EDUCATION AND SCIENCE MINISTRY OF UKRAINE
NATIONAL AVIATION UNIVERSITY
DEPARTMENT OF COMPUTER INTEGRATED COMPLEXES

ADMIT TO DEFENSE
Head of department
Viktor M. Sineglazov
"____" ______________ 2020

MASTER'S THESIS

(EXPLANATORY NOTE)

GRADUATE OF EDUCATION AND QUALIFICATION LEVEL "MASTER"

THEME: IMAGE PROCESSING SYSTEM USING FUZZY NEURAL NETWORK

Executor: **Tofaniuk O. R.**

Supervisor **Professor Sineglazov V.M.**

Advisor on environmental protection: **Frolov V.F.**

Advisor on labor protection: **Konovalova O. V.**

Norms inspector: **Ph.D., Associate Professor Tupitsyn M.F.**

KYIV 2020

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ
КАФЕДРА КОМП'ЮТЕРНО-ІНТЕГРОВАНИХ КОМПЛЕКСІВ

ДОПУСТИТИ ДО ЗАХИСТУ
Завідувач кафедри
В.М. Синєглазов
"_____"____________2020 р.

ДИПЛОМНА РОБОТА

(ПОЯСНЮВАЛЬНА ЗАПИСКА)

ВИПУСКНИКА ОСВІТНЬО-КВАЛІФІКАЦІЙНОГО РІВНЯ "МАГІСТР"

Тема: СИСТЕМА ОБРОБКИ ЗОБРАЖЕНЬ ЗА ДОПОМОГОЮ НЕЧІТКИХ НЕЙРОННИХ МЕРЕЖ

Виконавець: **Тофанюк О. Р.**

Керівник: **д.т.н., професор, Синєглазов В. М.**

Консультант з екологічної безпеки: **Фролов В. Ф.**

Консультант з охорони праці: **Коновалова О. В.**

Нормоконтролер: **к.т.н, доцент, Тупіцин М. Ф.**

Київ 2020

NATIONAL AVIATION UNIVERSITY

Faculty of aeronavigation, electronics and telecommunications

Department of Aviation Computer Integrated Complexes

Education level master

Field of study: 15 “Automation and Instrumentation"

APPROVED BY

Head of department

Victor M. Sineglazov

“_____”______________ 2020

Graduate Student’s Diploma Thesis Assignment

Student’s name: Oleksii Tofaniuk

1. The thesis title: “Image processing system using fuzzy neural networks”.

2. The thesis to be completed between: from 01.09.2020 to 18.12.2020

3. Background to the project (work): To focus on the most famous algorithms of neural fuzzy logic.

4. The content of the explanatory note (the list of problems to be considered):

1. Analysis of issues and relevance. 2. Study of the ecological state of the environment. 3. General methods of image processing. 4. Review of the most well-known topologies of fuzzy neural networks 5. Hybridization of neural networks. 6. Algorithmic application of fuzzy neural network for classification problems. 7. The structure of the developed system. 8. Development of system software.

5. List of required graphics:

1. Structural diagrams of existing neural networks; 2. Structural diagrams of existing neural fuzzy network; 3. Structural diagrams of developed neural fuzzy network.

6. Planned schedule:

№	Task	Term of execution	Performance mark
1	General methods of image processing	10.10-20.10.2020	**Done**
2	Review of topologies of fuzzy neural networks	20.10-01.11.2020	**Done**
3	Algorithmic application of fuzzy neural network for classification problems	01.11-30.11.2020	**Done**
4	Forming conclusions about the work performed	01.12-18.12.2020	**Done**

7. Special chapters' advisors

Chapter	Advisor (position, name)	Date, signature	
		Assignment issue date	Assignment accepted
Labor protection	Konovalova O. V.		
Environmental protection	Frolov V.F.		

8. Date of task receiving: ______________________________

Diploma thesis supervisor ______________ Viktor M. Sineglazov
(signature)

Issued task accepted ______________ Oleksii R. Tofaniuk
(signature)

НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

Факультет аеронавігації, електроніки та телекомунікацій

Кафедра авіаційних комп’ютерно - інтегрованих комплексів

Освітній ступінь магістр

Напрям 15 – Автоматизація та приладобудування

ЗАТВЕРДЖУЮ

Завідувач кафедри

Синєглазов В.М.

“ ____ ” __________2020 р.

ЗАВДАННЯ
на виконання дипломної роботи студента
Тофанюка Олексія Руслановича

1. Тема роботи: «Система обробки зображень за допомогою нечітких нейронних мереж»

2. Термін виконання роботи: з 01.09.2020р. до 18.12.2020.

3. Передумови проекту (роботи): Зосередити увагу на найвідоміших алгоритмах нейронної нечіткої логіки.

4. Зміст пояснювальної записки (перелік питань, що підлягають розробці):

1. Аналіз питань та актуальність. 2. Вивчення екологічного стану навколишнього середовища. 3. Загальні методи обробки зображень. 4. Огляд найбільш відомих топологій нечітких нейронних мереж 5. Гібридизація нейронних мереж. 6. Алгоритмічне застосування нечіткої нейронної мережі для задач класифікації. 7. Структура розробленої системи. 8. Розробка системного програмного забезпечення.

5. Перелік обов’язкового графічного матеріалу:

1. Структурні діаграми існуючих нейронних мереж; 2. Структурні діаграми існуючої нейронної нечіткої мережі; 3. Структурні діаграми розвиненої нечіткої нейронної мережі.

6. Календарний план-графік

№	Задачі	Кінцевий термін	Стан
1	Загальні методи обробки зображень	10.10-20.10.2020	**Виконано**
2	Огляд топологій нечітких нейронних мереж	20.10-01.11.2020	**Виконано**
3	Алгоритмічне застосування нечіткої нейронної мережі для задач класифікації	01.11-30.11.2020	**Виконано**
4	Формування висновків про виконану роботу	01.12-18.12.2020	**Виконано**

7. Консультанти зі спеціальних розділів

Розділ	Консультант (посада, П. І. Б.)	Дата, підпис	
		Завдання видав	Завдання прийняв
Охорона праці	Коновалова О.В,		
Охорона навколишнього середовища	Фролов В.Ф.		

8. Дата видачі завдання ______________________________

Керівник: д.т.н., професор ______________________ Синєглазов В.М
(підпис)

Завдання прийняв до виконання ______________________ Тофанюк О. Р.
(підпис)

ABSTRACT

Explanatory note to the thesis "Image processing system using fuzzy neural networks" contains pages - 110, sections - 6, figures - 30, tables - 2, used sources - 17.

Keywords - neural network; fuzzy network; fasification; dephasification; segmentation; filtration.

The purpose of scientific work: development of a fuzzy neural network for image processing.

Thesis deals with the theoretical and software part of the development of fuzzy neural network for medical image processing and classification. The author substantiates the relevance of using fuzzy neural networks to solve the classification problem, analyzes the existing topologies of fuzzy neural networks and fuzzy classifiers, the main algorithms for improving the results, identified their shortcomings and proposed a solution to eliminate them.

To solve the classification problem, it is proposed to use the fuzzy neural network NEFCLASS. This software architecture allows you to create a neural classifier that can compete with the results of existing solutions using other types and topologies of neural networks. And expands the range of calculations performed to classify the input data.

РЕФЕРАТ

Пояснювальна записка до дипломної роботи «Система обробки зображень за допомогою нечітких нейронних мереж» містить сторінок - 110, розділів - 6, рисунків - 30, таблиць – 2, використаних джерел - 17.

Ключові слова – нейронна мережа; нечітка мережа; фазифікація; дефазифікація; сегментація; фільтрація.

Мета наукової роботи: розробка нечіткої нейронної мережі для обробки зображень.

В дипломній роботі розглядається теоретична та програмна частина розробки нечіткої нейронної мережі для обробки медичних зображень та класифікації. Автором обґрунтовано актуальність використання нечітких нейронних мереж для вирішення задачі класифікації, проведено аналіз існуючих топологій нечітких нейронних мереж та нечітких класифікаторів, основних алгоритмів для покращення результатів, виявлено їх недоліки та запропоноване рішення, що дозволяє їх усунути.

Для вирішення задачі класифікації запропоновано використати нечітку нейронну мережу NEFCLASS. Дана програмна архітектура дозволяє створити нейронний класифікатор, який може мати конкурентну спроможність з результатами уже існуючих рішень за допомогою інших типів і топологій нейронних мереж. Та розширює спектр виконуваних обчислювань для класифікації вхідних даних.

CONTENT

Glossary

CT - Computed tomography

US – Ultrasound

ADD - Analog-Digital Device

MRI - Magnetic resonance imaging

RND - radionuclide (research)

RFP – radiopharmaceutical (research)

ANN - Artificial neural network

SVM - Support Vector Machine

FNN - Fuzzy Neural Network

TSK - Takagi-Sugeno-Kanga (network)

ANFIS - Adaptive Network Based Fuzzy Inference System

CNN - Convolutional Neural Network

RBFNN - Radial Basis Functions Neural Network

AF - Activation Function

INTRODUCTION

Today it is difficult not to notice the rapid development of technology. They penetrate into every sphere of human life, thus creating more comfortable conditions. However, one of the most discussed and interesting topics today is neural networks. It is difficult not to appreciate their contribution to the existence of each person. They are able to solve a wide range of problems, from forecasting to decision-making using artificial intelligence and automatic control, such as autopilot in cars.

It is also worth noting the great contribution of artificial neural networks in the development of modern medicine. It is thanks to neural networks, doctors and medical scientists that it has become much easier to study, investigate various drugs, organs, tissues, cells, and diagnose patients with various diseases. Since most studies require different types of visual information, whether cells in a microscope or an X-ray, they can all be presented as an image that can be digitized and processed for more convenient and detailed study. For these tasks, artificial neural networks are used, which are able to process medical images and classify objects on them. The most popular neural networks used for image processing tasks are convolutional neural networks.

The relevance

Today's predictions on data coming from medical systems have become realistic, it requires better and better algorithms and computing power. That is why researchers who are at the forefront of scientific positions, turn to computer systems to obtain the maximum possible computing power. An analysis of the complexity of current computational problems in various fields of science and technology shows that computers with very high power are needed to solve them.

This paper will investigate the influence of neural networks on the processing of medical images and the classification of objects in these images. Fuzzy neural networks will be used as classifiers. The principle of fuzzy neural networks is slightly different from convolutional neural networks, so the aim of this work is to study the effectiveness of fuzzy neural networks as classifiers of objects in medical images, as well as the effectiveness of hybrid neural networks in solving the classification problem.

The purpose and objectives of the thesis is to develop topology of the fuzzy neural network to solve the problem of classification of images on a set of images of computed tomography of the human lung, which are presented in the form of a data sample. Based on this sample, it is necessary to teach fuzzy neural network to recognize artifacts in the lungs, tumors, malignancies, inflammation, etc. The paper will present several types of neural network topologies, consider the topology of a hybrid neural network, evaluate the advantages and disadvantages of each.

The object of study for one of the selected topologies it is necessary to consider the methods of solving the classification problem, to build an algorithm for structural-parametric synthesis, to evaluate the problems of constructing a training sample for a fuzzy neural network.

The subject of research there is a neuro fuzzy classifier for solving problem image processing.

CHAPTER 1. IMAGE PROCESSING IN MEDICAL DIAGNOSTIC SYSTEMS

1.1. World experience of automated image processing. Intelligent approach.

Today, the rapid development of information technology has a significant impact on all spheres of human life. An important place was occupied by the development of automated image processing systems, which are increasingly beginning to be used in human life.

The principle of image processing is any form of information processing for which the input data is represented by images, such as photographs or video frames. Image processing can be performed both to obtain an output image (eg, preparation for printing, to telecast, etc.) and to obtain other information (eg, text recognition, counting the number and type of cells in the microscope field, etc. e.). In addition to static 2D images, you also need to process images that change over time, such as video.

In the middle of the last century, image processing was performed by optical devices. Although such optical methods remain important in areas such as holography, due to the sharp increase in computer performance, these methods have been increasingly supplanted by digital image processing. Digital image processing methods are usually more accurate, reliable, flexible and easy to implement than analog methods. Specialized equipment, such as processor processors and multiprocessor systems, is widely used in digital image processing. This is especially true for video processing systems. Image processing is also performed using computer mathematics software, such as MATLAB, Mathcad, Maple, Mathematica, and others. To do this, they use both basic tools and extension packages Image Processing.

Automated image processing systems are a set of computer program instructions that automate the image processing process itself. It is enough for a person to understand how to use the program, and the process of image processing is automatic according to the instructions.

Automated image processing systems are used for applied and scientific purposes and solve the following typical problems:

• Text recognition

• Processing of satellite images

• Machine vision

• Data processing to highlight different characteristics

• Image processing in medicine

• Identification of the person (on the face, iris, dactyloscopic data)

• Automatic car control

• Determining the shape of the object

• Determining the movement of the object

• Applying filters

For example, text recognition is widely used to convert books and documents into electronic form, to automate accounting systems in business or to publish text on a web page. The text recognition process itself is the electronic translation of images with handwritten, typewritten, or printed text into an electronic format that can be viewed in a text editor. Text recognition allows you to edit text, search for a word or phrase, store it in a more compact form, display or print material without losing quality, analyze information, and apply electronic translation, formatting, or speech conversion to text.

The task of processing medical images has also made a significant breakthrough. Thanks to the use of neural networks, doctors have become able to more accurately diagnose patients. After all, in addition to the process of image processing, neural networks are able to classify objects in the picture, which in turn makes them a kind of assistants in decision-making.

1.2. Hardware for obtaining medical images.

Medical imaging is one of the important means of obtaining visual information about the internal structures and functions of the human body. It can be obtained by radiation (radiological) or non-radiological methods.

Images taken with a video camera (endoscopy) or photographed (microscopic images in histology, pathology, dermatological images, etc.) are obtained by non-radiological methods. These types of images can also be digitized and subsequently processed.

Radiation (radiological) methods make available for visual perception information that is not directly perceived by sight. Such information (images of organs or parts of organs) is obtained using a variety of physical radiation and fields. Sound waves (mainly ultrasound), electromagnetic radiation of different ranges, constant and alternating electromagnetic field, gamma radiation of a radiopharmaceutical can be used. Medical imaging obtained by radiological diagnostics is the main source of information in the field of health care. All of these methods use computational procedures to obtain images.

Radiation diagnostic methods include:

- radiology,
- computed tomography,
- ultrasound,
- magnetic resonance imaging,
- radionuclide method, thermography.

Radiology uses ionizing radiation from an X-ray source. There are 2 main methods in the radiological method: radiography, radioscopy. Currently, X-ray images are obtained using a wide range of different methods using direct analog, indirect analog and digital technologies.

Computed tomography (CT) is a method of layer-by-layer X-ray examination of organs and tissues using computer processing of multiple X-ray images made at different angles, followed by image reconstruction and determination of the density of any area of these tissues. The method was proposed in 1972 by Godfrey Hounsfield and Allan Cormack. The main and fundamental difference between the image in computed tomography from the usual X-ray image is that it appears as a result of accurate measurements and calculations that relate to the selected layer. Therefore, images in X-ray computed tomography have ten times higher than in traditional X-rays, the resolution of tissue density, which allows good differentiation of soft tissues, to separate images of structures that overlap, and accurately identify areas of pathological changes.

Ultrasound (US) is one of the main methods of medical imaging. This uses ultrasonic waves, and their ability to reflect from the boundaries of media that differ in density. The ultrasound method is based on the echolocation of deep tissues of the body, namely on the study of the probing pulse of ultrasound and the reception of signals reflected from the interface of tissue media that have different acoustic properties. The greater the difference in the impedance of the media adjacent to each other, the greater the signal amplitude. The reflected ultrasonic waves are captured by the sensor. After amplification and conversion into electrical signals, the information is digitized using an ADD (analog-digital device) and fed to a computer. With the help of software, the information is processed and a two-dimensional image of the tissues through which the ultrasonic waves have passed is displayed on the screen.

Magnetic resonance imaging (MRI) is the most important clinical method for diagnosing many human diseases. The method can detect tumors of any location, most diseases of the brain and spinal cord, heart, musculoskeletal system, etc. With the help of MRI, you can examine the vessels without the use of contrast agents. Magnetic resonance imaging is a radiological method of research based, like CT, on obtaining layered images. However, it is not based on X-rays, but on nuclear magnetic resonance - a physical phenomenon of interaction of external magnetic fields with the protons of the nuclei of the substance under study. The nuclei of certain elements have the ability under the action of an external electromagnetic field to absorb energy and then give it away in the form of a radio signal (magnetic resonance). With the help of a computer, information on the distribution of dipoles of matter and the amount of energy released in a given plane of the object is collected and processed. The display shows a tomographic image, which characterizes not only the physical but also the chemical properties of tissues.

The method of radionuclide research (RND) is based on the measurement of gamma radiation of a radiopharmaceutical (RFP), which is injected into the body for diagnostic purposes. This drug is selectively retained by various organs and tissues, contains a radionuclide that decays with the emission of certain quanta. Areas of increased accumulation of RFP (hot foci) are found in foci of inflammation, hyperplasia, some tumors and metastases. Areas of reduced accumulation of RFP (cold foci) reflect the loss of functional activity of the tissue in the area of the tumor, cyst, connective tissue growth, decreased blood flow.

