agglomeration, the number of live births is high, so it can be said that in the future these counties also tend to keep their place in this class. This class is also characterized by the small number of temporary and permanent emigration, the average net earnings is higher in Romania, the number of schools is high, as well as the number of computers used in education.

Regarding the outlier found at the beginning of the analysis, being Bucharest, at a closer analysis of the values of the indicators chosen for the application, we can be seen that they have much higher values compared to other counties. From the point of view of the economic and social reality, Bucharest can be associated with the third class (c3) which contains counties with above average development.

The transition of a county from one lower class to another upper one can be done through medium and long term economic and social policies. Thus, in order to pass a county from first class (c1) to second class (c2), it must increase the values of the following indicators: average net earnings through policies to attract new investments at the local level, the number of school units and the number of computers for education. From ancient times the people went to the places where they were well. So, as a consequence of the firm investments made in a certain county, immediately the number of unemployed will decrease because labor is needed, and any investor prefers local labor because it involves less transportation costs and others. Over time, the number of temporary emigrants will also decrease, and in the long run the final emigration will also decrease. Thus, people returning to their native counties, automatically and the number of children born alive will increase this ensures the preservation of that county in the desired class in the long run.

All locally adopted policies have consequences in the cascade. Even if the model allows the transition from one class to another to be done exclusively by changing a single indicator, from the point of view of economic reality it is almost impossible to achieve this because a measure adopted has greater or lesser implications on all indicators.

4. Conclusion

With the change of technology, the increase of computing speed and the automation of many activities with the help of computers, education and implicitly the education system change too. [29-31] The transmission of information, knowledge, skills is a challenge in the current conditions of technological progress. [32] Every person, whether he is a student from the pre-university system, a student in the university system or an adult enrolled in various training courses, has a unique way of learning, accumulating and applying knowledge. Through the policies adopted at the level of Romania, region, county, local community, the education system represented by educational institutions must adapt and respond to these requirements, the development of complex thinking and different skills being prerequisites for long-term development. The policies adopted at the level of the education system must be in

agreement both with the other policies adopted at the level of a country, but also with those adopted at the level of the European Union. Even if in the short term it seems that the education system is a big consumer of resources, in the medium and long term, the degree of development of a country is measured in overwhelming proportion by the level of training of its specialists. [33] The more refined know-how skills the specialists have, the more the amortization of the costs with their education will be achieved in a shorter time and, implicitly, the added value they will bring will be higher.

