Applying Recurrence Plots to Classify Time Series

Lyudmyla Kirichenko, Tamara Radivilova and Juliia Stepanenko

Kharkiv National University of Radio Electronics, 14 Nauky ave., Kharkiv, 61166, Ukraine

Abstract

The article describes a new approach to the classification of time series based on the construction of their recurrence plots. After transforming the time series into recurrence plots, two approaches are applied for classification. In the first case, quantitative recurrence characteristics are used for classification as features. In the second case, the time series is presented in the form of a black and white image of its recurrence plot. A convolutional neural network is used as an image classifier. The data for the classification are the electrocardiograms realizations of 100 values, which contained records of healthy people and patients with a diagnosis of ischemia. Research results showed the advantages of classifying images of recurrence plots, indicate a good classification accuracy in comparison with other methods and the potential capabilities of this approach.

Keywords 1

Time series classification, machine learning classification, recurrence plot, ECG time series, quantitative recurrence characteristic

1. Introduction

Analysis and classification of time series plays an important role in many areas of science and technology: in biology, seismology, physics, economics, in particular, in solving problems of diagnostics and forecasting. When time series classifying using machine learning, most often a set of some statistical features is extracted from the time series, which is input of the classifier. Various methods can be used as classifiers, including widespread neural networks [1,2].

A new and non-trivial approach to the classification of time series is the transformation of a series into another structure, for example a graph, a surface or a table, and the classification of features obtained on the basis of this structure [3-5]. If the structure obtained from the time series can be visualized, that is, represented as an image, then the resulting images can be classified by computer vision methods [3, 6-8].

One of the methods that allows visualizing the time series dynamics is the method of recurrence plots, proposed for the analysis of nonlinear dissipative systems and widely used in other areas of research [9-12]. In recent years, visualization of recurrence plots has been used to analyze and classify time series of various nature.

In this paper, the classification of electrocardiograms (ECG) is considered. Cardiac diseases are referred to those diseases that respond well enough to treatment if they are at an early stage. The main diagnostic method in cardiology for a long time has been ECG, which is widely used for the functional study of the cardiovascular system.

The purpose of the presented work is to classify ECG time series based on the construction of recurrence plots. After transforming the time series into recurrence plots, two approaches are applied for classification: the use of quantitative recurrence characteristics as classifier features and the recognition of recurrence plot images using a convolutional neural network.

COLINS-2021: 5th International Conference on Computational Linguistics and Intelligent Systems, April 22–23, 2021, Kharkiv, Ukraine
EMAIL: lyudmyla.kirichenko@nure.ua (L. Kirichenko); tamara.radivilova@nure.ua (T. Radivilova); yuliia.stepanenko@nure.ua (J. Stepanenko)
ORCID: 0000-0002-2780-7993 (L. Kirichenko); 0000-0001-5975-0269 (T. Radivilova)

2. Recurrence plots

Recurrence analysis is one of the nonlinear dynamic's methods used for time series analysis and is designed to identify non-obvious dependencies in the time series dynamics. Recurrence analysis of time realizations is based on the fundamental property of the trajectories of dissipative systems: repeat of states (recurrence).

This property was formalized in the "recurrence theorem", which says that if a system reduces its dynamics to limited subset of a n- dimensional space, then the system with a probability almost equal to 1, returns arbitrarily close to some initially specified state [11].

Let's consider some time realization, represented by its values $\{x_1, x_2, ..., x_i, ..., x_N\}$. Recurrence states of a point x_i are states x_j that fall into n-dimensional neighborhood of x_i with a given radius ε .

The recurrence of states x_i, where i is a time moment, is reproduced using a two-dimensional square matrix (recurrence plot) with black and white dots, where both coordinate axes i and j are discrete time axes, black dots with coordinates (i, j) indicate the presence of recurrence between points x_i and x_j . Thus, the recurrence plot is a black and white image.

For clarity, Fig. 1 shows a recurrence plot of a sinusoidal periodic trajectory (left), and a plot of the stochastic realization of encephalogram [13] (right).