As mentioned above, modern methods of medical diagnostics and biomedical research are largely based on the analysis of images obtained using technical means (light and electron microscopes, X-ray and thermographic devices, tomographs, etc.). However, the solution of diagnostic and scientific problems when working with visual information requires knowledge of specific methods of formation, registration, digital processing and image analysis. This is especially evident when using new types of information systems that solve the problem of extracting hidden diagnostic information (computer tomographs, laser confocal microscopes, ultrasonic diagnostic devices, etc.). More than 90% of information about the world around him a person receives through sight. However, for all the sensitivity of the eye, its capabilities are limited by the ability to perceive electromagnetic radiation only in the visible range, and man has long sought to expand them. Not surprisingly, the improvement of methods and means of collecting and processing visual information is an extremely important area of scientific and technological progress. Modern computer means of image formation and processing can cover almost the entire electromagnetic spectrum from gamma radiation to radio waves. Images can be generated by sources with which it is unusual for a person to associate the observed images (for example, ultrasound images; images in the radio range, etc.).

Medical imaging is an important tool in many areas of medical use, used for both diagnosis and treatment. However, reading medical images and diagnosing or recommending treatment requires specially trained medical professionals. Modern practice of reading medical images is time consuming, costly and error prone. It would be preferable to have an automated system that can automatically provide recommendations for diagnosis and treatment.

1.3. Problem statement of the image processing. Features of medical image processing.

Objects of interest of the researcher on medical images used in early diagnosis are usually small and low-contrast in comparison with the surrounding background. Visual detection of these objects - the first step in the diagnostic path for a medical image - can cause problems. On the one hand, they are due to these features of the images themselves, on the other - the limited characteristics of the visual system of the researcher and the distortions that occur in the images when they are received and displayed. The study of the functions and features of modern specialized systems for analysis and processing of medical images for various purposes has shown that these systems have a number of disadvantages. The main disadvantage is that most of the systems contain only a wide range of methods of analysis and image processing, available to the researcher, without specifying which method should be used to achieve the goal of transformation. In this regard, the following problems have been identified: it is impossible to guarantee the optimal (in the sense of achieving the goal of transformation) choice of method (or combination of methods) for image processing, as this choice is based only on user knowledge and experience; it is impossible to search all the methods available to the researcher (and their combinations) to achieve the best processing result, as it will be too time consuming. Therefore, to improve the operation of systems for analysis and processing of medical images, obviously, you need a method that provides automated selection of image conversion.

When processing and analyzing images, the following main stages are distinguished: filtering; pre-treatment; segmentation; recognition; diagnostics. The efficiency of the subsequent stages of image processing directly depends on the results of filtering and pre-processing.

The filtration step is necessary to reduce various obstacles. There are different types of filters: low frequency, high frequency, median, adaptive and other types of digital filters. Linear filtering is widely used in digital image processing. It is based on the use of fast convolution algorithms. Static filter masks do not always guarantee an acceptable result, because linear filters lead to the smoothing of brightness differences, which in turn complicates the task of highlighting borders. Nonlinear filtering has a number of advantages over linear: it distorts the brightness differences less, which makes it possible to find the boundaries of objects more accurately and removes impulse interference. Another type of filtering is adaptive filtering. This type of filtering has a number of advantages, such as: local filtering, changing the local filter mask and the size of the filter aperture.

The methods used at the pre-treatment stage depend on the objectives of the research, and they are quite diverse. They may include highlighting the most informative fragments, enlarging them, color mapping, changing the spatial resolution, changing the contrast resolution, etc. One of the main actions performed at the pre-processing stage is to change the contrast and brightness of the image. Methods of changing contrast and brightness are divided into linear, nonlinear and adaptive. When applying the appropriate masks, you can combine the two stages (filtration step and pre-treatment step) to ensure speed. The pre-processing stage also involves geometric operations on the image. These include methods of image rotation, image enlargement and reduction.

1.4. Review of medical image processing methods. Artificial neural networks.

The design of integrated medical systems based on conceptual specifications involves the use of graphical databases of medical images. Image processing and analysis is a step-by-step procedure that depends on the results of the previous stage, as well as the knowledge and experience of the operator. The pre-processing phase improves the image quality, and the segmentation phase highlights the elements, its composition, which ultimately improves the quality and accuracy of diagnosis.

Methods and phases of medical image processing in a graphic database. The pre-processing phase eliminates deviations associated with the image generation system and reduces noise. Methods are used that process digital data with the help of special programs and, thus, improve the visibility of some anatomical structures.

The method of changing the contrast of the image. Histogram analysis reveals the distribution of gray levels in the image and helps to judge the quality of digitization. If the histogram has a nonlinear distribution, many details will be lost. Histogram alignment operations improve the contrast and, consequently, the display of details.

To increase the contrast of the image often use methods that operate with the histogram of the image. The essence of these methods is to convert the brightness of the original image so that the histogram of the brightness distribution took the desired shape.

The simplest is the method of linear scaling, which consists in stretching the histogram of the image to the maximum allowable range. Usually the minimum and maximum limit levels are set, which provides a higher quality of subjective perception of the image, especially if the processed image contains relatively few elements exceeding the level of restriction.

Methods of sharpening. Visually, images with blurring and blurring are perceived as defocusing, or deterioration of image sharpness. Therefore, increasing the sharpness of the image should consist in increasing the level of high frequencies of the image spectrum (in its high-frequency filtering).

The simplest and most effective method is to process a "sliding window" of small size. The sharpening algorithm is implemented as a two-dimensional filter with a finite pulse response.

Noise filtration methods. Due to especially the equipment, as well as errors in the methods of reconstruction in the final image there is a noise component. Possible noise models are additive and pulse.

An effective method of improving image quality is median filtering.

In addition, there are adaptive filters with a finite impulse response, where the impulse response coefficients of the filter vary according to the structure of the processed image.

Segmentation methods. Image segmentation is the process of dividing an image into areas with the same characteristics. This phase of image processing isolates individual elements of the image (organs, cells, etc.). The method is based on the identification of identical pixels with an acceptable level of error. Comparing two time-segmenting images reveals dynamics.

Automatic computer interpretation has not yet been implemented [1]. A database on comparative and pathological anatomy is required for its high-quality performance. The resulting structures and parameters should be classified and compared with known structures.

Segmentation can be performed by a diagnostician manually, based on his experience using special atlases of clinical images. Modern means of digital processing allow to make the process of image segmentation automated, while retaining, however, the right to make the final decision by an expert diagnostician. In recent decades, many approaches have been developed to address the problem of segmentation, most of which operate with binary or monochrome images. Methods of segmentation of such images can be divided into the following groups:

1. Segmentation based on the definition of regional boundaries (contour segmentation),

2. Clustering,

3. The method of growth of areas

4. The method of division-merger of areas.

All segmentation methods can be divided into binary and fuzzy. Binary segmentation implies the exact determination of the affiliation of a pixel in a given area, and fuzzy segmentation assigns only the probability of belonging to a particular structure.

Methods based on defining the boundaries of areas, operate on the digital characteristics of the image, analyzing both the range of local data and two-dimensional vector space, using gradients calculated in this space.

Segmentation is designed to highlight in the images of areas with certain properties. Such areas usually correspond to objects or their parts, which are determined by researchers. The result of segmentation is a binary or hierarchical (multiphase) image, in which each level (phase) of the image corresponds to a specific class of selected objects. Segmentation is a difficult point in the processing and analysis of medical images of biological tissue, as it is necessary to identify areas that correspond to different objects or structures on histological specimens: cells, organelles, artifacts, etc. This is due to the high variability of their characteristics, low contrast of the processed images and complex geometric organization of objects. From the mathematical apparatus used to implement segmentation methods, they are divided into three types: threshold segmentation; morphological segmentation; unification (building) of areas.

In many cases, to obtain a more effective result, you can consistently use different segmentation methods. For example, a morphological gradient operation is used to select boundaries, after which threshold segmentation is performed for fragments corresponding to small brightness differences. In order to identify the features of the object under study, compare them with data from the library and draw a conclusion about the probability of an anomaly, it is necessary to pre-select the object from the set present in a particular image. In most cases, the studied image contains noise, distortion, textural areas similar to the areas belonging to the object under study. All this complicates the process of selection of objects and the correct display of their boundaries, so the algorithms of contouring and segmentation play a very important role in the process of automated processing.

The Sobel operator is one of the best algorithms for delimiting borders, it is often used as one of the stages of more complex and accurate algorithms, such as the Cannes operator. The Sobel operator is used in the field of image processing. It is often used in border allocation algorithms. This is a discrete differential operator that calculates the approximate value of the image gradient. The result of applying the Sobel operator at each point of the image is either the brightness gradient vector at this point, or its norm. In other words, the operator calculates the gradient of the brightness of the image at each point. So is the direction of the greatest increase in brightness and the magnitude of its change in this direction. The result shows how “sharply” or “smoothly” changes the brightness of the image at each point, and hence the probability of finding a point on the border, as well as the orientation of the border. In practice, the calculation of the magnitude of the change in brightness (probability of belonging to the limit) is more reliable and easier to interpret than the calculation of the direction. Mathematically, the gradient of the function of two variables for each point of the image (which is the function of brightness) – a two-dimensional vector, the components of which are derivatives of the brightness of the image horizontally and vertically. At each point of the image, the gradient vector is oriented in the direction of the greatest increase in brightness, and its length corresponds to the magnitude of the change in brightness. The process of segmentation using the Sobel operator is based on a simple movement of the filter mask from point to point of the image: at each point (x, y) the filter response is calculated using predefined links. The direction of the gradient vector coincides with the direction of the maximum rate of change of the function f at the point (x, y). An important role in detecting contours is played by the modulus of this vector, which is denoted by ∇f and is equal to

(1.4.1)

This value is equal to the value of the maximum rate of change of the function f at the point (x, y). The result of using the Sobel operator is a two-dimensional gradient map for each point.

Artificial neural networks

Artificial neural networks (ANN) are a mathematical, software and hardware model built on the principle of functioning of biological neural networks of living organism cells.

Biological neurons transmit chemical signals to each other using electrical impulses. As a result of such activity, a person develops thinking and feeling. Artificial intelligence and cognitive modeling try to mimic some properties of biological neural networks.

ANN is a system of interconnected and interacting artificial neurons (simple processors). Each LV ANN deals only with signals that it periodically receives and signals that it periodically sends to other MVs.

ANN works as follows: the inputs of neurons receive signals that are summed. This takes into account the synaptic weight, ie the significance of each of the inputs. Next, the input signals of some neurons are fed to the inputs of other neurons. The weight of each such bond can be positive (excitatory bonds) or negative (brake bonds). They determine the calculation of a neural network, which means its memory and behavior.

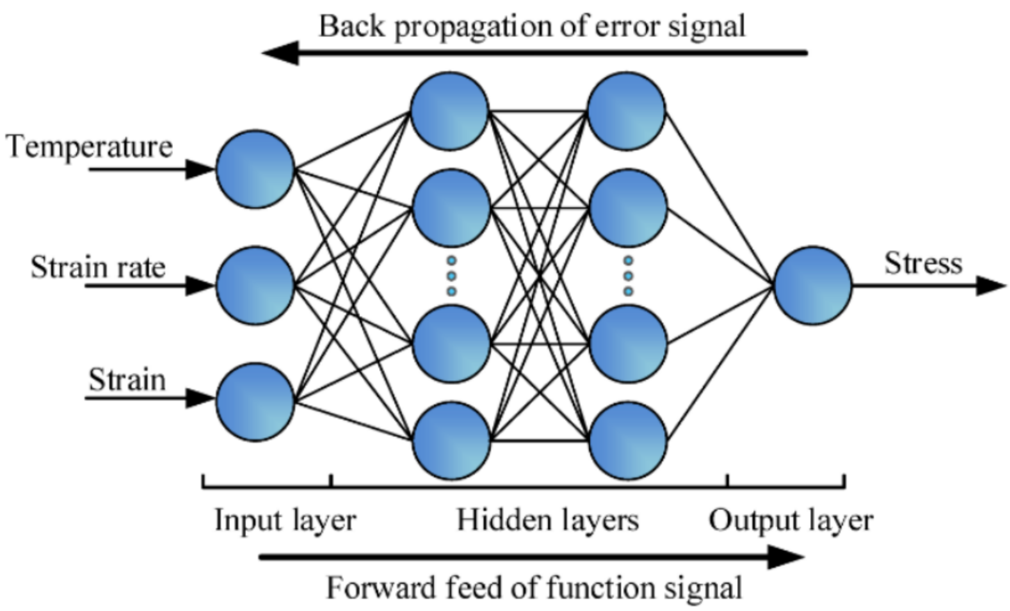


Fig.1.4.1. Simplest structure of multilayer artificial neural networks

In order for ANN to perform complex tasks, it must be trained.

Artificial neural networks are not programmed in the usual sense of the word, they are learned. Learning capability is one of the main advantages of neural networks over traditional algorithms. Technically, learning is about finding the connection coefficients between neurons. In the process of learning, the neural network is able to detect complex relationships between input and output, as well as to make generalizations.

To learn, it is necessary to have a model of the external environment in which the neural network operates - the information needed to solve the problem. Secondly, it is necessary to determine how to modify the weight parameters of the network.

There are three general paradigms of learning: "with a teacher", "without a teacher" (self-learning) and mixed. In the first case, the neural network has the correct answers (network outputs) to each input example. The scales are adjusted so that the network produces the answers closest to the known correct answers. Learning without a teacher does not require knowledge of the correct answers to each example of the training sample. In this case, the internal structure of the data and the correlation between the samples in the training set are revealed, which allows to divide the samples into categories. In blended learning, some weights are determined through teacher training, while others are determined through self-learning.

Artificial neural networks have the following advantages:

1. High reliability. The information in the ANN is encoded and stored not in the individual elements of memory, but in the distribution of connections between neurons and their power, so the state of each individual neuron is determined by the state of many other neurons associated with it. Therefore, the loss of one or more connections does not significantly affect the performance of the system as a whole, which ensures its high reliability.

2. High "natural" noise immunity and functional reliability apply to both distorted (noisy) information flows and failures of individual neurons. This ensures high efficiency and reliability of information processing, and simple additional training and retraining of networks allow when changing external factors to make a timely transition to a new level of tasks.

These advantages of artificial neural networks have given them the opportunity to actively use in the tasks of pattern recognition and classification, in particular in medicine. Objects of different nature can act as images. When teaching the network, various samples of images are offered, indicating to which class they belong. When an image is presented to a network, one of its outputs should indicate that the image belongs to that class. At the same time, other outputs should indicate that the image does not belong to this class.

CHAPTER 2. FUZZY NEURAL NETWORKS IN IMAGE PROCESSING PROBLEMS

2.1. Fuzzy neural networks. Topologies. Parameters.

Interest in neuro intelligences arose even in the early stages of the development of computer technology. It is based on the neural organization of artificial systems, which has biological preconditions. The ability of biological systems to learning, self-organization and adaptation has a greater advantage over modern computing systems. The virtue of computer systems is the high speed of dissemination of information and the ability to account for a large amount of knowledge accumulated by humanity in this area. The development of artificial intelligent systems that combine the benefits of biological creatures and modern computer technology creates the potential prerequisites for the transition to a qualitatively new stage of evolution in computing.

The first in the field of artificial neural networks (1943) were W. McSulloch and W. Pitts. They showed that with the help of threshold neural elements it is possible to realize the calculation of any logical functions. In 1949, D. Hebb proposed a rule that became the mathematical basis for the study of a number of neural networks. In 1959, F. Rosenblatt created a model of the neural network, which he called perceptron. He summarized the results of his research in the book "Principles of Neurodynamics". In 1959, W. Widrow and M. Hoff proposed a training procedure for the ADALINE linear adaptive element called "delta rule." In 1969 M. Minsky and S. Papert published a monograph "Perceptron", in which they carried out a mathematical analysis of the perceptron and noted the inherent limitations. Their conclusions were rather pessimistic, and it played a negative role in the further development of neural networks. Suspended in this area of work was restored in the second half of the 1970's. In 1976, S. Grossberg developed a theory of adaptive resonance that can be used to construct associative memory. G. Anderson (J. Anderson) proposed in 1977 a model of linear associative memory. The research in this direction was continued by T. Kohonen, who created the model of optimal linear associative memory.

The intensity of research in the field of neural networks increased significantly in the 1980's. J. Hopfield (1982) analyzed the stability of neuron networks with feedback and proposed to use them as associative memory. T. Kohonen developed self-organizing neural networks. In 1986, D. Rumelhat, J. Hinton and R. Williams created an algorithm for the propagation of error, which became an effective means for teaching multilayer neural networks. At the same time, in the scientific environment until 2006, the concept prevailed, according to which it makes no sense to use more than two hidden layers in the perceptron. This was due to the fact that the algorithm of back propagation of the error did not give any gain in solving problems when learning the perceptron with more than two hidden layers. Also, this concept was based on the theorem on the universal approximation of a perceptron with one or two hidden layers of any function with arbitrarily given accuracy. In addition, there appeared support vector machine (SVM), which often showed greater efficiency in image recognition operations compared to the perceptron. Therefore, interest in multilayer perceptron was gradually falling. And only after the publication of G. Hinton's work since 2006 began a new stage: there appeared deep neural networks, which are the result of the development of multilayer perceptrons and integrate various paradigms of teaching neural networks.