References

1. I. Cancarevic, L. Plichtová, H. Tohid, "Similarities and the Differences in the Education Systems Around the World", *International Medical Graduates in the United States*, Springer, 2021, https://doi.org/10.1007/978-3-030-62249-7_2
2. B. Dallago, "European Higher Education: Challenges and Achievement", *Comparative Economic Studies in Europe, Studies in Economic Transition*, Palgrave Macmillan, Cham, pp. 111-147, 2021, https://doi.org/10.1007/978-3-030-48295-4_7
3. H. Shvindina, S. Kotenko, A. Vorontsova, V. Gordienko, Y. Petrusenko, D. Jha, "Reforming the education system to prevent labor migration", *Knowledge and Performance Management*, vol. 4, no 1, pp. 52-66, 2020, doi:10.21511/kpm.04(1).2020.05
4. A. Vorontsova, H. Shvindina, T. Mayboroda, A. Mishenina, I. Heiets, "The impact of state regulation in a sphere of education on sustainable development of national economy", *Problems and Perspectives in Management*, vol. 18, no. 4, pp. 275-288, 2020, doi:10.21511/ppm.18(4).2020.23
5. Shubham Joshi et al, "Evaluating Artificial Intelligence in Education for Next Generation", *Journal of Physics: Conference Series*, Vol. 1714, 2nd International Conference on Smart and Intelligent Learning for Information Optimization (CONSILIO) 2020 24-25 October 2020, Goa, India.
6. Z. Xin, D. Wang, "Application of artificial intelligence algorithms in image processing", *Journal of Visual Communication and Image Representation*, Vol. 61, pp. 42-49, 2019, ISSN 1047-3203, <https://doi.org/10.1016/j.jvcir.2019.03.004>
7. R. C. Suganthe, R. S. Latha, M. Geetha, G. R. Sreekanth, "Diagnosis of Alzheimer's Disease from Brain Magnetic Resonance Imaging Images using Deep Learning Algorithms," *Advances in Electrical and Computer Engineering*, vol. 20, no. 3, pp. 57-64, 2020, doi:10.4316/AECE.2020.03007
8. G. Mukhtar, S. Farhan, "Convolutional Neural Network Based Prediction of Conversion from Mild Cognitive Impairment to Alzheimer's Disease: A Technique using Hippocampus Extracted from MRI," *Advances in Electrical and Computer Engineering*, vol. 20, no. 2, pp. 113-122, 2020, doi:10.4316/AECE.2020.02013
9. K. Abaci, V. Yamacli, "Hybrid Artificial Neural Network by Using Differential Search Algorithm for Solving Power Flow Problem," *Advances in Electrical and Computer Engineering*, vol. 19, no. 4, pp. 57-64, 2019, doi:10.4316/AECE.2019.04007
10. S. Zhang, X. Guo, X. Zhang, "Modeling of Back-Propagation Neural Network Based State-of-Charge Estimation for Lithium-Ion Batteries with Consideration of Capacity Attenuation," *Advances in Electrical and Computer Engineering*, vol. 19, no. 3, pp. 3-10, 2019, doi:10.4316/AECE.2019.03001

11. E. J. Gill, E. B. Singh, E. S. Singh, “Training back propagation neural networks with genetic algorithm for weather forecasting”, IEEE 8th International Symposium on Intelligent Systems and Informatics, Subotica, pp. 465-469, 2010, doi: 10.1109/SISY.2010.5647319
12. A. Kardan, H. Sadeghi, S.S. Ghidary, M.R.F. Sani, “Prediction of student course selection in online higher education institutes using neural network”, Computers & Education, Vol. 65, pp. 1-11, 2013, ISSN 0360-1315
13. Institutul National de Statistica, baze de date statistice - Tempo Online, <http://statistici.insse.ro:8077/tempo-online/>, 2021
14. F. Murtagh, P. Legendre, “Ward’s Hierarchical Agglomerative Clustering Method: Which Algorithms Implement Ward’s Criterion?”, Journal of Classification, Vol. 31, pp. 274–295, 2014, <https://doi.org/10.1007/s00357-014-9161-z>
15. O. Yu, M. Kon, “Two clustering methods based on the Ward's method and dendrograms with interval-valued dissimilarities for interval-valued data”, International Journal of Approximate Reasoning, Vol. 129, pp. 103-121, 2021, ISSN 0888-613X, <https://doi.org/10.1016/j.ijar.2020.11.001>
16. M. C. Thrun, A. Ultsch, “Swarm intelligence for self-organized clustering”, Artificial Intelligence, Vol. 290, 2021, ISSN 0004-3702, <https://doi.org/10.1016/j.artint.2020.103237>
17. R. Ahuja, P.P. Agarwal, T. Kumar, “Clustering Methods Analysis in the E-Learning”, Kumar A., Mozar S. (eds) ICCCE 2020, Lecture Notes in Electrical Engineering, Vol. 698, 2021, Springer, Singapore. https://doi.org/10.1007/978-981-15-7961-5_118
18. A. Adolfsson, M. Ackerman, N. C. Brownstein, “To cluster, or not to cluster: An analysis of clusterability methods”, Pattern Recognition, Vol. 88, pp. 13-26, 2019, ISSN 0031-3203, <https://doi.org/10.1016/j.patcog.2018.10.026>
19. P. Koehn, “Neural Machine Translation”, pp.11-18, 2017, <https://arxiv.org/pdf/1709.07809.pdf>
20. P. Diaconescu, V. E. Neagoie, “Credit Scoring Using Deep Learning Driven by Optimization Algorithms”, 12th International Conference on Electronics, Computers and Artificial Intelligence (ECAI), Bucharest, Romania, pp. 1-6, 2020, doi: 10.1109/ECAI50035.2020.9223139
21. T. Damrongsakmethee, V. E. Neagoie, “Stock Market Prediction Using a Deep Learning Approach”, 12th International Conference on Electronics, Computers and Artificial Intelligence (ECAI), Bucharest, Romania, pp. 1-6, 2020, doi: 10.1109/ECAI50035.2020.9223142
22. E. Faraggi, R.L. Jernigan, A. Kloczkowski, “A Hybrid Levenberg–Marquardt Algorithm on a Recursive Neural Network for Scoring Protein Models”, In: Cartwright H. (eds) Artificial Neural Networks. Methods in Molecular Biology, Vol 2190, 2021, Humana, New York, https://doi.org/10.1007/978-1-0716-0826-5_15
23. Y. Yang, J. Wang, “Forecasting wavelet neural hybrid network with financial ensemble empirical mode decomposition and MCID evaluation”, Expert Systems with Applications, Vol. 166 2021, ISSN 0957-4174, <https://doi.org/10.1016/j.eswa.2020.114097>
24. K.K. Natarajan, J. Gokulachandran, “Application of artificial neural network techniques in computer aided process planning - a review”, International Journal of Process Management and Benchmarking, Vol. 11, No. 1, 2020, ISSN: 1460-6739, 1741-816X