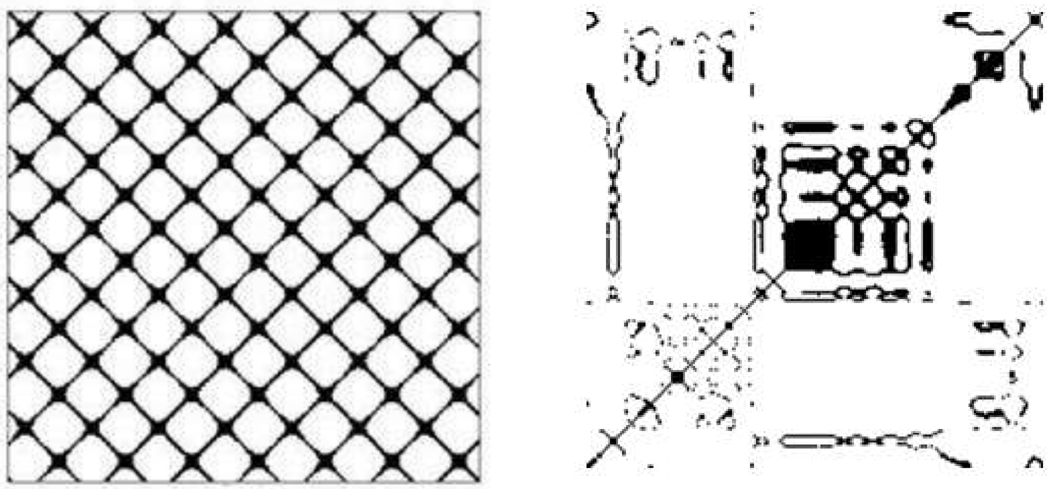

Figure 1: Recurrence plots: sinusoid (left) and encephalogram realization (right)

In [9-12], an approach was proposed for the numerical analysis of recurrence plots, which allows to obtain quantitative recurrence characteristics of a time series. Let us consider some quantitative characteristics that can be used as features for time series classification.

The most obvious characteristic is the recurrence rate (*RR*), which shows the density of points in a recurrence plot that corresponds to the probability of states repeating.

A number of characteristics are based on the calculation of the diagonal lines lengths L in the recurrence plot. The presence of a diagonal line corresponds to a situation when a some part of the phase trajectory repeats itself (within the specified accuracy ε), passing through the same region of the phase space at different time intervals. The average length of the diagonal lines *Lavg* corresponds to the average time during which the dynamics of the trajectory is repeated. Usually, random time series have a small length of diagonal lines, and a large number of separate recurrence points. Regular, in particular periodic time series, correspond to recurrence plots with long diagonal lines and a small number of separate recurrence points.

Shown in fig. 1 the recurrence plot of sine wave actually contains only diagonal lines that indicate the periodic nature of the trajectory of the system.

The derived characteristic from the lengths and number of diagonal lines is a measure of the system predictability (determinism, *DET*). It is based on the fact that the average length of the diagonal lines corresponds to the average predictability time of the system behavior. Entropy of the diagonal lines (*L_ENTR*) is calculated based on the frequency distribution of L and shows the complexity of the trajectory deterministic component.

Some of the quantitative characteristics are calculated on the basis of the vertical lines lengths *V* in the recurrence plot, which correspond to the trajectory being in the same state (laminarity, *LAM*) The value *LAM* indicates the presence of system conditions when the system movement stops or moves very slowly.

The average length of vertical structures (trapping time, *TT*) indicates the time that a trajectory can spend in the neighborhood ε of a certain state. Entropy of the vertical lines (*V_ENTR*) is calculated based on the frequency distribution of the vertical lines lengths and indicates the complexity of the laminar component of the trajectory.

Fig. 1 shows recurrence plot of the realization of an encephalogram, where both diagonal and vertical structures are present. Visually, you can determine that the lengths of the diagonal lines are small, and there are also a significant number of separated recurrence points in the structure.

Table 1 shows the values of the above-described quantitative characteristics corresponding to the recurrence plots shown in Fig. 1. Obviously, there is a significant difference between the characteristics of the deterministic and stochastic trajectories.

Table 1
Quantitative Recurrence Characteristics

	RR	Lavg	Det	L_ENTR	LAM	TT	V_ENTR
Sinusoid	0.12	39.76	0.998	0.03	0.67	0.92	0.024
Encephalogram	0.045	4.58	0.247	1.51	1.32	7.83	2.51

Thus, a time series can be represented by recurrence plot that reflects its dynamics. The classification of recurrence plots can be carried out on the basis of their quantitative characteristics, which act as features for the classifier. Another classification method could be to classify recurrence plot images using a convolutional neural network.

2. Convolutional Neural Networks

Convolutional neural network is a special architecture of artificial neural networks aimed at efficient image recognition. The idea behind convolutional neural networks is to alternate between convolutional layers, sub-sampling and regular layers of a neural network.

The structure of the network is unidirectional, in principle multilayer. For training, standard methods are used, most often the method of back propagation of an error. The function of activating neurons can be different, depending on the task solved by the neural network. Each fragment of the image is multiplied by the matrix (kernel) of the convolution element by the element, and the result is summed and written to the same position in the original image.

The investigated image is passed through a series of convolutional nonlinear, merge, and fully connected layers and an output is generated. The output can be a label or a probability of the class that best describes the image.