Neural networks are convenient for image recognition tasks, but very uncomfortable to explain how they implement it. They can automatically acquire knowledge, but the process of their learning is often slow enough, and the analysis of the trained network is very complicated (a trained network is usually a "black screen" for the user). At the same time, it is difficult to enter any prior information (expert knowledge) to accelerate the learning process in the neural network.

Systems with fuzzy logic, on the contrary, are convenient to explain the conclusions obtained with their help, but they cannot automatically acquire knowledge for their use in the mechanisms of outputs. The need to split universal sets into separate areas, as a rule, limits the number of input variables in such systems to a small value.

The direct distribution neural network can approximate any system that is based on fuzzy rules, and any neural network of direct distribution can be approximated by a system based on fuzzy rules.

The neuro-fuzzy network is a representation of a fuzzy output system in the form of a neural network that is convenient for learning, analysis and use. The structure of the neuro-fuzzy network corresponds to the main blocks of fuzzy output systems.

The main properties of neuro-fuzzy networks is that:

- neuro-fuzzy networks are based on fuzzy systems that are learned using methods used in neural networks;
- a neuro-fuzzy network is usually a multi-layer (often triple) neural network. The first layer is the input variables, the average is fuzzy rules, and the third is the output variables. The weight of the connection corresponds to the fuzzy sets of input and output variables. Sometimes a five-layer architecture is used. In the general case, a fuzzy system does not necessarily have to be presented in this form, but it is a convenient model for the application of teaching methods;
- neuro-fuzzy network always (before, during and after training) can be interpreted as a system of fuzzy rules;
- the learning procedure takes into account the semantic properties of the

fuzzy system. This is manifested in the limitation of the possible modifications that apply to the parameters that are being adjusted. It is necessary, however, to say that not all methods of teaching neuro-fuzzy networks take into account the semantics of the system.

Topologies of neuro-fuzzy networks

Let's consider the most popular topologies of neuro-fuzzy networks:

NEFCLASS

The application of Fuzzy Neural Network FNN NEFCLASS is rational by following its properties:

a. NEFCLASS may work with fuzzy and qualitative input information; it enables to attain better classification accuracy than conventional non-fuzzy classifiers;

b. it has accelerated convergence as compared with crisp classification methods.

The system NEFCLASS has 3-layer successive architecture (see fig. 2.1.1). The first layer contains inputs neurons which inputs patterns are fed in. Activating of these neurons does not change usually input values. The hidden layer contains fuzzy rules, and the third layer consists of output neurons (classifiers). Activations of rule neurons and neurons of output layer with the pattern of p are calculated so:

(2.1.1)

(2.1.2)

or alternatively

(2.1.3)

where W(x, r) is a fuzzy weight of connection of input neuron x with a rule neuron R, and W(R, c)-fuzzy weight of connection of a rule neuron R with the neuron C of output layer. Instead of application of operations of maximum and minimum it is possible to use other functions of so-called "t-norm" and "t- co-norm" accordingly [2].

A rule base is approximation of unknown function and describes a classification task"(x), that $c_i = 1$, $c_i = 0$

(j = 1 … N, ∀j ≠ i), if pattern x belongs to the class of C_i .

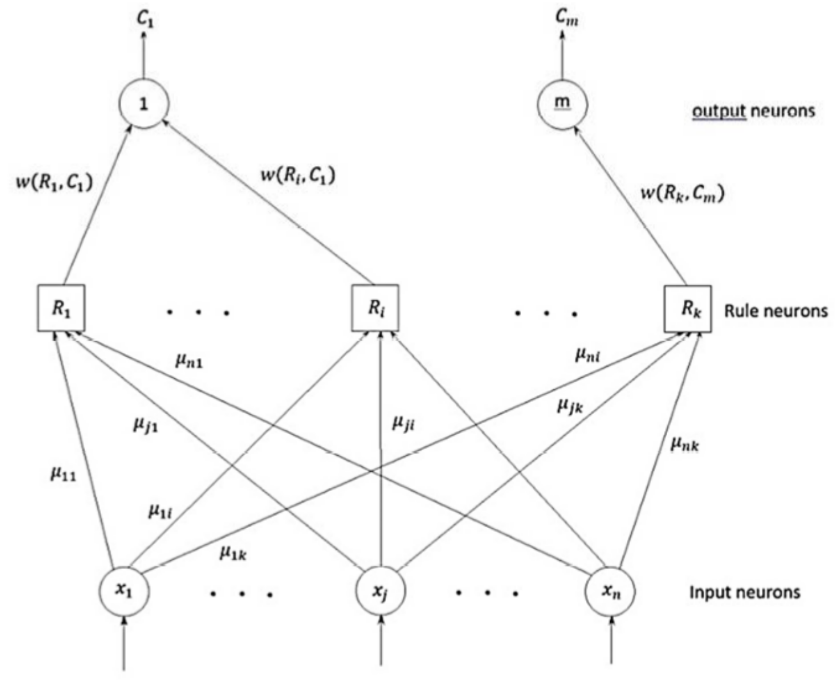


Fig.2.1.1. Structure of FNN NEFCLASS.

Every fuzzy set is marked a linguistic term, such as «large», «small», «middle» et cetera. Fuzzy sets and linguistic rules present approximation of classifying function and determine the result of the system NEFCLASS. They are obtained from a sample by learning. It's necessary, that for every linguistic value (for example, «x_1 is positive and large») there should be only one presentation of fuzzy set.

Takagi-Sugeno-Kanga network

In the Takagi-Sugeno-Kanga network (abbreviated as TSK), the output signal is calculated using the expression

$$y(X)=sum[i=1:M](w_i*y_i(X))/sum[i=1:M](w_i), \qquad (2.1.4)$$

where $y_i(X)=p_{i0}+sum[j=1:N](p_{ij}*x_j)$ is the i-th polynomial component of the approximation.

The weights w_i of the components are calculated by the following formula (using the rational form of the Gaussian function)

$$w_i=prod[j=1:N](w_{ij}(x_j))=prod[j=1:N](1/(1+((x_j-c_{ij})/s_{ij})^{2*b}{}_{ij})). \qquad (2.1.5)$$

The above expressions correspond to a five-layer neural network, the structural diagram of which is presented below.

The first layer contains N * M nodes, each of which implements the calculation of the Gaussian function with the parameters cij, sij and bij. From the point of view of fuzzy systems, this is a layer of fuzzification of input variables. The layer is called parametric, because in the process of training the network, the parameters of this layer are subject to selection. On Fig.2.1.2 shows structure neuro-fuzzy network TSK.

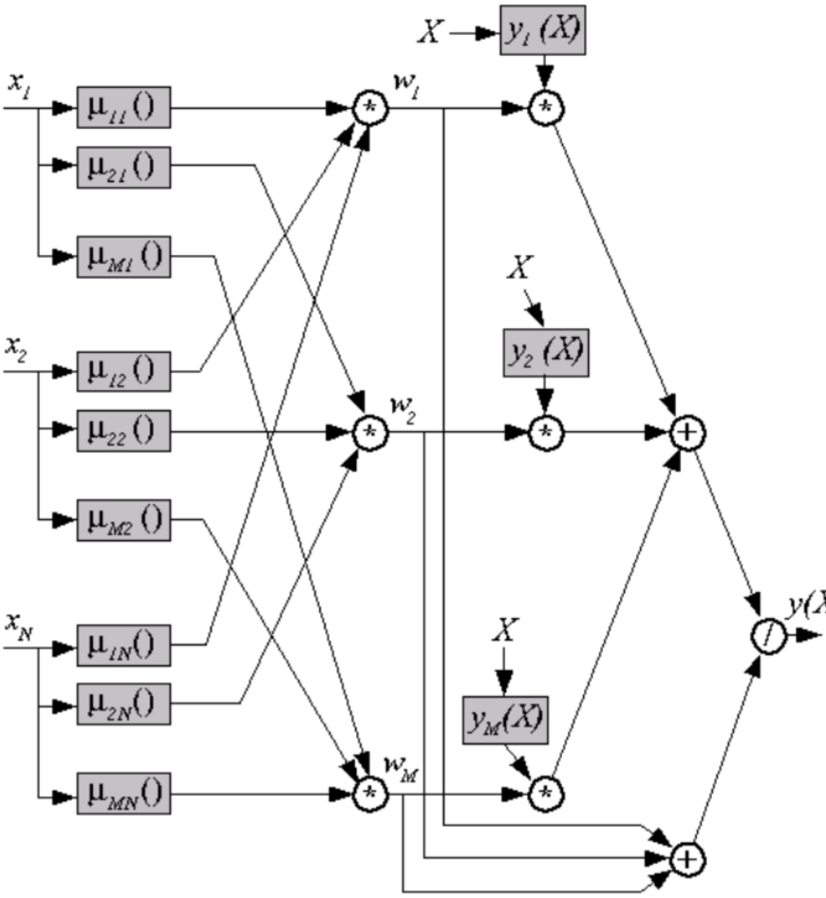


Fig.2.1.2. Structure of TSK

The first layer contains N * M nodes, each of which implements the calculation of the Gaussian function with the parameters cij, sij and bij. From the point of view of fuzzy systems, this is a layer of fuzzification of input variables. The layer is called parametric, because in the process of training the network, the parameters of this layer are subject to selection.

The second layer contains no parameters. From the point of view of fuzzy systems, this is the aggregation layer of the left parts of the products.

The third layer is the generator of (polynomial) functions TSK yi (X) and their multiplier by the weight factor wi. This is a parametric layer in which, in the process of training the network, the coefficients pij, i = 1,2, ..., M, j = 0,1, ..., N, are subjected to adaptation. The total number of coefficients pij in the network is equal to M * (N +1).

The fourth layer is made up of two adder neurons. The first calculates the weighted sum of the signals yi (X), and the second calculates the sum of the weights wi, i = 1,2, ..., M. This is a nonparametric layer.

The last, fifth, layer performs weight normalization. It is also a nonparametric layer.

From the description of the TSK network, it follows that it contains two parametric layers (first and third), the parameters of which are subject to selection during training. The parameters of the first layer will be called nonlinear, since they refer to a nonlinear function, and the parameters of the third layer are linear.

The total number of parameters (linear and non-linear) of the TSK network is

$$M*3*N+M*(N+1)=M*(4*N+1). \qquad (2.1.6)$$

In many practical applications, this is an excessive value; therefore, a limited set of functions m_u (x_j) is often used for the input variables x_j, which reduces the number of nonlinear parameters.

Wang-Mendel network

In a network of this type, the output signal is calculated using the expression

$$y(X) = sum[i=1:M](c_i*w_i)/sum[i=1:M](w_i) \qquad (2.1.7)$$

$$= sum[i=1:M](c_i*prod[j=1:N](mu_{ij}(x_j)))/sum[i=1:M](prod[j=1:N](mu_{ij}$$

$(x_j))),$

where ci is the weight coefficient (from the point of view of fuzzy systems, this is the center of the membership function of the right side of the product), muij () is the Gaussian function (in exponential or rational form) with the parameters of the center cij, width sij and form bij (from the point of view of fuzzy systems muij () - function of belonging to a fuzzy set).

It is easy to see that the expression for y (X) in the Wang-Mendel network is a special case of a similar expression in the TSK network if yi (X) = ci in the latter. Therefore, the Wang-Mendel network is simpler and has the following three-layer structure.

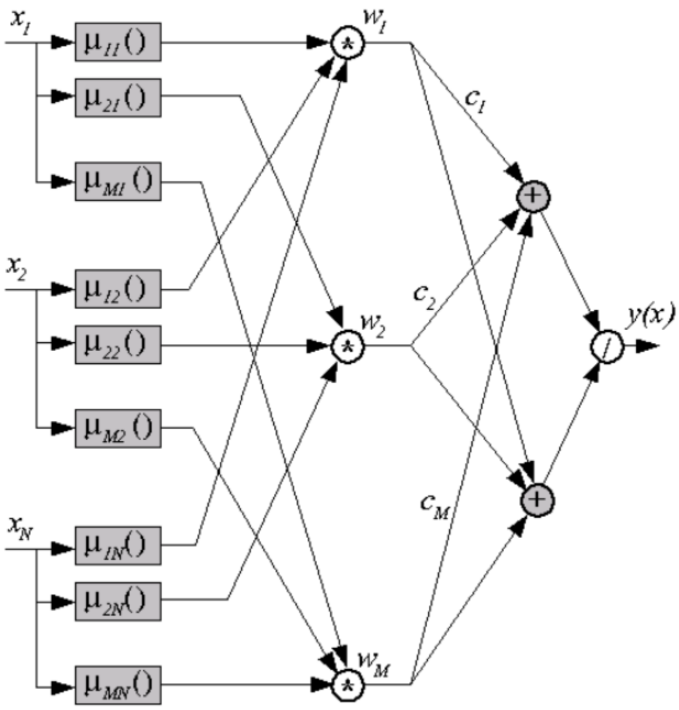


Fig.2.1.3. Structure of Wang-Mendel network

In this network (see Fig.2.1.3), the first and third layers are parametric. The first contains M * N * 3 nonlinear parameters of the Gaussian function, and the third contains M linear parameters ci.

Fuzzy neural networks (both Wang-Mendel and TSK) can be generalized to the case of many output variables. Their training, as well as classical networks, can be conducted with or without a teacher. Supervised learning is based on minimizing the objective function defined using the Euclidean norm

$$E=(1/2)*sum[k=1:p]((y(X^k)-d^k)^2). \qquad (2.1.8)$$

Unsupervised learning is based on self-organizing a network that provides input data clustering.

Adaptive Network Based Fuzzy Inference System (ANFIS)

ANFIS presents an adaptive neuro-fuzzy output system. On the one hand, the ANFIS network is a single-entry neural network and multiple inputs that represent fuzzy linguistic variables. In this case, the terms of incoming linguistic variables are described by standard membership functions, and the terms of the output variable are represented by linear or constant membership functions. On the other hand, the ANFIS network is a system of fuzzy output, in which each rule of fuzzy products has a constant weight of 1. This network ANFIS can be successfully used to configure membership functions and set the base rules in the fuzzy expert system. The algorithms for adaptive fuzzy systems training are relatively labor-intensive and complex in comparison with the training algorithms for neural networks, and, as a rule, consist of two stages:

1. Generation of linguistic rules;

2. Adjustment of membership functions. The first problem relates to the problem of over-the-counter type, the second - to optimization in continuous spaces.

Below is an ANFIS adaptive network model consisting of two rules:

(2.1.9)

Output network is forming by the formula:

(2.1.10
)

ANFIS network layers perform the following functions.

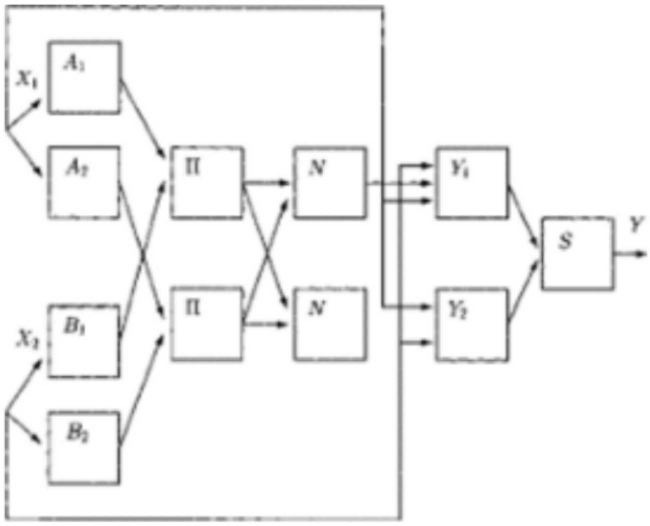


Fig. 2.1.4. ANFIS Network Structure

Layer 1 represented by radial basis neurons and modeling the functions of belonging.

Layer 2 is a layer of "I" neurons that simulate a logical connection and a product $w_{:} = /л_{A}$ (*,)•$ц_{в}$ (л:$_{2}$)-...

Layer 3 calculates the rule that is standardized:

(2.1.11)

Layer 4 forms the value of the output variable:

(2.1.12)

Layer 5. Performs defuzzification:

(2.1.13)

Hybrid learning algorithm

This algorithm is applicable to both of the structures described above, but we will consider it regarding TSK networks as more general. The hybrid algorithm for training fuzzy networks can be considered a variant of the hybrid algorithm for training radial networks.

The algorithm is implemented by alternating two stages:

1. at fixed values of nonlinear parameters cij, sij and bij of the first layer of neurons, the values of linear parameters pij of the third layer of the network are found;
2. at fixed values of the linear parameters pij of the third layer, the nonlinear parameters cij, sij and bij of the first network layer are refined.