25. L. Chankyu, S. S. Shakib, P. Priyadarshini, S. Gopalakrishnan, R. Kaushik, “Enabling Spike-Based Backpropagation for Training Deep Neural Network Architectures”, *Frontiers in Neuroscience*, vol. 14, 2020, ISSN 1662-453X
26. H. Wang, R. Czerminski, A.C. Jamieson, “Neural Networks and Deep Learning”, *The Machine Age of Customer Insight*, Emerald Publishing Limited, pp. 91-101, 2021, <https://doi.org/10.1108/978-1-83909-694-520211010>
27. X. Jin, Q. Liu, H. Long, “Impact of cost–benefit analysis on financial benefit evaluation of investment projects under back propagation neural network”, *Journal of Computational and Applied Mathematics*, Vol. 384, 2021, ISSN 0377-0427
28. S.M. D’Souza, K.S. Reddy, “Design and Development of Efficient Cost-Saving Algorithms for Guiding Customer Purchasing Patterns in Modern Consumerism Scenario Using Feed Forward Back Propagation Neural Networks”, *Smart Trends in Computing and Communications: Proceedings of SmartCom 2020*, Smart Innovation, Systems and Technologies, Vol. 182, 2021, Springer, Singapore. https://doi.org/10.1007/978-981-15-5224-3_47
29. EduPedu, <https://www.edupedu.ro/>, 2021
30. Enciclopedia Romaniei, Sistemul de invatamant din Romania, 2021, http://enciclopediaromaniei.ro/wiki/Sistemul_de_%C3%AEenv%C4%83%C5%A3%C4%83m%C3%A2nt_din_Rom%C3%A2nia
31. National Education Systems - EURYDICE, 2021 https://eacea.ec.europa.eu/national-policies/eurydice/content/organization-education-system-and-its-structure-64_ro
32. D. G. Broo, U. Boman, M. Törngren, “Cyber-physical systems research and education in 2030: Scenarios and strategies”, *Journal of Industrial Information Integration*, Vol. 21, 2021, ISSN 2452-414X, <https://doi.org/10.1016/j.jii.2020.100192>
33. P. A. Otero Cano, E. C. Pedraza Alarcón, “Recommendation Systems in Education: A review of Recommendation Mechanisms in E-learning Environment”, *Revista Ingenierías Universidad De Medellín*, Vol. 20, No. 38, Pp. 147-158, 2020, <https://doi.org/10.22395/v20n38a9>