The first layer in the network is always convolutional. This is a set of functional cards of the same size. Each map has a synaptic core, which is a window that slides over the entire area of the preliminary map and finds certain features of objects. The neurons on the first convolutional layer are not associated with each pixel of the input image, but only with pixels in their own receptor fields. Further, the neuron of the second convolutional layer is connected only to the neurons inside the rectangular region of the first layer. The considered architecture of the neural network makes it possible to focus on low-level objects in the first hidden layer in order to further combine them into high-level objects.

The first level output is the second level input value. After applying a set of filters after the first layer, filters will be activated, which represent the properties of the highest level. The more

COLINS'2021, Volume II: Workshop. Kharkiv, Ukraine, April 22-23, 2021, ISSN 2523-4013
http://colins.in.ua, online

convolutional layers an image passes and the further it travels through the network, the more complex the characteristics are reflected in the object maps.

After the convolutional layers comes the pooling layer, the main task of which is to thin out the input image to reduce the computational load, memory consumption and the number of parameters, reducing the risk of overfitting.

The last type of layer is a regular multi-layer perceptron layer. The purpose of the layer is classification, it models a complex non-linear function, the optimization of which increases the quality of recognition. The output layer is connected with all neurons in the preceding layer. The number of neurons on the output layer is equal to the number of classification classes. [14, 15].

3. Description of the Experiment

3.1. Input Data

The input data for research in this work were data obtained from the repository "UEA & UCR Time Series Classification Repository" [16]. The dataset name is "ECG200".

It contains medical time series that are ECG realizations: 200 samples, of which 100 are intended for training the classifier and 100 for testing. Each series corresponds electrical activity recorded during the heartbeat and contains 100 values. The ECG realizations are divided into two classes: "norm" (class 0) and "ischemia" (class 1).

Fig. 2 shows schematic images of the ECG realizations for a healthy person and a patient diagnosed with ischemia [16].

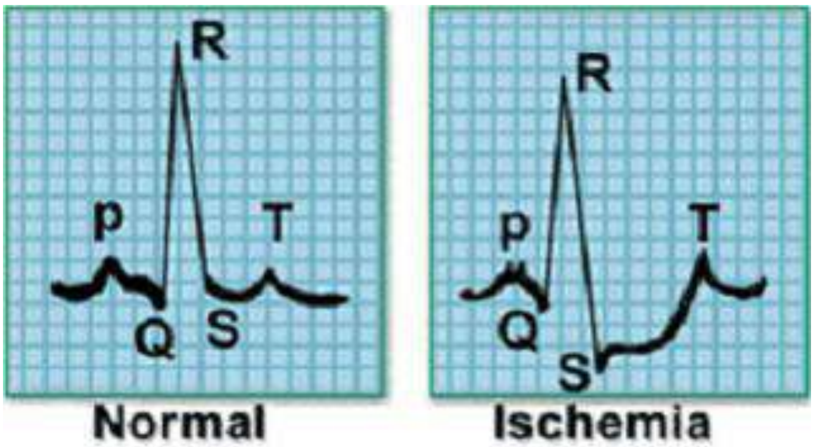

Figure 2: Schematic ECG, "Normal" and "Ischemia"

Table 2 shows the number of time series for classes “0” and “1”.

Table 2
Number of Time Series for Experiment

Dataset	Class 0	Class 1	Total
Train	31	69	100
Test	36	64	100
Total	67	133	200

Fig. 3 shows examples of ECG time series from the dataset, which are typical for a healthy person (class “0”) and person with ischemia disease (class “1”).

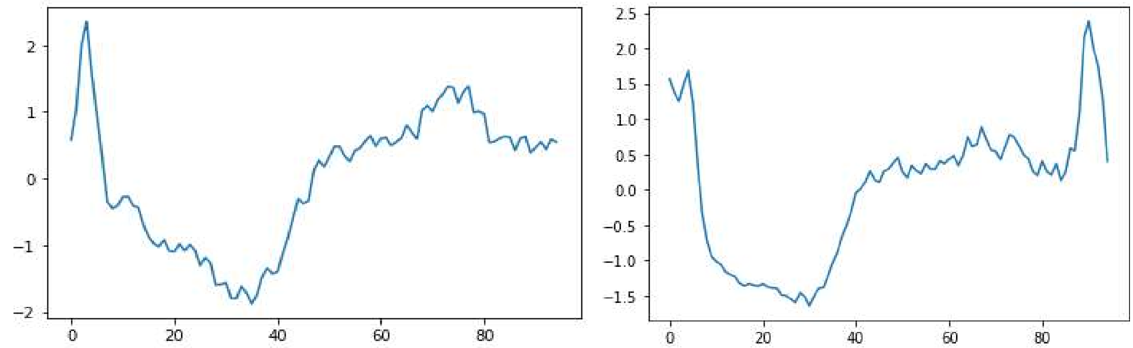

Figure 3: Realizations of ECG; Left - Class “0”, Right - Class “1”

Fig. 4 presents examples of the recurrence plots which correspond ECG time series of Fig. 3. It should be noted that, in contrast to the schematic image, the difference between the ECG time series of class "0" and class "1" is not visually observed, while the recurrence plots have visual differences.