Stage 1. At this stage of training, nonlinear parameters are fixed. The output signal is defined as

$$y(X)=sum[i=1:M](w'*(p_{i0}+sum[j=1:N](p_{ij}*x_j))), \quad (2.1.14)$$

Where $w'=v_i=prod[j=1:N](mu_{ij}(x_j))/sum[l=1:M](prod[j=1:N](mu_{lj}(x_j)))=const.$

For K training samples <X^k,d^k>, k=1, 2 , ..., K, we obtain a system of K linear equations $A*P=D,$

where $P=[p_{10}, p_{11}, ..., p_{1N}, ..., p_{M0}, p_{M1}, ..., p_{MN}]^T$ is the weight vector of the third network layer, and $D=[d^1, d^2, ..., d^k]^T$ is a vector of expected values composed of all K training samples. Matrix A is shown below:

$$\begin{array}{ccccccccc}
v^1_1 & v^1_1*x^1_1 & \ldots & v^1_1*x^1_N & \ldots & v^1_M & v^1_M*x^1_1 & \ldots & v^1_M*x^1_N \\
v^2_1 & v^2_1*x^2_1 & \ldots & v^2_1*x^2_N & \ldots & v^2_M & v^2_M*x^2_1 & \ldots & v^2_M*x^3_N \\
. & . & . & . & . & . & . & . & . \\
v^k_1 & v^k_1*x^k_1 & \ldots & v^k_1*x^k_N & \ldots & v^k_M & v^k_M*x^k_1 & \ldots & v^k_M*x^k_N
\end{array}$$

The number of rows K of the matrix A is much greater than the number of its columns M * (N + 1). The solution to this system of linear algebraic equations can be obtained in one step as follows: $P=A^+*D,$

where A + is the pseudo-inversion of the matrix A.

Stage 2. Here, the values of the coefficients of the polynomials of the third layer are fixed and the refinement (usually multiple) of the coefficients of the Gaussian function for the first layer of the network is carried out using the standard gradient method:

$$c^{k+1}_{ij}=c^k_{ij}-nu_c*\partial E^k/\partial c^k_{ij},$$

$$s^{k+1}_{ij}=s^k_{ij}-nu_s*\partial E^k/\partial s^k_{ij},$$

$$b^{k+1}_{ij}=b^k_{ij}-nu_b*\partial E^k/\partial b^k_{ij},$$

where k is the number of the next training cycle (in the "online" mode it coincides with the number of the training sample). From a technical point of view, obtaining analytical expressions for the derivatives of the objective function with respect to nonlinear parameters presents no problems. However, due to their cumbersomeness, these expressions are not given here.

Since in a series of stages, the stage of refining the nonlinear parameters of the Gaussian function has a much lower convergence rate, then in the course of training, the implementation of stage 1, as a rule, is accompanied by the implementation of several stages 2.

2.2. Classification of fuzzy neural networks. Fuzzy conclusions.

Neural networks are convenient for image recognition tasks, but very uncomfortable to explain how they implement it. They can automatically acquire knowledge, but the process of their learning is often slow enough, and the analysis of the trained network is very complicated (a trained network is usually a "black screen" for the user). At the same time, it is difficult to enter any prior information (expert knowledge) to accelerate the learning process in the neural network.

Systems with fuzzy logic, on the contrary, are convenient to explain the conclusions obtained with their help, but they cannot automatically acquire knowledge for their use in the mechanisms of outputs. The need to split universal sets into separate areas, as a rule, limits the number of input variables in such systems to a small value.

The direct distribution neural network can approximate any system that is based on fuzzy rules, and any neural network of direct distribution can be approximated by a system based on fuzzy rules.

The neuro-fuzzy network is a representation of a fuzzy output system in the form of a neural network that is convenient for learning, analysis and use. The structure of the neuro-fuzzy network corresponds to the main blocks of fuzzy output systems.

The main properties of neuro-fuzzy networks is that:

- neuro-fuzzy networks are based on fuzzy systems that are learned using methods used in neural networks;
- a neuro-fuzzy network is usually a multi-layer (often triple) neural network. The first layer is the input variables, the average is fuzzy rules, and the third is the output variables. The weight of the connection corresponds to the fuzzy sets of input and output variables. Sometimes a five-layer architecture is used. In the general case, a fuzzy system does not necessarily have to be presented in this form, but it is a convenient model for the application of teaching methods;
- neuro-fuzzy network always (before, during and after training) can be interpreted as a system of fuzzy rules;
- the learning procedure takes into account the semantic properties of the fuzzy system. This is manifested in the limitation of the possible modifications that apply to the parameters that are being adjusted. It is necessary, however, to say that not all methods of teaching neuro-fuzzy networks take into account the semantics of the system.

Types of combination of fuzzy logic and neural networks by way of interaction distinguish the following:

- fuzzy neural systems. In this case, neural networks use the principles of fuzzy logic to accelerate the process of debugging or improve other parameters. With this approach, fuzzy logic is just a tool of neural networks, and such a system can not be interpreted in fuzzy rules, since it is a "black chest";

- competing neuro-fuzzy systems. In such models, the fuzzy system and the neural network work on one task, without affecting the parameters of each other. It is possible to sequentially process data first by one system, then another;

- parallel neuro-fuzzy systems. In such systems, debugging of parameters is performed with the help of neural networks. Then the fuzzy system functions independently. The following types of parallel neuro-fuzzy models are distinguished:

- Integrated (hybrid) neuro-fuzzy systems with close interaction of fuzzy logic and neural networks.

The term "neuro-fuzzy networks" most often refers to systems of this type. Typically, integrated systems are Mandani or Takagi-Sugeno systems.

By way of displaying fuzzy sets in the network structure, neuro-fuzzy networks have three main types:

- systems built on random fuzzy sets. In such systems, the degree of affiliation is described only for some of the values in the definition area and the function of membership is presented as a vector. Each degree of affiliation corresponds to only one input or output neuron. There are two approaches to implementing such systems. In the first system, it simply approximates the correspondence of the outputs to the inputs, such a system is a "black chest". The second creates a system with a special architecture, in which fuzzy rules are implemented;

- systems, parameterized functions whose belongings are stored in the neurons. An example of such systems is ANFIS (Adaptive-Network-based Fuzzy Inference System);

- systems in which parametric functionality of belonging is used as the

weight of connections between neurons. Such a system can otherwise be called a perceptron with fuzzy connections or fuzzy perceptron.

By the nature of training distinguish the following types of neuro-fuzzy networks:

- self-adjusting neuro-fuzzy networks - with the adaptation of structure and

parameters;

- adaptive neuro-fuzzy networks - with rigid structure and adaptation of

network parameters.

Adaptive neuro-fuzzy networks by the type of optimization method are divided into those using deterministic methods such as gradient search, and those that use stochastic methods, in particular evolutionary ones.

Adaptive neuro-fuzzy networks by type of adaptation parameters are divided on the network with the adaptation of the parameters of the functions of affiliation, networks with the adaptation of the weight of the rules and the network with the adaptation of the parameters of the aggregation operator.

Fuzzy sets have become popular thanks to membership features that add flexibility when processing large amounts of data and its scalability. Consider the basic membership functions that can be used in neuro-fuzzy networks.

Bulk-linear membership functions

As the first type of membership functions we consider functions that, as their names indicate, consist of lines of straight lines, forming a continuous or piecewise-continuous function. The most typical example of such functions is the "triangular" (Fig. 2.2.1.) and "trapezoidal" (Fig. 2.2.2.) Of the membership function.

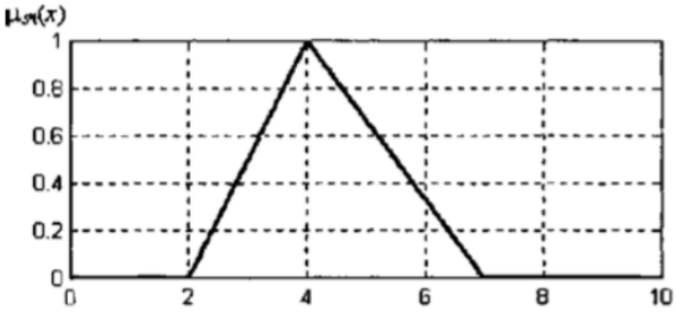


Figure 2.2.1. Triangular membership function

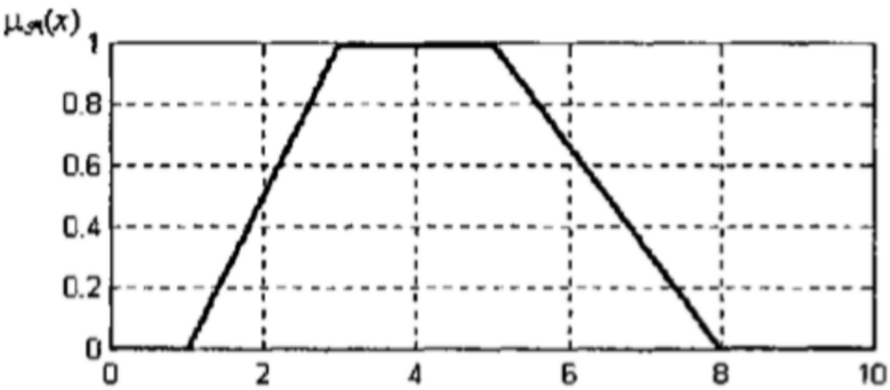


Figure 2.2.2. Trapezoid function of membership

The first of these membership functions in the general case can be given analytically by the following expression:

(2.2.1)

where a, b, c are some numerical parameters that take arbitrary real values and are ordered by the ratio: .

The trapezoidal membership function in the general case can be given

analytically by the following expression:

(2.2.2)

where a, b, c, d are some numerical parameters that take arbitrary real values and arranged by the ratio:.

These functions are used for the task of such properties of sets that characterize the uncertainty of type: "approximately equals", "average value", "located in interval", "similar object", "similar to object", etc. They also serve to represent fuzzy numbers and intervals, which will be discussed further.

Π- similar membership functions

To this type of membership functions can be attributed a whole class of curves, which in their shape resemble a bell, a smooth trapeze or the letter "P".

The first of these functions is called - the P-image function, and in the general case is given analytically by the following expression:

(2.2.3)

where a, b, c, d are some numerical parameters that take arbitrary real values and are arranged by the ratio: a≤b≤c≤d, and the sign "*" denotes the usual arithmetic product of the values of the corresponding functions. This function is shown in Figure 2.2.1

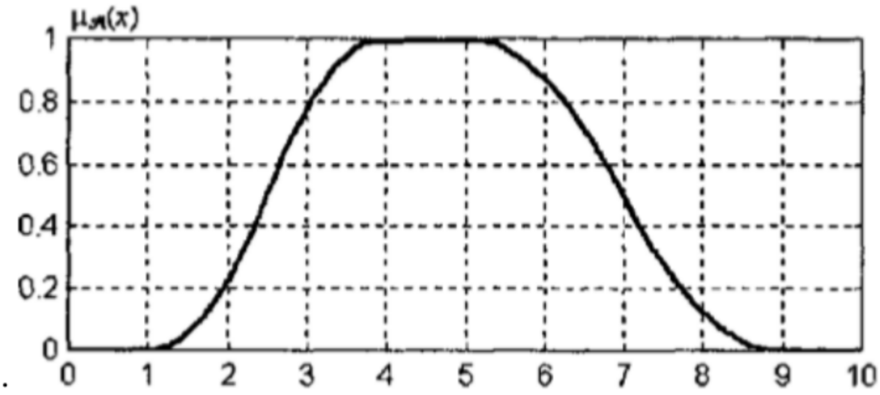


Figure 2.2.3. P-shaped membership function

To the P-shaped functions also refers to the so-called bell-like function (Fig. 2.2.3.), Which in general is given analytically by the following expression:

(2.2.4)

where a, b, c are some numerical parameters that take arbitrary real values and are arranged by the ratio: a <b <s, with the parameter b> 0. Here the function | x | denotes the module of the real number.

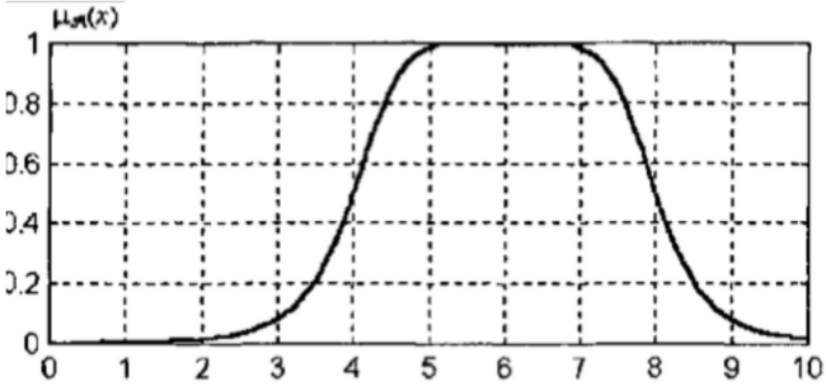


Figure 2.2.4. Bell-like functions

Finally, the last of the functions under consideration of this type is well known in the theory of probabilities the function of the density of normal distribution in the assumption that and which in our case is given analytically by the following expression:

(2.2.5)

where - numerical parameters, with the square of the first of them in the theory of probabilities is called the dispersion of the distribution, and the second parameter c is the mathematical expectation.

Obviously, these last types of membership functions generate normal convex fuzzy sets.

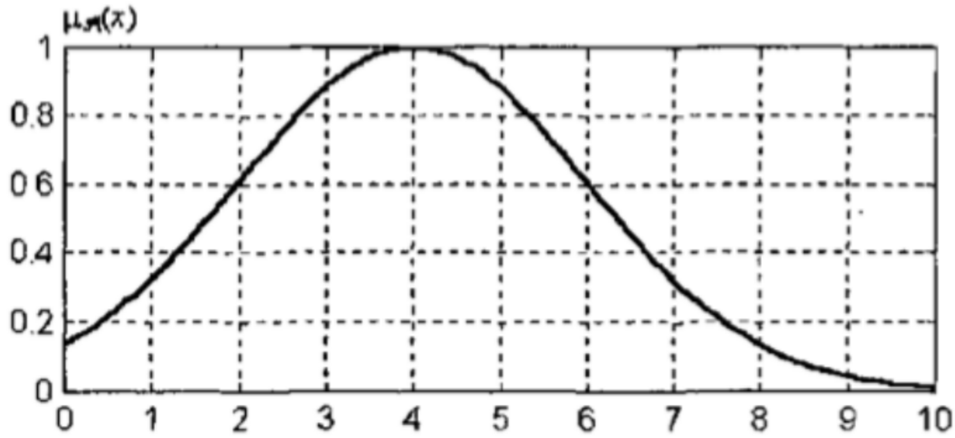


Figure 2.2.5. Normal distribution function

These membership functions are used most often in the construction of fuzzy neural networks. In this paper, the author considers the triangular and sigmoidal membership function, to create a hybrid network. The choice has fallen to these functions due to their simplicity and speed of calculation on personal computers.

2.3. Using fuzzy neural networks for image processing.

An important application sphere of information technologies is the problem of classification of optical medical images and diagnostics in medicine. The advantages of medical diagnostics systems are speed, automation and stability of work which make them very comfortable tools for express medical diagnostics. Despite young age of medical informatics which doesn't exceed 30 years information technologies in a whole are fast penetrating in various spheres of medicine and health defense (family medicine, insurance medicine, integration in European medical space, etc.).

The latest achievements in images processing technologies and machine learning enable to construct systems of automatic detection and diagnostics that may help pathologists-anatomists to make true diagnosis and accelerate his work. Traditional methods of medical images recognition are based on textual descriptors. They use texture features that are most of which are problem-specific and have shown to be complementary in medical images. But they lack generalization ability and demand to construct new textual descriptors for another image [3]. Therefore deep learning and convolutional neural networks (CNN) became powerful and efficient tools for processing and pattern recognition of medical images. Nowadays CNN are widely used for medical images of human organs classification obtained by different sources (MRI, CT, X –rays images, etc.). The main advantages of CNN are abilities to learn presentations, extract informative features of images and application of efficient training methods for end-to-end learning [4, 5]. Therefore CNN now are widely used for medical images analysis and classification, especially for cancer diagnostics. Usually CNN consists of convolutional layers, pooling layers which are used or features extraction and decision layer-MLP (multilayer perceptron) as classification unit [4, 5]. An interesting approach based on application of dual deep convolutional neural network with a synergic signal system is suggested in the study [6]. The synergic signal is used to verify whether the input image pair belongs to the same category and to give the corrective feedback if a synergic error exists. Synergic deep learning (SDL) model can be trained 'end to end'. In the test phase, the class label of an input can be predicted by averaging the likelihood probabilities obtained by two convolutional neural network components. Up to date a lot of modern methods for medical images classification were developed. A comprehensive review of modern classification techniques is presented in book [7]. The book covers several complex image classification problems using pattern recognition methods, including Artificial Neural Networks (ANN), Support Vector Machines (SVM), Bayesian Networks (BN) and deep learning. Further, numerous data mining techniques are discussed, as they have proven to be good classifiers for medical images. To the problem of recognition time-dependent data sets of medical images is devoted the monograph [8]. An important feature of this book is the exploration of different approaches to handle and identify time dependent biomedical images. Biomedical imaging analysis and processing techniques deal with the interaction between all forms of radiation and biological cells or tissues in order to visualize small particles and opaque objects and to enable efficient recognition of biomedical patterns [9].

CHAPTER 3. IMPROVING THE QUALITY OF MEDICAL IMAGE PROCESSING BASED ON FUZZY NEURAL NETWORKS

3.1. Rationale for the choice of topology. Hybridization.

The other alternative approach to CNN for medical images analysis and classification is application of fuzzy neural networks (FNN), in particular FNN NEFClass. FNN NEFClass was firstly suggested by D. Nauck and W. Kruse in the studies [10, 11]. It was modified and developed in the study [12] (so-called FNN NEFClass M) The main advantages of FNN NEFClass as classifier are: possibility to work with incomplete and fuzzy input data; performing fuzzy classification of input patterns (images); speed and high accuracy [10-12]. FNN NEFCLASS was firstly applied for medical images diagnostics in the problem of breast tumors classification in two classes: benign or malicious [10, 12]. The important sphere of medical information systems is pattern recognition of tumors on human organs tissue and early detection of possible cancer It enables to take urgent healing and escape lethal outcome. One of such tasks is cervix epithelium state analysis and diagnostics using optical images obtained with colposcope (a method of survey of a mucous membrane of part of a neck of a uterus in the conditions of additional lighting and optical increase with the help of a colposcope). As a result of carrying out a colposcopy by the doctor the increased pictures of images with preliminary splitting into classes of diseases are provided. The problem of classification cervix epithelium state using images obtained with colposcope was considered in the studies [13, 14] where for its solution was suggested the application of crisp neural networks Back propagation, neural networks with radial basis functions (RBFNN) and cascade RBFNN and their efficiency investigated. The goal of this paper is the investigation of modified fuzzy neural network NEFClass M for recognition of state of cervix epithelium in medical diagnostics and comparison of its efficiency with conventional RBF and cascade neural networks [9].

Modern neuro-fuzzy approaches have this form. The neural network and the fuzzy system are combined into one homogeneous architecture. The system can be interpreted either as a special neural network with fuzzy parameters, or as a fuzzy system implemented in parallel. Some of these approaches are reinforcing types of learning that are particularly suitable for control tasks, while others are multi-purpose models that use teacher-based learning and can be used for data analysis, such as the NEFCLASS approach.

Since we are only interested in hybrid neuro-fuzzy systems here, in the following descriptions we will limit ourselves to the information needed as a basis for such an approach.

The hybrid approach is the ability to interpret the fuzzy system rule base from a neural network perspective. Fuzzy sets can be thought of as weights and input and output variables, and rules can be interpreted as neurons. Thus, fuzzy systems can be interpreted as a special neural network. The learning algorithm works by changing the structure and / or parameters, which means the inclusion or removal of neurons or adaptation of weights. The structure of these neural network is shown in Fig. 3.1.1.

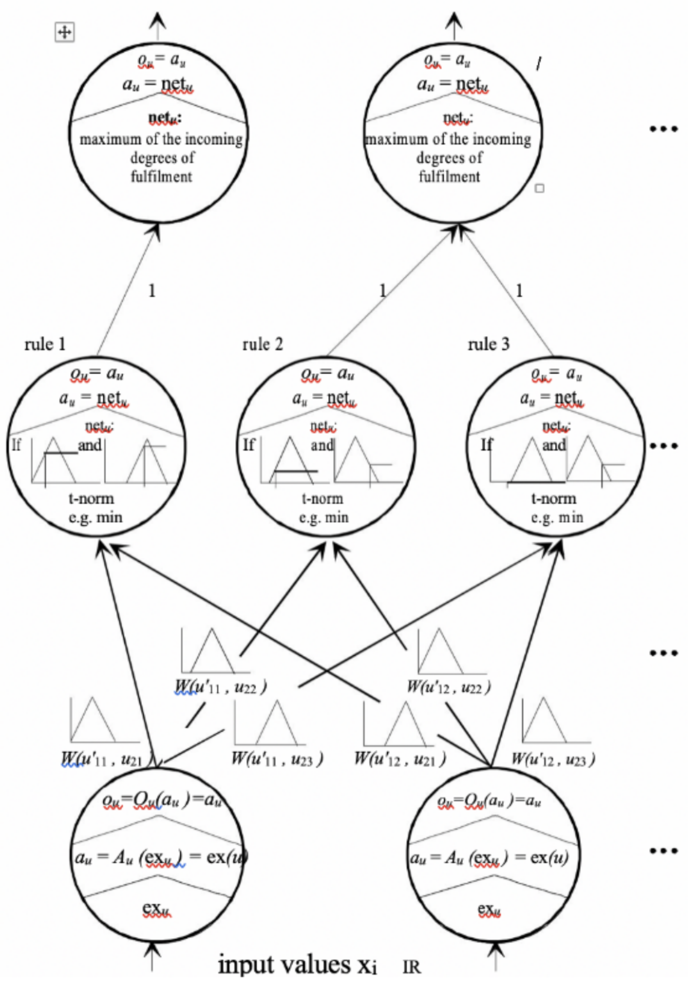


Fig.3.1.1. Representation of a neural network of a neurophysical classifier

3.2. Problem statement of structural-parametric synthesis.

For the process of developing and classifying medical images, the NEFCLASS neuro-fluffy classifier, which, on the basis of the initial vibration, is responsible for the task of classifying images of CT lungs. Algorithm for navchannya given unclear classifier will backpropagation algorithm.

The task of data classification (the main task of the theory of recognition) is currently one of the most urgent areas of application of artificial intelligence systems. To solve it, various approaches and directions were proposed, among which the most popular are solutions that combine neural networks and fuzzy inference systems. One of these solutions is the system NEFCLASS (NEuro-Fuzzy CLASSifier) based on generic architecture fuzzy perceptron. This system has a number of undoubted advantages that distinguish it from the rest. Among the most important should be mentioned the simplicity of implementation, the high speed of the learning algorithms, and also, which is the most important, high accuracy of data classification - at the level of the best systems in this area. At the same time, the basic NEFCLASS system has a number of disadvantages – the formulas for training parameters are empirical; moreover, it is not clear how choose the learning rate parameter a in the learning algorithm. The purpose of this work is create an alternative model of a fuzzy neural system.

We will assume that there is a training sample from M pairs of experimental data linking inputs with an exit y investigated dependence:

(3.2.1)

where is the input vector in the r-th pair and is the corresponding output. Tuning is the finding of such parameters of the membership functions of the terms of the input variables and the weighting coefficients of the rules, which minimize the deviation between the desired and actual behavior of the fuzzy classifier on the training sample.

3.3. Review of methods for solving the problem.

The fuzzy classification problem is the problem of dividing the space of classification features into fuzzy classes. You can describe the space of classification signs with fuzzy areas and manage each area using fuzzy rules [15]. A typical fuzzy P_i classification rule that demonstrates the relationship between input space and classes is as follows:

(3.3.1)

where x_{sj} represents the j-th input element of the s-th sample; Aij denotes the fuzzy set of the j-th input element in the m-th rule; C_k is a k-th class label. A_{ij} is determined by the corresponding membership function.

To achieve this goal, the fuzzy parameter of the membership function [16] and the fuzzy cluster must be optimized. When fuzzy classification rules are constructed as a network, their parameters can be adapted to the neural network. As a result, systems with fuzzy classification and neural networks can be combined while preserving their positive properties. The combined system is called a neuro-fuzzy classifier, which is an adaptive network with multiple inputs and multiple outputs [17].

Neuro-fuzzy classifier is a multilayer unidirectional network consisting of the following layers: input layer, fuzzy activation function (AF) layer, fuzzification layer, defuzzification layer, normalization layer and output layer. The classifier has multiple inputs and multiple outputs. In fig.3.3.1 shows an NFC with two classifications and three classes. Each input is identified by three linguistic variables. Thus, there are nine fuzzy rules.

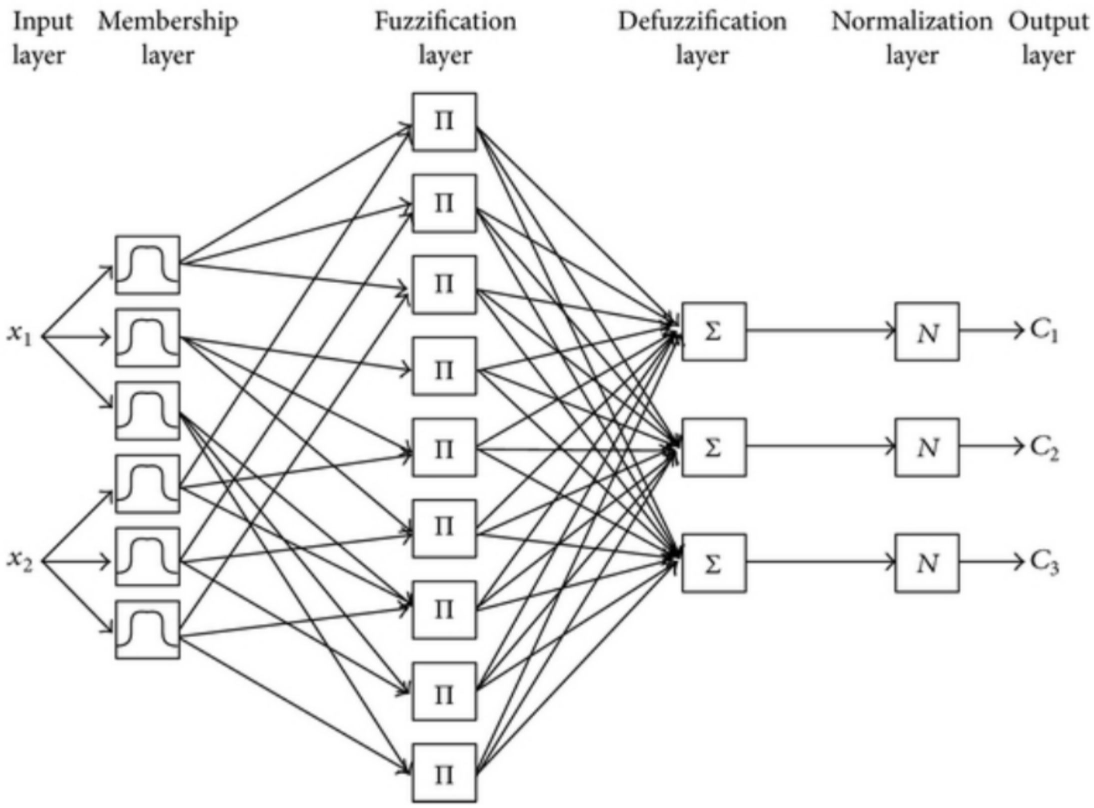


Fig. 3.3.1. The structure of a neuro-fuzzy classifier

Membership layer: the membership function of each input is identified in this layer. Several types of membership functions can be used. In this study, a Gaussian function is utilized, since this function has fewer parameters and smoother partial derivatives for parameters. The Gaussian membership function is defined as

(3.3.2)

where is the membership grade of *i*-th rule and *j*-th feature; represents the *s*-th sample and *j*-th feature; and are the center and the width of Gaussian function, respectively.

Fuzzification layer. Each node in this layer generates a signal corresponding to the degree of fulfillment of the fuzzy sampling rule. This signal can be called the degree of excitation of the fuzzy rule in relation to the object to be classified. The degree of excitement according to this rule is as follows:

(3.3.3)

where N – number of classification features.

Defuzzification layer. In this layer, the weights of the outputs are calculated; each rule affects each class according to their weights. If a rule governs a specific area of a class, the weight between that rule's output and the specific class will be greater than the other weights. Otherwise, the class weights are small. The output weight for the s-th sample belonging to the k-th class is calculated as follows:

(3.3.4)

where w_{ik} denotes the degree of belonging to the k-th class, which is controlled by the i-th rule and U represents the number of rules; o_{sk} denotes the weighted output for the s-th sample belonging to the k-th class.

Normalization layer. The network outputs must be normalized, since the summation of the weights can be greater than 1, in some cases

(3.3.5)

where h_{sk} represents the normalized output value for the s-th sample, which belongs to the k-th class, and K is the number of classes. After that, the class label of the s-th sample is determined by the maximum value as

(3.3.6)

where C_s denotes the class label for the s-th sample.

3.4. Construction of the algorithm of structural-parametric synthesis.

The purpose of the NEFCLASS (NEuro Fuzzy CLASSifier) model is to obtain fuzzy rules from a multitude of data that can be divided into different classes. Fuzzy rules describe the data in the form:

R: if has the function of belonging ,

then the sample belongs to class i, where - fuzzy sets. The task of NEFCLASS is to determine whether the class belongs to the input sample. It is implied that the intersection of two different sets is empty. Let's take a closer look at the architecture of the NEFCLASS model.

The base of rules is an approximation of an unknown function and describes a classification problem, where such that belongs to the class. Fuzzy sets and linguistic rules represent an approximation and determine the result of the NEFCLASS system. They proceed from a multitude of samples by way of study. It is obligatory to follow the rule that for each linguistic meaning there can be only one representation of a fuzzy set.

The NEFCLASS system has a 3-layer sequential architecture. The first layer contains the input neurons in which the input samples are presented. Neuron activation usually does not change the input value. The hidden layer contains fuzzy rules, and the third layer consists of output neurons of each class. Activation for rule neurons and for source layer neurons with sample p is calculated as:

(3.4.1)

(3.4.2)

where *W (x, R)* is the fuzzy weight of the connection of the input neuron *x* with the neuron of the rule *R*, and *W (R, c)* is the fuzzy weight of the connection of the neuron of the rule *R* with the neuron of the output layer c. Instead of using the operations of taking the maximum and minimum, it is possible to use other functions of the t-norm and the t-conorm, respectively. The base of rules is an approximation of an unknown function and describes a classification problem, where such that belongs to the class.

Fuzzy sets and linguistic rules represent an approximation and determine the result of the NEFCLASS system. They proceed from a multitude of samples by way of study. It is obligatory to follow the rule that for each linguistic meaning there can be only one representation of a fuzzy set.

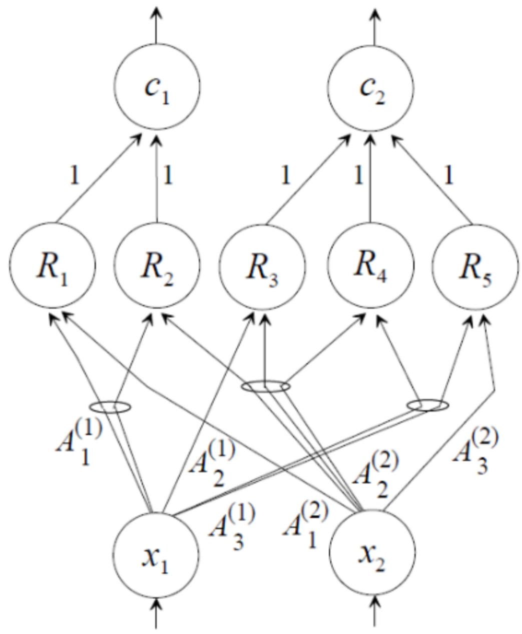


Fig. 3.4.1. Architecture of the model NEFCLASS

On Fig. 3.4.1 R_k represent rules, $A_j^{(i)}$ weight $W(x_i\ R_k)$, where index j selects fuzzy sets of partition. NEFCLASS uses common weights for some connections (on Fig. 3.4.1 it shown by the ellipses drawn by the connection).

$W(R_k,\ c_m)$ – is connections from rule R_k to output of block c_m. For semantic reasons, that is to avoid a weighted rule, these connections are fixed either 0 (the connection does not exist) or 1 (the connection exists). Each device is connected to one output. Activation of the output is calculated using the maximum operation instead of the weighted sum [2].

Learning a Rule Base - The Algorithm

A NEFCLASS system can be built from partial knowledge about the patterns, and can be refined by learning, or it can begin with an empty rule base that is filled by creating rules from the training data. For each input variable the user must decide how many fuzzy sets are to be used to partition the domain of the respective variable. By this the granularity for each variable and the linguistic terms that can be used by the classifier are given. For some variables the user might prefer to distinguish just between small and large, for some other variables a finer partition may be useful. The user must also specify a value of k_{max}, i.e. the maximum number of rule nodes that may be created in the hidden layer. For each class there must be at least one rule. It is also possible to let NEFCLASS find a suitable value for k_{max} by itself [2].

In the following it is assumed that triangular membership functions described by three parameters are used:

$$\mu : \mathbb{R}^n \rightarrow \mu(x) = \begin{cases} \frac{x-a}{b-a} & \text{if } x \in [a,b), \\ \frac{c-x}{c-b} & \text{if } x \in [b,c], \\ 0 & \text{otherwise.} \end{cases}$$

Let's consider system NEFCLASS with:

- n input units $x_1, ..., x_n$;
- $k \leq k_{max}$ initial rule units $R_1, ..., R_k$ (prior knowledge, $k = 0$, which means no prior knowledge is given);
- m output units $c_1, ..., c_m$;
- a learning set of s patterns, each consisting of an input pattern and a target pattern .

Assume that NEFCLASS is initialized with $k < k_{max}$ fuzzy rules. The rule base of NEFCLASS is completed by finding for each pattern p a combination of fuzzy sets that yields the highest degree of membership for each value p_i. So, for each value p_i from plural find the value of the membership function, $_i$) such that,

$_i$) = , де j ∈i…q, j – the class number, the values of the membership functions obtained are push into our linguistic norm A = (), this action must be performed for all input values p having formed all linguistic norms. For further work we introduce the concept of the degree of filling the model of a class C, and the index of the unit from our sample c, where c = index(p) from . Since 1 input object characterizes several attributes, for each sign we find the corresponding value of the membership function of Aj and we will write their sum in C_j, repeat for all incoming signs and fill in the corresponding ones

C_j, form of the entry in the form of a formula: $C_j(c) = C_j(c) + A(p)$. Based on the data of linguistic norms, we will create rules, and save them in the memory of our program.