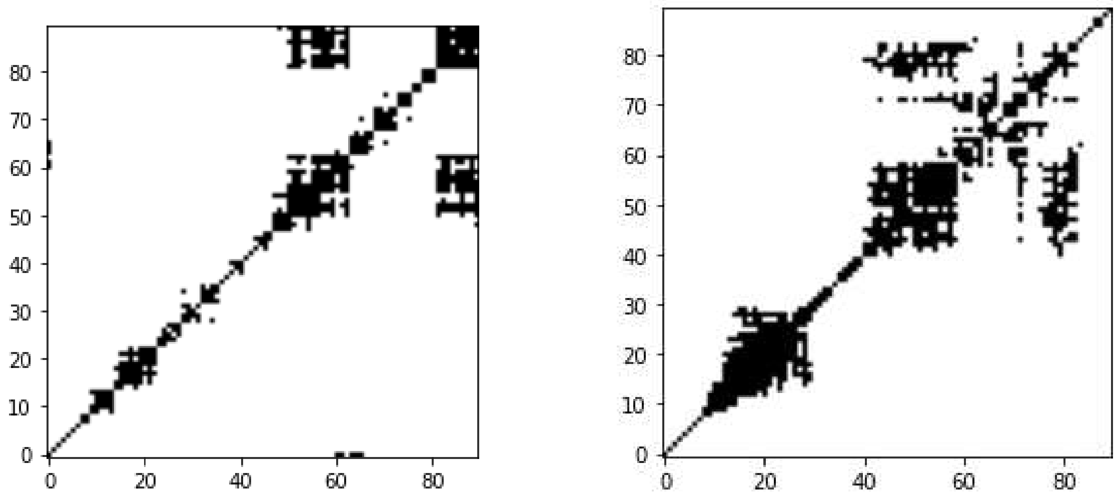

Figure 4: Recurrence Plots; Left - Class “0”, Right - Class “1”

3.2. Experiment

We need to solve the problem of binary classification of medical time series. For the classification, the Python language was chosen. Python is a high-level object-oriented programming language with strong dynamic typing. It is an open source language containing many libraries for processing and graphing data. Python is one of the most demanded and popular program language, as evidenced by numerous ratings and analysis of proposals on the software development market [17].

Time series classification was carried out by two methods. In the first case, the time series were transformed into recurrence plots, from which the quantitative characteristics were calculated. The obtained characteristics were the input features for the classifier.

The following sample recurrence measures were used as features: the recurrence rate *RR*, the predictability *DET*, the laminarity of the time series *LAM*, the maximum length of diagonal lines *Lmax*, the maximum length of vertical lines *Vmax*, inverse value of the maximum diagonal line length *DIV*, average length of a diagonal line *Lavg*, trapping time *TT*, entropy of diagonal lines *L_ENTR*, - entropy of vertical lines *V_ENTR*.

Fig. 5 shows several values of recurrence characteristics obtained from ECG realizations belonging to class "0".

	is_train	RR	DET	LAM	Lmax	Vmax	DIV	Lavg	TT	L_ENTR	V_ENTR	class
0	1.0	0.12889	0.91824	0.95115	26.0	25.0	0.03846	4.70968	6.32484	1.83398	2.07083	0.0
2	1.0	0.13630	0.89349	0.95924	28.0	18.0	0.03571	4.31429	5.16585	1.78724	2.10995	0.0
3	1.0	0.09333	0.93393	0.94180	30.0	27.0	0.03333	7.06818	7.82418	2.05041	2.44319	0.0
6	1.0	0.09432	0.78932	0.90969	25.0	21.0	0.04000	3.59459	4.04070	1.49819	1.79750	0.0
7	1.0	0.11531	0.83412	0.88330	39.0	20.0	0.02564	3.66667	4.91071	1.57012	2.06482	0.0

Figure 5: Recurrence Characteristics of Class “0”

Fig. 6 shows the values of the same recurrence characteristics obtained for class "1". As is clear from the above examples, the characteristic values of classes “0” and “1” differ from each other. For example, the average value *Lavg* for class "0" in the given five inputs is 4.67, which is higher than for class "1", where the average is correspondingly 3.15. Similar differences can be seen if we carry out calculations for all the given characteristics.