For each rule, we will calculate its efficiency, $perf_j = C(c)$ - , where *c* this is the index of the maximum sum of items $C_{j,}$ which we have formed in past action. So, now we have the established rules that are stored in our program's memory and the corresponding performance indexes for each rule. The best rules are determined by calculating performance values for each rule. If the rule correctly

classifies the template, then its degree of performance is added to its performance,

if not, then the degree of performance is subtracted. The value of productivity can be calculated on the fly, so that training according to the rules ends after one sweep through the training set. Rules for NEFClass are selected in two algorithms: "best in class" (1) and "best rule" (2).

For the algorithm (1), for all formed rules find $R = \text{argmax}\{ perf_j \}$, We will leave the rules with the best efficiency, all the others will be removed from the base until the number of rules will not be equal k_{max} – maximum number of rules given by a user.

For algorithm (2), until the number of rules will not be equal , where m the number of input features for a class, find $R = \text{argmax} \{ perf_j \}$, We remove all other rules from the base. On this, the formation of rules can be considered complete, then proceed to determine the algorithm for learning NEFClass network based on their errors.

Training

After distributing the template, an error is determined for each output. Based on this error, for each active block of rules is, the degree of its implementation is greater or less. The membership function that is responsible for the degree of execution is identified, and only this fuzzy set is adapted accordingly. Fuzzy sets change many times only if it does not violate the restrictions specified by the user.

For each input value p from the set we find its output - si from the system. For each output we find the error of our corresponding output:

$ec_i = t_i$ – activation(c_i) -, where t is the expected value (class label), c_i is the output of the system. For each rule R, where activation (R)> 0 we find the error for the corresponding rule:

(3.4.2)

where W (R, c) exit from the rule, define

j = argmin{W(x_i, R(p_i)} the found minimum value is used in the calculations to determine the new value of the membership function , for the new value of the membership function we find the new maximum, minimum and average values c, a, b - in accordance

$$\delta_b = \sigma \cdot e_R \cdot (c_\mu - a_\mu) \cdot \mathrm{sgn}(p_i - b_\mu);$$
$$\delta_a = -\sigma \cdot e_R \cdot (c_\mu - a_\mu) + \delta_b;$$
$$\delta_c = \sigma \cdot e_R\, (c_\mu - a_\mu) + \delta_b;$$

We use the found values to shift our set, for clarity we use Fig.3.4.2.

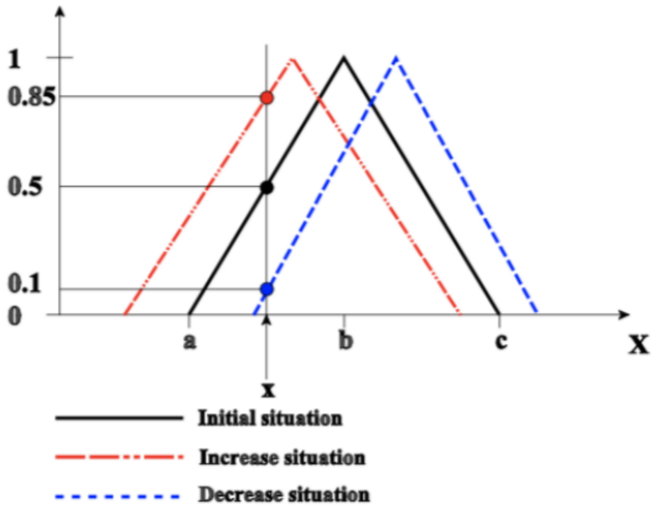


Fig 3.4.2. The shift of the set depending on the new parameters.

Adaptation of fuzzy sets is carried out by simply changing the parameters of the membership function so that the degree of membership for the current value of the object increases or decreases, respectively.

Fig. 3.4.3 displays the situation after the learning algorithm for the membership functions was applied to improve the classification result. Here no constraints were used to restrict the learning process. As can be seen the resulting fuzzy partitions are no longer nicely interpretable. Such a result is an indication to repeat the learning process with other parameters. In this case it would have been better to allow the system to create four rules and to use constraints for training the membership functions. This example is to illustrate that it is important for the user to work interactively with approaches like NEFCLASS to obtain readable solutions.

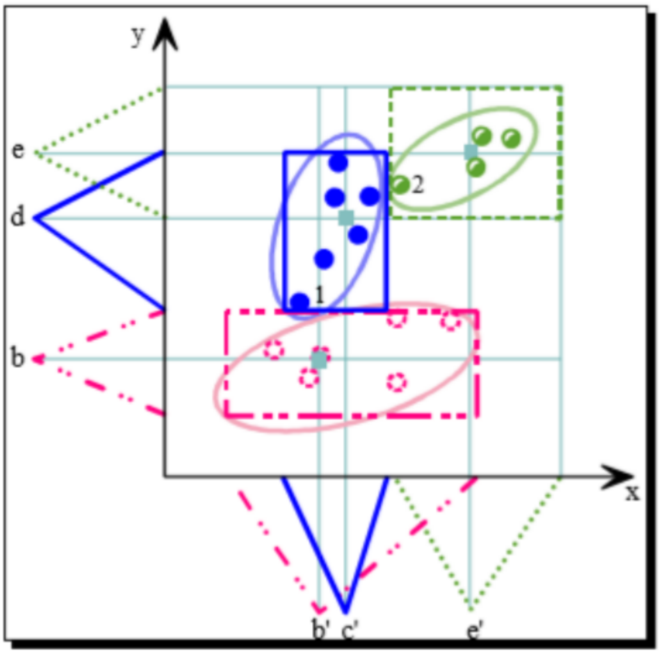


Fig 3.4.3. Classification after learning fuzzy sets

Compared to neural networks, NEFCLASS uses a much simpler learning strategy. There is no vector quantisation involved to find rules (clusters), and the membership functions are not trained by gradient descent. Fuzzy rule creation can be seen as a selection from an initially given virtual rule base, specified by a fuzzy partition of the input domain.

In terms of NEFCLASS architecture and data flow, fuzzy sets are learned by an algorithm similar to backpropagation. We use the term backpropagation to denote the idea of a training procedure, rather than a special implementation in the form of an algorithm. Reducing the spread means calculating the original error and spreading it back through the architecture from the source blocks to the input units. This error signal is used to change the parameters locally. Neural networks often implement back propagation by means of a gradient descent. NEFCLASS does not calculate gradient information. Instead, it uses a much simpler heuristic. In addition, the adaptability of the

NEFCLASS system is limited compared to neural networks. This restriction is due to the initially given fuzzy sections, which determine the shape and maximum number of clusters, as well as constraints that do not allow certain changes in fuzzy sets [2].

CHAPTER 4. SOFTWARE DEVELOPMENT

4.1. Software structure

In this section we will discuss about software for the NEFCLASS neural netvork. In order to accomplish the task in this work, it was decided to make the most of the hardware capabilities of modern computers, in particular, the use of NVIDIA graphics cards is promising. This makes it possible to accelerate the execution of matrix operations (which are the main operations in the training of artificial neural networks) in 10-20 times compared with the performance of the Intel i9 processor. This enables to quickly teach the chosen architecture of the neural network and, based on the results obtained, quickly change it.

There remains the issue of choosing a program for writing and training our deep believe neural network and automated adjustment system. The choice of programming language was between MATLAB and Python.

MATLAB is a package of numerical analysis applications, as well as the programming language used in this package. The system was created by The MathWorks and is a convenient tool for working with mathematical matrices, drawing functions, working with algorithms, creating user interfaces with programs in other programming languages. Although this product specializes in numerical computing, special tools work with Maple software, which makes it a complete system for working with algebra. However, this package also has a library for creating and training neural networks, which is called the Neural Network Toolbox. Its current name is Deep Learning Toolbox.

Deep Learning Toolbox is the MATLAB expansion pack containing tools for designing, modeling, developing neural network visualization.

Neural network technologies allow solving such problems, the solution of which by classical formal methods is difficult or impossible.

The package provides comprehensive support for typical neural network paradigms and has an open modular architecture. The package contains command line functions and a graphical user interface for quick step-by-step creation of neural networks.

In addition, Deep Learning Toolbox provides support for Simulink, which allows you to simulate neural networks and create blocks based on the developed neural network structures.

The main features of the package:

- Graphical user interface for step-by-step creation, training and simulation modeling of neural networks;
- Supports the most common managed and unmanaged network structures;
- Full list of training and testing functions;
- Dynamic network learning algorithms, including time delay, non-linear autoregression (NARX), chain and tunable dynamic structures;
- Simulink blocks for creating neural networks and advanced blocks for monitoring systems;
- Automatic generation of Simulink blocks from neural network objects;
- A modular network view that allows you to create an unlimited number of input layers and interconnected networks, as well as a graphical representation of the network architecture;
- Advance and post processing functions and Simulink blocks to improve the learning process and evaluate network performance;
- Visualization of the topology and learning process of the neural network.

However, access to this package is limited and requires payment for use, as it is a private property of the developer company. Therefore, this fact is a big drawback of using both the MATLAB package and its separate library Deep Learning Toolbox.

Another option that allows you to create and training neural networks is the programming language - Python.

Python is a popular high-level programming language that is designed to create applications of various types. This includes web applications, games, desktop programs, and work with databases. Python has become quite common in the field of machine learning and artificial intelligence research.

The main features of the Python programming language:

- Script language. Program code is defined in the form of scripts.
- Support for a wide variety of programming paradigms, including object-oriented and functional paradigms.
- Interpretation of programs. To work with scripts you need an interpreter that runs and executes the script.

Running a Python program is as follows. First we write in a text editor a script with a set of expressions in a given programming language. We transfer this script to execution to the interpreter. The interpreter translates the code into intermediate bytecode, and then the virtual machine translates the received bytecode into a set of instructions that are executed by the operating system.

It is worth noting here that although formally the translation by the source code interpreter into bytecode and the translation of the bytecode by the virtual machine into a set of machine instructions represent two different processes, but in fact they are combined in the interpreter itself.

Fig. 4.1.1.

- Portability and platform independence. It does not matter what operating system we have - Windows, Mac OS, Linux, we just need to write a script that will run on all of these operating systems with an interpreter.
- Automatic memory management.
- Dynamic typing.

Python is a very simple programming language, it has a concise and at the same time quite simple and understandable syntax. Accordingly, it is easy to learn, and in fact this is one of the reasons why it is one of the most popular programming languages for learning. In particular, in 2014 it was recognized as the most popular programming language for studying in the USA.

Python is also popular not only in the field of learning, but in writing specific programs including commercial ones. To a large extent, therefore, many libraries have been written for this language that we can use.

In addition, this programming language has a very large community, on the Internet you can find a lot of useful materials and examples on this language, get expert help from specialists.

The open-source Anaconda Distribution is the easiest way to perform Python/R data science and machine learning on Linux, Windows, and Mac OS X. With over 11 million users worldwide, it is the industry standard for developing, testing, and training on a single machine, enabling *individual data scientists* to:

- Quickly download 1,500+ Python/R data science packages
- Manage libraries, dependencies, and environments with Conda
- Develop and train machine learning and deep learning models with scikit-learn, TensorFlow, and Theano
- Analyze data with scalability and performance with Dask, NumPy, pandas, and Numba
- Visualize results with Matplotlib, Bokeh, Datashader, and Holoviews

Open-source is a type of computer software in which source code is released under a license in which the copyright holder grants users the rights to study, change, and distribute the software to anyone and for any purpose.

Thanks to open source, the Anaconda distribution for Python will be used in this work. And also the Keras neural network library, which is an add-on over the Deeplearning4j, TensorFlow and Theano frameworks. These frameworks and the Keras library use free licenses BSD, MIT, Apache 2.0.

A free license is a license that allows any person to be freely used or distributed commercially or non-commercially, and allows the creation of derivative works (translations, adaptations, remixes, just modified versions, etc.) and distribute such works on the same terms. These licenses are very similar and they all have permission licenses, that is, they allow programmers to use the licensed code in the closed software, provided that the license text is provided with this software.

The following software and hardware were used to develop the software:

1. Processor Intel i5 5-th generation;
2. RAM 12 GB;
3. SSD 512 GB;
4. GPU NVidia GeForce 820M;
5. OS Windows10
6. Anaconda Distribution
7. Tensorflow 1.4
8. Keras 2.0
9. Python 3.6.

4.2. User interface description

So, we decided to use the Python program language with their distribution Anaconda 3.6 and libraries Keras 2.0, Tensorflow 1.4 for creation our neural networks.

Firstly, it is necessary to install Python for our PC and other dependencies using the package manager **pip:**

1) Download and install Python 3 from the official Python site. https://python.org
2) Install the following dependencies using **pip** package manager using according comands
   i. Tensorflow
      pip install tensorflow
   ii. Numpy

pip install numpy

iii. SciPy

pip install scipy

iv. OpenCV

pip install opencv-python

v. Pillow

pip install pillow

vi. Matplotlib

pip install matplotlib

vii. H5py

pip install h5py

viii. Keras

pip install keras

ix. TansorFlow

pip install tensorflow

Consider installation Anaconda. In official website https://www.anaconda.com/products/individual DOWNLOAD this libraries and execute installation process.

After successful installation of Anaconda, start the Anaconda Navigator program from the Start menu. You should see this logo when starting up:

Fig. 4.2.1. Anaconda logo

Then the navigator itself will open. This is the starting point for working with Anaconda.

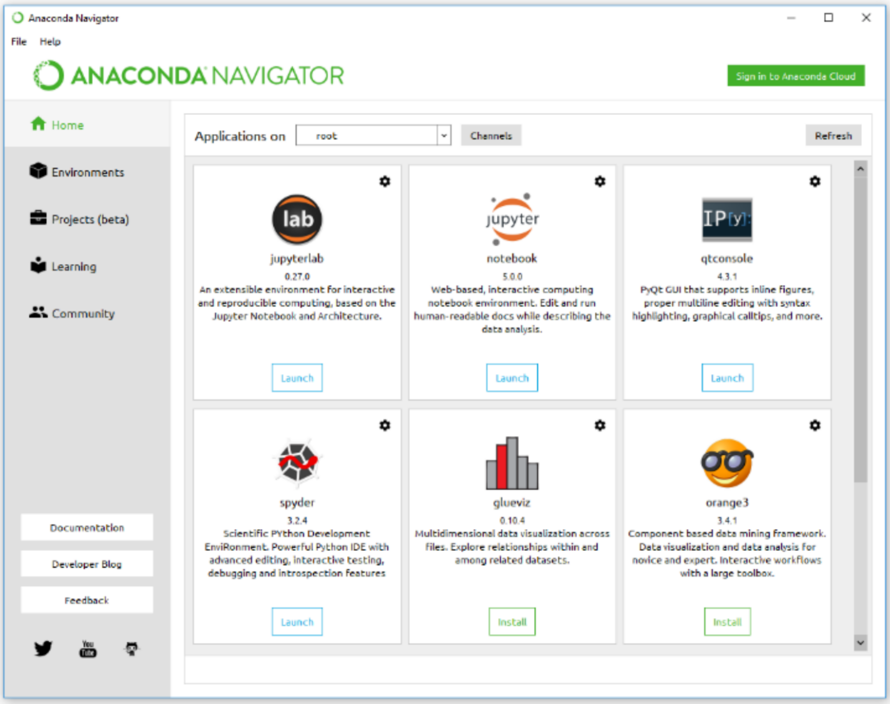


Fig. 4.2.2. Anaconda dashboard

In the central part of the window there are various programs that are included in the Anaconda package. Some of them have already been installed.

Basically, we will use “jupyterlab”: beautiful documents are created in it.

The left side shows the sections of the navigator. By default, the "Home" section is open. In the “Environments” section you can enable / disable / load additional Python modules using a convenient interface.

JupyterLab

In the “Home” section of the navigator, launch (“Launch” button) the “jupyterlab” program (the very first in the list).

You should have a default browser open with the JupyterLab environment in a separate tab.