	is_train	RR	DET	LAM	Lmax	Vmax	DIV	Lavg	TT	L_ENTR	V_ENTR	class
1	1.0	0.03531	0.10204	0.31818	3.0	4.0	0.33333	2.50000	2.45946	0.69315	0.63964	1.0
4	1.0	0.06889	0.73077	0.83333	24.0	13.0	0.04167	3.71739	4.42857	1.32311	1.93830	1.0
5	1.0	0.06790	0.43913	0.55455	4.0	7.0	0.25000	2.58974	3.01980	0.96288	1.29269	1.0
8	1.0	0.07679	0.49248	0.66238	6.0	13.0	0.16667	2.56863	3.16923	0.95351	1.38169	1.0
9	1.0	0.09506	0.85588	0.92597	27.0	16.0	0.03704	4.40909	5.32090	1.85633	1.88740	1.0

Figure 6: Recurrence Characteristics of Class “1”

To carry out the classification in the first case, a fully connected multilayer perceptron with an activation function of the ReLU type was chosen [18]. This neural network is a versatile approximator and is capable of detecting hidden patterns in data. To prevent overfitting of the model and to increase the classification accuracy, several layers of batch normalization were included in the structure of the neural network [19].

In the second case, the classification was based on the recognition of images of recurrence plots. Input ECG time series for training and test samples were transformed into recurrence plots images. Some of the resulting images for both classes are shown in Fig. 7.

To create a neural network, the Keras library was used, which is the most popular for creating neural networks. The developed convolutional neural network contains five layers; the first two ones are convolutional. The output of the last layer is fed to a 2-sided softmax, which produces a distribution over 2 classes. Neurons in fully connected layers are connected to all neurons in the previous layer. The non-linear ReLU function is applied to the output of each convolutional and fully connected layer. The Adam stochastic optimization method was chosen as the training method [20].

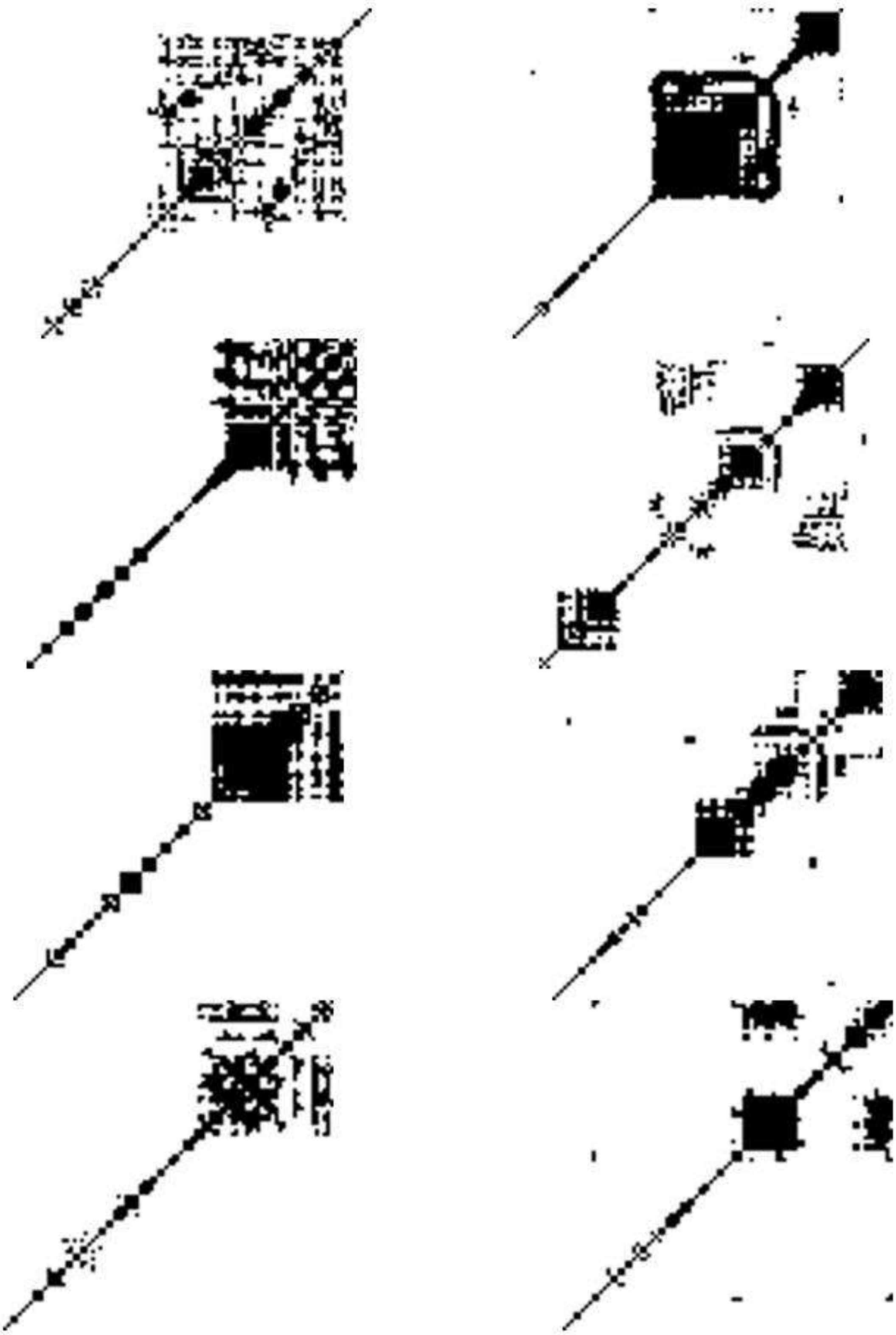

Figure 7: Recurrence Plots of Class "0" (Left) and Class "1" (Right)

3.3. Classification quality metrics

To determine the classification accuracy, metrics were used that are determined by the number of correctly and falsely detected cases presented in the confusion matrix, namely: true positive (*TP*) - when the ECG of a healthy person was correctly identified; true negative (*TN*) - when the disease was correctly recognized; false positive (*FP*) - when the ECG was healthy, but was classified as a disease; and false negative (*FN*) - when the disease ECG was taken for the healthy ECG. The classification metrics are calculated as a function of these four values.

Accuracy is the proportion of correctly defined ECGs for healthy and diseased person:

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \qquad (1)$$

The Precision metric can be interpreted as the proportion of objects called positive by the classifier and, at the same time, are really positive, and the Recall metric shows what proportion of objects of a positive class from all objects of a positive class was found by the algorithm:

$$Precision = \frac{TP}{TP + FP} \;, \qquad (2)$$

$$Recall = \frac{TP}{TP + FN} \qquad (3)$$

F -metric is the harmonic mean of *Precision* and *Recall*:

$$F = 2\frac{Precision * Recall}{Precision + Recall} \,. \qquad (4)$$

The ROC curve was also plotted. The Area Under Curve - Receiver Operating Characteristic curve (AUC-ROC) is a way to evaluate the model as a whole. The ROC curve is a curve from point (0,0) to point (1,1) in the coordinates True Positive Rate (TPR) and False Positive Rate (FPR), where

$$TPR = \frac{TP}{TP + FN} \,, \qquad (5)$$

$$FPR = \frac{FP}{FP + TN} \,. \qquad (6)$$

Ideally, when the classifier makes no mistakes ($FPR = 0, TPR = 1$), the area under the curve is equal to 1; when the classifier determines the probabilities of the classes at random, the AUC-ROC will approach 0.5, since the classifier will issue the same number of *TP* and *FP*; it is obvious that the value of the area under the curve evaluates the quality of the algorithm.

4. Research results and discussion

Consider the results of the classification carried out by the two methods described above and compare them using classification quality metrics.

The results of the classification on the basis of numerical recurrence characteristics are presented in Table 3. It is clear that the ECG time series related to class “1”, i.e. electrocardiograms of patients with ischemic disease are recognized much more accurately than normal ECG records.

Table 3
Classification Evaluation Metrics

	Precision	Recall	F -metric
Class 0	0.80	0.64	0.71
Class 1	0.75	0.94	0.83
Accuracy			0.81

ROC-curve is a reliable method for assessing accuracy. In fig. 8 the ROC-curve for classification based on quantitative characteristics is presented, the value of the area under the ROC-curve is 0.76.

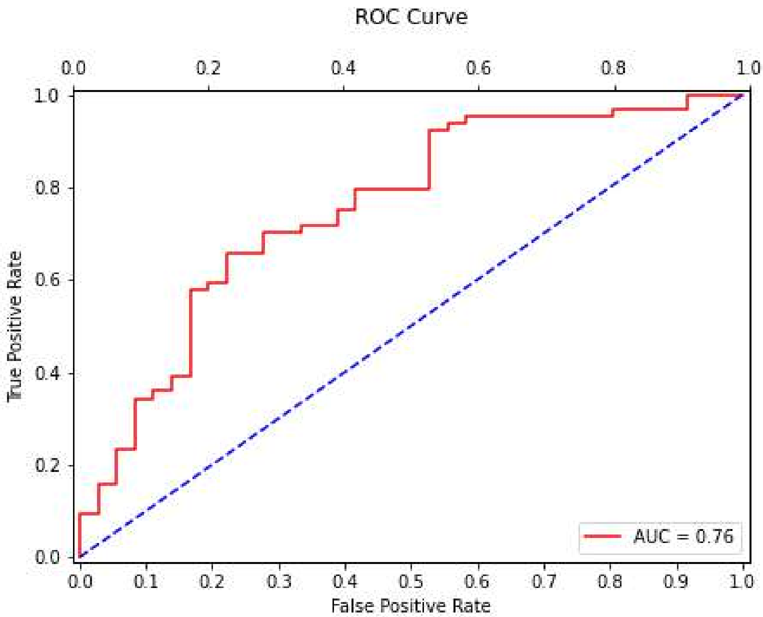

Figure 8: ROC Curve for Classification Based on Quantitative Characteristics

The evaluation metrics for classification of recurrence plot images using developed convolutional neural network were calculated and presented in Table 4.