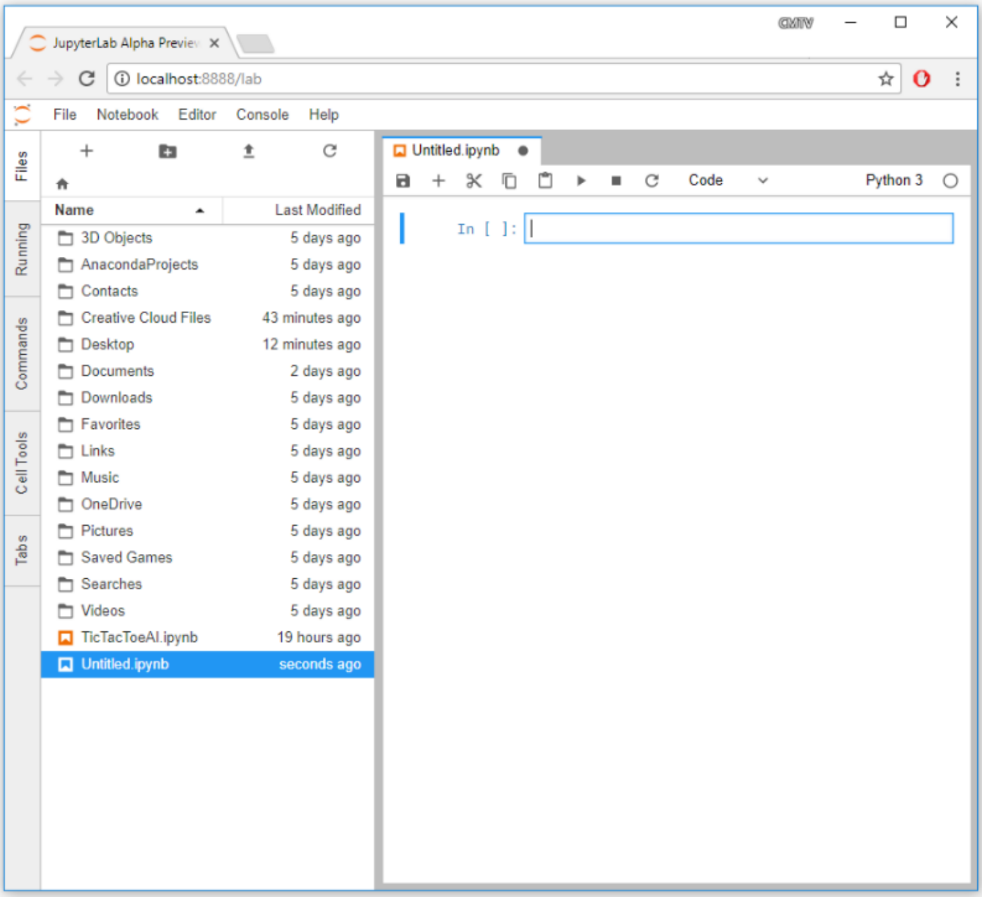
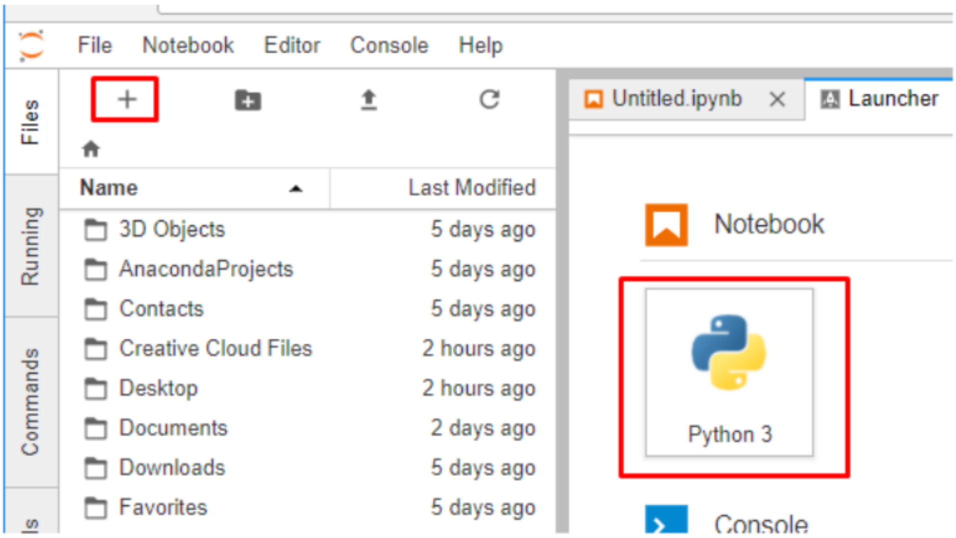


Fig. 4.2.1. JupiterLab environment

The notepad file “untitled.ipynb” is open on the right. If there is nothing on the right, you can create a new empty notebook by clicking on the “+” in the upper left corner and selecting «Notebook Python 3»:



Usually we write Python code in files with the *.py* extension, and then the Python interpreter executes them and outputs the data to the console. For convenient work with such files, programming environments are often used (IDE).

4.3. Description of input, output data

The structure of model neuro fuzzy classifier NEFCLASS in program must be looks like as shown below:

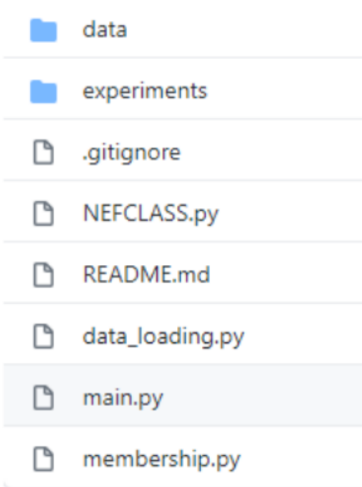


Fig. 4.3.1. Structure of model NEFCLASS

In folder **data** is locate .csv files with training and test samples and in **experiments** folder are written the results of training and testing the neural network.

main.py – is the file where is the initialization all neural network.

```python
from membership import *
from NEFCLASS import *
from data_loading import *
import argparse
import numpy as np
seed = 42

def train(args, labels, train_data, train_targets, test_data, test_targets, universe_max, universe_min, verbose=False):
    '''
    initialize model
    '''
    model = NEFCLASS(num_input_units= args.num_input_units, num_fuzzy_sets=args.num_sets, \
                     kmax=args.kmax, output_units=args.output_units, universe_max=universe_max, universe_min=universe_min,\
                     membership_type=args.mf)
    abcs = [build_membership_function(train_data[d],labels) for d in range(train_data.shape[1])]
    model.init_fuzzy_sets(abcs)

```

Fig. 4.3.2. Initialization network

There is initialization and import of data from other files like **NEFCLASS.py**, **membership.py**, **data_loading.py.**

In **NEFCLASS.py** consider the structure neural network

```python
from membership import *
import numpy as np
EPSILON = 0.0000001

class NEFCLASS:
    def __init__(self, num_input_units, num_fuzzy_sets, kmax, output_units, universe_max, universe_min, membership_type):
        self.input = _input_layer(num_input_units, num_fuzzy_sets, universe_max, universe_min,membership_type)
        self.rule = _rule_layer(kmax, output_units)
        self.output = _output_layer(output_units)
        self.universe_max = universe_max
        self.universe_min = universe_min
        self.membership_type = membership_type

    def init_fuzzy_sets(self,abcs):
        self.input.init_abcs(abcs)

    def __call__(self,x, t):
        m,ante = self.input(x)
        o = self.rule(m)
        c = self.output(o)
        return(c)
```

Fig.4.3.3. Main structure NEFCLASS

Here is 3 layer are described:

1. Input layer (fig. 4.3.4)
2. Rule layer (fig. 4.3.5)
3. Output layer (fig. 4.3.6)

```python
class _input_layer:
    def __init__(self,num_input_units, num_fuzzy_sets, universe_max, universe_min, membership_type):
        self.num_fuzzy_sets =  num_fuzzy_sets
        self.num_input_units = num_input_units
        self.abcs = None
        self.last_m = None
        self.last_ante = None
        self.last_input = None
        self.universe_max = universe_max
        self.universe_min = universe_min
        self.membership_type = membership_type
    def init_abcs(self,abcs):
        self.abcs = abcs

    def __call__(self,x):
        self.last_input = x
        m = []
        for i in range(len(x)):
            m.append([determine_membership(x[i], v, self.universe_max[i], self.universe_min[i], self.membership_type) for k, v in self.abcs[i].items()])
        ante = [mem.index(max(mem)) for mem in m]
        self.last_m = m
        self.last_ante = ante

        return m, ante

    def update_fuzzy_sets(self, sigma, interm):
        error_rule, (j1,j2), mu = interm
        key = list(self.abcs[j1].keys())[j2]
        abc= self.abcs[j1][key]
        # print(abc)
        delta_b = sigma * error_rule * (abc[2] - abc[0]) * np.sign(self.last_input[j1]- abc[1])
```

Fig. 4.3.4. Input layer

```python
class _rule_layer:
    def __init__(self, kmax, output_units):
        self.kmax = kmax
        self.output_units = output_units
        self.nodes = []
        self.antes = []

    def __call__(self, m):
        tally = [[] for i in range(self.output_units)]
        for n in self.nodes:
            tally = n(m, tally)
        return tally



    def learn(self, antecedent, consequent):
        if len(self.nodes) < self.kmax:
            if str(antecedent) not in self.antes:
                self._create_node(antecedent, consequent)
                self.antes.append(str(antecedent))
        # print(len(self.nodes), len(self.antes))


    def _create_node(self, antecedent, consequent):
        self.nodes.append(RuleNode(antecedent, consequent, self.output_units))

```

Fig. 4.3.5. Rule layer

```python
class _output_layer:
    def __init__(self, output_units):
        self.output_units =  output_units

    def __call__(self,o):
        #o is tally
        #max as t-conorm
        output = [max(node) if len(node) != 0 else 0 for node in o]
        # print(output)
        total = sum(output)
        # print(total)
        output = [o/total for o in output]
        return output

```

Fig. 4.3.6. Output layer

Process of initializing membership functions are provided in **membership.py** file. There is determine 3 types of membership function: triangular, gaussian and semicircle.

```python
import numpy as np

def determine_membership(x,abc, max, min, type='tri'):
    assert(len(abc) ==3)
    a,b,c = abc
    if x <= min or x>= max:
        return 1
    if type == 'tri':
        if a<x and x <=b:
            return (x-a)/(b-a)
        elif b<x and x < c:
            return (c-x)/(c-b)
        else:
            return 0

    elif type =='gaussian':
        return a * np.exp(- ((x-b)**2) / (2*c**2))
    elif type == 'semicircle':
        r = c - b
        y = np.sqrt(r**2 - (x-b)**2)
        return y
    else:
        print('mf type not supported')
        assert False
```

Fig. 4.3.7. Determining memberships functions

And the last file **data_loading.py** has a code which starts loading the training samples for neural network.

```
import os
import pandas as pd
import random
import numpy as  np

def shuffle_idxs(length):
    tmp =  list(range(length))
    random.shuffle(tmp)
    return tmp


def load_iris(args, path='data/iris.csv'):
    data = pd.read_csv(path)
    class_mapping = {}
    for c in data['Species'].unique():
        class_mapping[c] = len(class_mapping)
    data['class'] = [class_mapping[c] for c in data['Species']]

    targets = data['class'].to_numpy()
    data = data[['SepalLengthCm','SepalWidthCm','PetalWidthCm']].to_numpy()
    vars(args)['num_input_units'] = data.shape[1]
    vars(args)['output_units'] = len(class_mapping)

    if args.cv:
        return data, targets, np.max(data,axis=0), np.min(data,axis=0)

    else:
        train_data, train_targets = data[0::2], targets[0::2]
        test_data, test_targets = data[1::2], targets[1::2]
        return train_data, train_targets, test_data, test_targets, np.max(data,axis=0), np.min(data,axis=0)
```

Fig. 4.3.8. Data loading

4.4. A control example of lung CT processing

A chest CT scan is a grayscale 3-dimensional medical image that depicts the chest, including the heart and lungs. CT scans are used for the diagnosis and monitoring of many different conditions including cancer, fractures, and infections.

Clinical Goal

The clinical goal refers to the medical abnormality that is the focus of the study. The following figure illustrates some example abnormalities, shown as 2D axial slices through the CT volume:

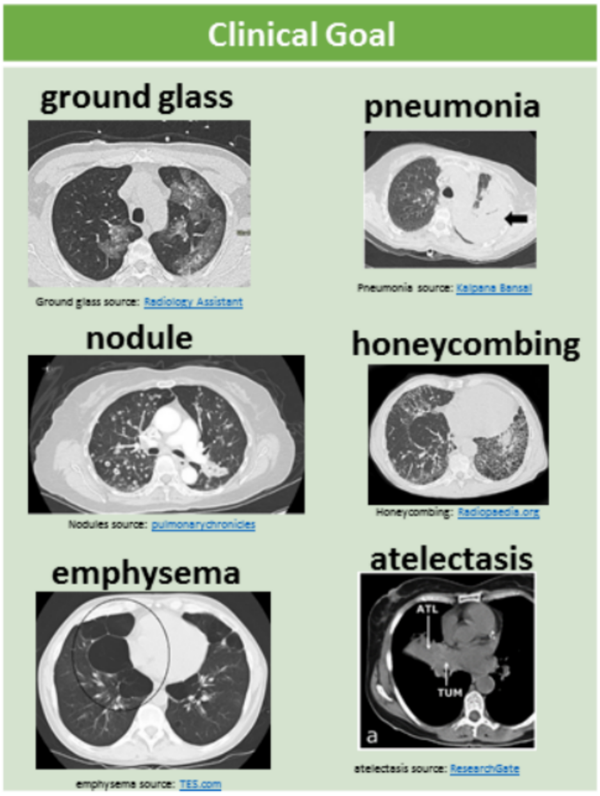


Fig. 4.4.1. Clinical goal

Many CT machine learning papers focus on lung nodules.

Other recent work has looked at pneumonia (lung infection), emphysema (a kind of lung damage that can be caused by smoking), lung cancer, or pneumothorax (air outside of the lungs rather than inside the lungs).

I have been focused on multiple abnormality prediction, in which the model predicts 83 different abnormal findings simultaneously.

3D representations include a whole CT volume which is roughly 1000 x 512 x 512 pixels, and a 3D patch which can be large (e.g. half or a quarter of a whole volume) or small (e.g. 32 x 32 x 32 pixels).

2.5D representations make use of different perpendicular planes.

- The axial plane is horizontal like a belt, the coronal plane is vertical like a headband or old-style headphones, and the sagittal plane is vertical like the plane of a bow and arrow in front of an archer.
- If we take one axial slice, one sagittal slice, and one coronal slice, and stack them up into a 3-channel image, then we have a 2.5D slice representation.
- If this is done with small patches, e.g. 32 x 32 pixels, then we have a 2.5D patch representation.

Finally, 2D representations are also used. This could be a full slice (e.g. 512 x 512), or a 2D patch (e.g. 16 x 16, 32 x 32, 48 x 48). These 2D slices or patches are usually from the axial view.

Training and validation datasets

We used two datasets in our study for training and validation. The training cohort consisted of data from 25 097 participants in NLST who received repeat annual low-dose CT screenings.[7] The validation cohort comprised data from 2350 individuals enrolled in the Pan-Canadian Early Detection of Lung Cancer study (PanCan) who received follow-up screenings, of whom 56 were removed because they received investigational screenings before the next scheduled annual screening for suspicious lung nodules.[22] Researchers on the NLST recruited ever-smokers (aged 55–74 years) from 33 screening centres in the USA who had a smoking history of at least 30 pack-years (pack-years calculated as total years smoked × cigarettes per day / 20) and had last smoked within the previous 15 years. Median follow-up in NLST was 6·5 years. Researchers on the PanCan dataset recruited current and former smokers (aged 50–75 years) from eight screening centres across Canada with a 6-year risk for lung cancer of at least 2% (ascertained by the PanCan model, a precursor to the validated $PLCO_{M2012}$ model).[23] Median follow-up in PanCan was 5·5 years. Both datasets included demographic information and radiology reports for all participants from the most recent follow-up low-dose CT scan (referred to as the S2 scan), which was done within 2 years from baseline screening, and from the most recent previous scan (referred to as the S1 scan).

To train on dataset, with best per class rule learning, is necessary execute command:

```
python main.py -v --dataset nlst
```

And the result of testing will be next:

```
Sigma:0.005, kmax:15, num_sets:5, Train: 88.89% Test: 86.67%
Sigma:0.01, kmax:15, num_sets:5, Train: 93.33% Test: 100.00%
Sigma:0.1, kmax:15, num_sets:5, Train: 83.70% Test: 80.00%
Sigma:1.0, kmax:15, num_sets:5, Train: 62.22% Test: 66.67%
Sigma:0.001, kmax:20, num_sets:5, Train: 88.89% Test: 86.67%
Sigma:0.005, kmax:20, num_sets:5, Train: 93.33% Test: 100.00%
Sigma:0.01, kmax:20, num_sets:5, Train: 93.33% Test: 100.00%
Sigma:0.1, kmax:20, num_sets:5, Train: 83.70% Test: 86.67%
```

Fig. 4.4.2. Results of testing

Conclusion. Using NEFCLASS for diagnostic and classification images with CT lunge from dataset NLST, shows the good results in classification lung cancer on the basis of a test sample. But there are quite big mistakes that can be associated with the choice of neural network activation function, as well as its possible retraining.

CHAPTER 5
OCCUPATIONAL SAFETY

5.1. Analysis of harmful and dangerous production factors

A room with the following geometric parameters was selected for the computer laboratory: width - 4 m, length - 6.25 m, area - 25 m2, ceiling height - 3.8 m. The building and premises are constructed in accordance with the requirements [10]. The computer lab is equipped with four workstations for developers. The volume of production space for developers, operators of video terminal devices per employee is 19.5 m3, the area of the premises - 6 m2 taking into account the maximum number of employees in one shift. The plan of the computer laboratory is shown in fig. 5.1.