Table 4
Classification Evaluation Metrics

	Precision	Recall	F -metric
Class 0	0.93	0.72	0.81
Class 1	0.86	0.97	0.91
Accuracy			0.89

From the obtained values of the metrics it follows that as well as in the first case the ECG of patients with ischemia is determined more accurately than the ECG of patients without heart disease. Perhaps this is due to the greater number of realizations in the sampled data or some characteristic features of the ECG.

In fig. 9 the ROC curve is presented, the value of the area under the ROC-curve is 0.92.

The results showed that the classification of recurrence plot images using a convolutional neural network gave significantly higher accuracy for all metrics than classification based on quantitative characteristics using a fully connected multilayer perceptron.

It should be noted that in the dataset description it was indicated that the best classification accuracy of these data was obtained using the Bag-of-SFA-Symbols (BOSS) classifier and equals 89% [16]. Although we have achieved the same precision, we used the simple neural network. Obviously, when using a deep neural network aimed at recognizing black and white images, the classification accuracy will be higher.

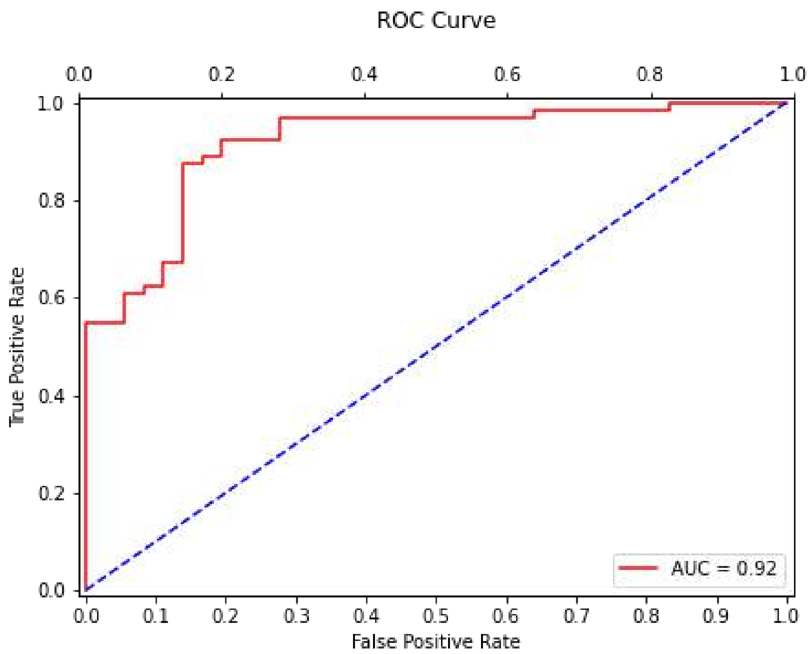

Figure 2: ROC Curve for Classification of Recurrence Plot Images

5. Conclusion

The article discussed comparative analysis of the time series classification based on the application of recurrence plot method. Two approaches were applied for classification: the use of quantitative recurrence characteristics as features and the recognition of recurrence plot images

The input data for the experiment were ECG time series containing 100 values, which have been divided into two classes: "normal" and "ischemia". Research results have shown the advantages of classifying images of recurrence plots. With this approach the classification accuracy has been 89%, while the accuracy of classification based on numerical characteristics has been 81%. Image classification have been carried out using a simple convolutional network, however, the accuracy value was equal to the best accuracy obtained by classifying this dataset using was equal methods.

The considered approach of image recognition has great potential for other applications related to the analysis and classification of time series. Our future research will focus on improving the neural network architecture in order to better recognize black and white images of typical recurrence plots.

6. References

[1] J. C. B. Gamboa, Deep learning for time-series analysis, 2017. URL: https://arxiv.org/pdf/1701.01887.pdf.

[2] H. Ismail Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P. A. Muller, Deep learning for time series classification: a review. Data Mining and Knowledge Discovery 33.4 (2019): 917-963. doi: 10.1007/s10618-019-00619-1.

[3] Marisa Faraggi, Time series features extraction using Fourier and Wavelet transforms on ECG data. URL: https://slacker.ro/2019/11/23/time-series-features-extraction-using-fourier-and-wavelet-transforms-on-ecg-dat.

[4] L. Kirichenko, T. Radivilova, V. Bulakh. Binary Classification of Fractal Time Series by Machine Learning Methods, in: V. Lytvynenko, S. Babichev, W. Wójcik, O. Vynokurova, S. Vyshemyrskaya, S. Radetskaya (Eds.), Lecture Notes in Computational Intelligence and Decision Making, volume 1020 of Advances in Intelligent Systems and Computing, Springer, Cham, 2020, pp. 701-711. doi: 10.1007/978-3-030-26474-1_49

[5] T. Radivilova, L. Kirichenko, V. Bulakh, Comparative analysis of machine learning classification of time series with fractal properties, in: Proceedings of 8th International Conference on Advanced Optoelectronics and Lasers, CAOL 2019, IEEE, Sozopol, Bulgaria, 2019, pp. 557-560. doi: 10.1109/CAOL46282.2019.9019416

[6] L. Kirichenko, P. Zinchenko, T. Radivilova, M. Tavalbeh, Machine Learning Detection of DDoS Attacks Based on Visualization of Recurrence Plots, in: Proceedings of the International Workshop of Conflict Management in Global Information Networks, CMiGIN 2019, Ceur, Kyiv, Ukraine, 2019, pp. 23–34.

[7] L. Kirichenko, P. Zinchenko, T. Radivilova, Classification of Time Realizations Using Machine Learning Recognition of Recurrence Plots, in: S. Babichev, V. Lytvynenko, W. Wójcik, S. Vyshemyrskaya (Eds.), Lecture Notes in Computational Intelligence and Decision Making, volume 1246, of Advances in Intelligent Systems and Computing, Springer, Cham, 2021, pp. 687-696. doi: 10.1007/978-3-030-54215-3_44.

[8] N. Hatami, Y Gavet, J. Debayle, Classification of time-series images using deep convolutional neural networks, in: Proceedings of Tenth International Conference on Machine Vision, ICMV 2017, 10696, 106960Y, 2018.

[9] J. P. Eckmann, S. O. Kamphorst, D. Ruelle, Recurrence plots of dynamical systems. Europhysics Letters 4.9, (1987): 973-977.

[10] N. Marwan, N. Wessel, U. Meyerfeldt, A. Schirdewan, J. Kurths, Recurrence-plots-based measures of complexity and application to heart-rate-variability data. Physical Review E 66.2 (2002): 026702-1-026702-6. doi: 10.1103/PhysRevE.66.026702

[11] N. Marwan, M. C. Romano, M. Thiel, J. Kurths, Recurrence plots for the analysis of complex systems. Physics reports 438.5-6 (2007): 237-329.

[12] L. Kirichenko, T. Radivilova, V. Bulakh, Classification of fractal time series using recurrence plots, in: Proceedings of 2018 International Scientific-Practical Conference Problems of Infocommunications. Science and Technology, PIC S&T 2018, IEEE, Kharkiv, Ukraine, 2018, pp.719-724. doi: 10.1109/INFOCOMMST.2018.8632010.

[13] L. Kirichenko, T. Radivilova, V. Bulakh, P. Zinchenko and A. Saif Alghawli, Two Approaches to Machine Learning Classification of Time Series Based on Recurrence Plots, 2020 IEEE Third International Conference on Data Stream Mining & Processing (DSMP), Lviv, Ukraine, 2020, pp. 84-89, doi: 10.1109/DSMP47368.2020.9204021.

[14] Y. LeCun and Y. Bengio, Convolutional Networks for Images, Speech, and Time-Series, in M. A. Arbib (Eds.), The Handbook of Brain Theory and Neural Networks, MIT Press, 1995.

[15] C. Dan, U. Meier, J. Masci, L. M. Gambardella, J. Schmidhuber, Flexible, High Performance Convolutional Neural Networks for Image Classification, in: Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence, volume 2, 2013, pp.1237–1242. URL: http://people.idsia.ch/~juergen/ijcai2011.pdf.

[16] Time series classification. URL: http://www.timeseriesclassification.com

[17] D. Cielen, A. Meysman, M. Ali, Introducing Data Science: Big Data, Machine Learning, and more, using Python tools, Manning Publications, 2016.

[18] J. Brownlee, A Gentle Introduction to the Rectified Linear Unit (ReLU), Machine learning mastery, January 2019. URL: https://machinelearningmastery.com/rectified-linear-activation-function-for-deep-learning-neural-networks

[19] S. Ioffe and C. Szegedy, Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift, in: Proceedings of the 32nd International Conference on Machine Learning, volume 37, 2015, pp. 448-456.

[20] D. P. Kingma and J. Ba Adam, A Method for Stochastic Optimization, in: Proceedings of the 3rd International Conference on Learning Representations (ICLR), 2015.