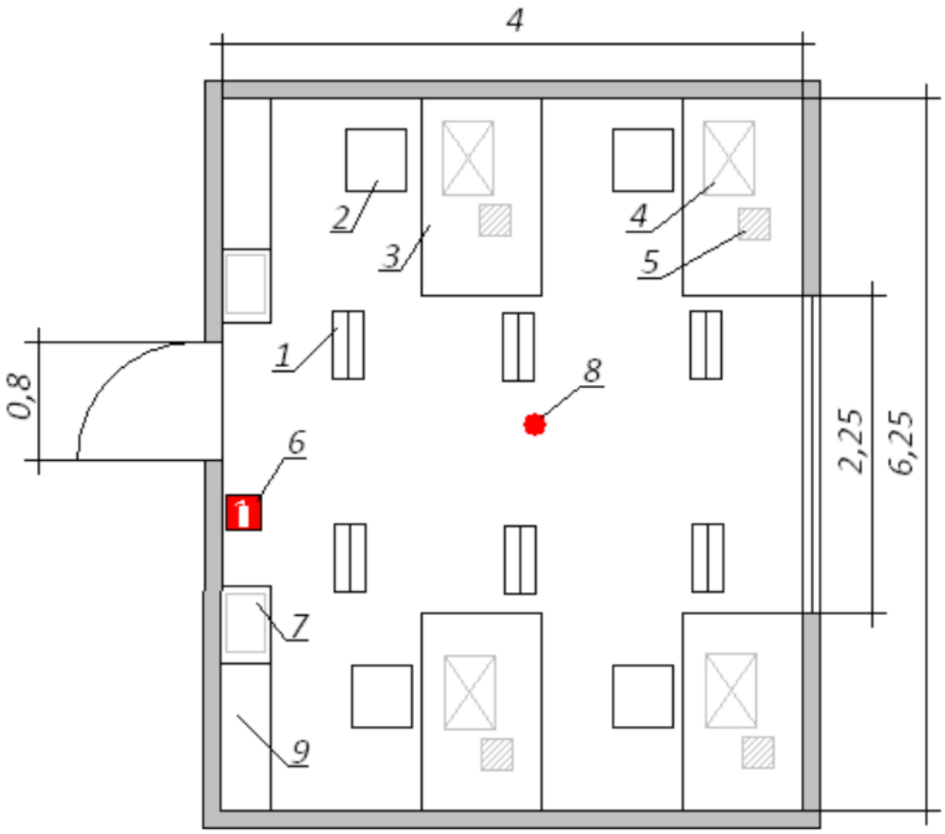


Figure 5.1 - Plan of the computer laboratory: 1 - lamp; 2 - chair; 3 - table; 4 - personal computer (PC); 5 - telephone - fax; 6 - fire extinguisher; 7 - multifunction device (BFP); 8 - fire sensors; 9 - wardrobe

The main production process is to develop algorithms, systems, technical documentation and write software that requires the use of computers. There are four workstations in the computer lab. They are all equipped with a PC with a liquid crystal display, and each location is connected to a local area network. A fax machine is additionally installed on the table. There are two MFPs in the laboratory. Six lamps are used for lighting. Each lamp contains two fluorescent lamps type LB-40-1. The windows of the computer lab are quite old. There is no special ventilation and sound insulation in the room. All equipment located in the computer lab is connected to a 220 V power supply.

Workload of the neural network operator

The work of the developer is associated with a significant visual load, which requires adequate lighting. In this room, the level of natural light is sufficient, and the level of artificial - reduced. The development engineer works with computers and other office equipment, which is a source of danger of electric shock. The work of the developer is associated with a constant stay in the room, so for comfortable working conditions it is necessary to create a proper microclimate in the computer lab.

According to standard ГОСТ 12.0.003-74 the following harmful production factors can be identified that affect the employee of this computer laboratory:

1. increased noise in the workplace;
2. increased or decreased air temperature of the working area;
3. insufficient lighting of the working area;
4. lack or absence of natural light;
5. monotony of work;

Level of artificial lightning

The computer lab has six luminaires with two LB40-1 fluorescent lamps in each. The power supply of the luminaires is a 220 V mains supply.

The actual value of the illumination of this working space is only E = 210-220 lux.The category of work performed by the developer refers to high-precision work with the assignment of the category of the III century. Therefore, the normative value of the general lighting of the working space should be E = 300-500 lux. Therefore, it is necessary to take measures to increase the illumination of the room. Lighting in the developer's workplace should be such that the employee can do his job without eye strain. The calculation of the illumination of the workplace is reduced to the choice of lighting system, determining the required number of lamps, their type and location.

According to the selected category of visual works, the allowable value of the illumination of the work surface is taken E = 400 lux.

To improve the lighting of the computer lab, LED lamps will be used, namely LITWELL LED-T8S-120, the luminous flux of which is $Ф_л$ = 1500лм.

Increased or decreased air temperature of the working area

Computers and office equipment are a source of significant heat, which can increase the temperature and reduce the relative humidity in the room. In rooms where computers are installed, certain parameters of the microclimate must be observed. Sanitary norms set the values of microclimate parameters that create comfortable conditions. (see Table 1).

Table 1 - Optimal and acceptable parameters of the microclimate

Period of the year	Microclimate parameter	Value		
		Optimal	Let's say	In fact
Cold	Air temperature in indoors	21,0-23,4°C	23,5-25,4°C	16,1-18,0°C
	Relative humidity	40-60%	75%	35%
	Air velocity	0,1м/s	до 0,1м/s	0,1м/s
Warm	Air temperature in indoors	21,0-23,4°C	23,5-25,4°C	26,7-27,4°C
	Relative humidity	40-60%	55%	55%
	Air velocity	0,1 м/s	0,2-0,1м/s	0,1м/s

To ensure comfortable conditions, both organizational methods (rational organization of work depending on the season and the day of alternation of work and rest) and technical means (ventilation, air conditioning, heating system) are used.

The value of the actual humidity in the room in the cold period - 35%, does not fall into the range of permissible values. Therefore, in the cold season in the room you need to use humidifiers, and to increase the temperature you need to install additional heating.

In the warm season to lower the temperature you need to install air conditioning.

Increased noise in the workplace

The increased noise level in the computer lab is caused by four PCs, two multifunction devices, and the hum of the starter relay. The actual value of the noise level is 88-92 dB, when the allowable sound level is ≤ GDR, namely 50 dB. Noise measurement methods and permissible sound pressure levels in octave frequency bands, equivalent sound levels at the workplace.

Noise worsens working conditions by having a harmful effect on the human body. Workers in conditions of prolonged noise experience irritability, headaches, dizziness, memory loss, fatigue, loss of appetite, ear pain, etc. Such disturbances in the work of a number of organs and systems of the human body can cause negative changes in a person's emotional state up to stressful situations. Under the influence of noise, the concentration of attention decreases, physiological functions are disturbed, fatigue appears due to increased energy expenditure and mental stress, speech commutation deteriorates. All this reduces a person's ability to work and his productivity, quality and safety.

Additional sound insulation is required to reduce the noise level. As sound-insulating materials used in the construction of floors to reduce the transmission of structural (impact) sound are mainly used mats and plates of glass and mineral fiber, soft boards of wood chips, cardboard, rubber, insulated linoleum, as well as the replacement of windows with soundproofing.

5.2. Calculation to improve the level of artificial lighting

To improve the lighting of the computer lab, LED lamps will be used, namely LITWELL LED-T8S-120, the luminous flux of which is Фл = 1500лм.

According to the selected category of visual works, the allowable value of the illumination of the work surface is taken E = 400 lux.

To calculate the illumination of the CL, we use the method of luminous flux. To determine the number of lamps determine the luminous flux incident on the surface by formula 4.1:

, where F - luminous flux, Лм

E - normalized optimal illuminance, Lk, E = 400 Lk;

S - area of the lighting room (in our case S = 25 m2);

Z - is the coefficient of minimum illumination, which characterizes the uneven lighting. Accepted at the most favorable location of the lamps, when the light flux is used to illuminate the work area most efficiently, (Z = 1.1)

k - is the stock factor, which takes into account the reduction of the luminous flux of the lamp as a result of contamination of the lamps during operation (its value is determined by the table of stock ratios for different rooms and in our case k = 1.2);

η - is the coefficient of utilization of the luminous flux from the lamp, which shows what part of the luminous flux of the lamp reaches the illuminating surface, including due to the reflection of the luminous flux from the walls, ceiling and work surface.

To determine the coefficient η you need to calculate the index of the room i by formula 4.2:

(4.2),

where S is the area of the room, S = 25 m2;

h - is the height of the suspension of lamps above the work surface, m;

A - width of the room, A = 4 m;

B is the length of the room, B = 6.25 m.

The height of the suspension is found by the formula 4.3:

(4.3)

where H - geometric height KL, H = 3,8 m;

3,2 - 0,3 - 0,9 = 2 m

According to the room index and luminous flux coefficients from the floor - 10% (0.1), from the walls - 30% (0.3) and from the ceiling - 50% (0.5), we determine the value for the LED lamp LITWELL LED-T8S-120 luminous flux utilization factor η = 0.51.

Substitute all the values in formula 5.1 to determine the luminous flux:

F = (400*1.2*25*1.1)/0,51 = 25882 Лм

Calculate the required number of lamps:

; N=25882/1500 = 18

So, for lighting we use 6 lamps, each lamp is completed with 3 lamps. Luminaires are placed in two rows, three in each row.

This room does not belong to those that need emergency lighting.

5.3. Occupational Safety Instruction

1. General labor protection requirements

1.1. The workplace for working with video terminals must be located so that the field of view of the worker does not get windows, lighting fixtures, surfaces that have the property of reflection. The surface of the desktop should not be polished. To prevent glare on the video monitor screen, especially in summer and sunny days, the video monitor screen should be positioned so that the light from the window falls on the side, preferably on the left.

1.2. The video screen of the PC monitor must be away from the user's eyes (hereinafter referred to as the operator)

1.3. at a distance of at least 500 - 700 mm. The angle of view is in the range of 10-40 degrees. The most rational is the location of the screen perpendicular to the line of sight of the operator.

1.4. The PC must be located at a distance of not less than 1 meter from the heat source.

1.5. The keyboard should be placed on a table surface or a special stand at a distance of 100-300 mm from the edge turned to the user. The angle of the keyboard panel to the horizontal surface should be between 5 and 15 degrees.

1.6. The height of the working surface of the table should be within 680-800 mm.

1.7. The chair must provide the operator with comfortable working conditions and physiological rational working posture during work. The chair must be able to adjust the height of the seat surface, the angle of the backrest and the height of the backrest.

1.8. To protect against direct sunlight, which creates glare on the video monitor screen, sun protection devices must be installed on the windows. The screen of the video monitor should be positioned so that the light from the window falls on the workplace from the side, preferably on the left.

2. Safety Requirements before starting work
    2.1. Before starting work, the employee must check the integrity of the enclosures of the system unit, video monitor, printer, keyboard.
    2.2. Check the integrity of power cables, their connection points (mains sockets, mains extensions, junction boxes, plugs).
    2.3. Prepare your workplace by removing things that may interfere with the work.
    2.4. Turn on the PC power.
    2.5. If the PC does not boot after the PC is turned on or the computer does not go into operation, the employee must notify the head or specialist of the information technology department.
    2.6. If damage or any other defects are found, notify the immediate supervisor. Do not start work without his instructions.
3. Safety Requirements during operation
    3.1. It is necessary to stably place all the components of the device on the table, including the keyboard. However, it must be possible to move the keyboard. Its location and angle should correspond to the wishes of the PC user. If the design of the keyboard does not provide space for palm support, it should be placed at a distance of at least 100 mm from the edge of the table in the optimal area of the monitor field. When working on the keyboard, you should sit straight, do not strain.
    3.2. To reduce the adverse impact on the user of devices such as "mouse" (forced posture, the need for constant quality control) should provide a free larger surface area of the table to move the "mouse" and a comfortable elbow joint.

3.3. Unauthorized conversations, annoying noises, etc. are not allowed.

3.4. Periodically, when the PC is turned off, dust should be removed from the surfaces of the equipment with a cotton swab slightly moistened with a soap solution. The screen and protective screen are wiped with an alcohol swab.

3.5. FORBIDDEN:

3.5.1. self-repair equipment in which the tube and other elements may be under high voltage (up to 25 kV.)

3.5.2. put any things on the PC hardware, sandwiches and drinks on or next to the keyboard. This can disable it;

3.5.3. cover the ventilation holes in the equipment, it can lead to overheating and failure.

3.6. To reduce the negative impact on the health of employees of various risk factors associated with work on the PC, there are additional regulated breaks for rest of PC users:

3.6.1. after each time of continuous work - 10 minutes;

3.6.2. every 2 hours - 15 minutes.

3.7. In order to reduce the negative impact of monotony, it is advisable to alternate operations of text input and data entry (change the content and pace of work), etc.

4. Safety Requirements after work

4.1. Finish and save the PC files that were in the works. Perform all steps to properly shut down the operating system.

4.2. Turn off the printer and other peripherals, turn off the system unit. If there is an uninterruptible power supply (UPS), turn off its power.

4.3. Turn off the PC with the “POWER” button and unplug the power cord.

4.4. Cover the keyboard with a lid to prevent dust from entering it.

4.5. Bring order to the workplace.

5. Safety Requirements at emergency situations

5.1. If the PC smells burnt or the metal parts of the PC are exposed to electric shock, unplug the PC immediately and notify your supervisor.

5.2. In case of fire, immediately start extinguishing with available fire extinguishing means and notify by phone 101 (city fire department) and the head of the DPD of the enterprise. Remember that extinguishing electrical installations should be carbon dioxide fire extinguishers, dry sand to avoid electric shock.

5.3. In case of injury, stop work, provide first aid, call an ambulance by phone 103, if necessary, take to a hospital.

CHAPTER 6
ENVIRONMENT PROTECTION

6.1. The impact of neural networks on the environment

The field of artificial intelligence is often compared to the oil industry: after extraction and refining, data, like oil, can become a very profitable commodity. However, it is now clear that this metaphor is expanding. Like fossil fuels, the process of deep learning has a great impact on the environment. Researchers at the University of Massachusetts warn about the shadow side of artificial intelligence in a new work: the amount of computation required to train the model requires a huge amount of energy and causes the release of carbon dioxide into the atmosphere. The figures given by the authors show that training one model costs humanity more than operating five cars.

It turned out that this process could emit more than 626,000 pounds (about 300,000 kg) in carbon dioxide equivalent, which is almost five times higher than the emissions of a typical car in five years (including the production of the car itself).

Personal computer as a source of pollution

A computer is a source of electromagnetic radiation. It is believed that electromagnetic radiation can cause disorders of the nervous system, decreased immunity, disorders of the cardiovascular system and abnormalities during pregnancy.

The biological response of a person is influenced by such parameters of computer electromagnetic fields as the intensity and frequency of radiation, the duration of irradiation and signal modulation, the frequency spectrum and the frequency of action.

Computer monitors are a source of X-rays, beta and gamma radiation. X-rays are present only when the monitor is running. The X-ray spectrum is safe with a set of monoenergetic lines. The maximum energy of the spectrum is ~ 20 KV. Beta and gamma radiation are present when the monitor is on and off. The source of these radiations is the radioactive decay of nuclei of the uranium and thorium families, as well as potassium-40 nuclei. The spectral composition of gamma radiation mainly consists of a set of monoenergetic lines. The beta radiation of the monitor is determined mainly by the radioactive decay of potassium-40 nuclei; the spectral composition of beta radiation is continuous, and its maximum energy is ~ 1.3 meV. Under certain conditions, these ionizing radiations can harm human health, including clouding of the lens of the eye. To reduce the harmful effects of ionizing radiation, the anode voltage was reduced in the monitors, and lead was added to the monitor glass.

During operation, the computer creates an electrostatic field around itself, which deionizes the environment, and when the board and the monitor case are heated, they emit harmful substances into the air. All this makes the air dry, weakly ionized, with a specific odor and generally "difficult" to breathe. Naturally, such air can not be useful for the body and can lead to allergic diseases, respiratory diseases and other disorders.

Since the most harmful are monitors and system units, we will consider them in more detail.

There are three types of monitors:

1. Monitors based on a cathode ray tube;

2. Liquid crystal;

3. Plasma.

The main source of negative impact of the computer on the environment and user health is monitors based on a cathode ray tube.

The degree of human hazard of ionizing radiation emitted by computer monitors depends on the levels of ionizing radiation that enter the eyes of users.

X-ray exposure dose rate at a distance of 0.05 m from the screen and housing of the video terminal at any position of control devices in accordance with the Radiation Safety Standards of Ukraine (NRBU-97), approved by the State Sanitary Doctor of the Ministry of Health of Ukraine from 18.08.97 №58, should not exceed A / kg, which corresponds to an equivalent dose of 0.1 mber / h (100 μR / h). The level of gamma radiation depends on the concentrations of natural radionuclides in the monitor glass, which for potassium-40 is 3-10%, for thorium and uranium. Based on this, we can show that at a distance of 5 cm from the monitor screen, the dose rate of gamma radiation is insignificant (~ 0.03-0.1 μR / hour) and is 0.5% of the background dose rate. Beta radiation can be easily measured with a beta counter. Such measurements show that at a distance of 5 cm from the monitor screen, the beta radiation flux density can be 0.2-0.5 parts/s.

Scintillation spectrometers with thin crystals of NA1 (T1) or Cs1 (T1) and with a sufficiently large surface are usually used to measure X-rays. The results obtained with the help of such counters show that the maximum dose rate of X-rays at a distance of 5 cm from the screen of the comparison monitor with the background and does not exceed 5-15 μR / hour.

Based on this, the power of the equivalent radiation dose for the adverse event, when the eyes of the computer operator are located at a distance of 5 cm from the monitor screen, is 0.3-0.4 μSv / hour. This result indicates the radiation safety of computer monitors, as the accumulated by the lens of the eye annual equivalent dose (~ 0.7 mSv) is 20 times less than the allowable NRBU-97 value.

Effects of electromagnetic oscillations on human health

Disorders caused by the action of electromagnetic fields on humans are manifested by higher nervous activity and bioelectrical activity of the brain. One of the reasons that contribute to the development of adverse factors in humans is the information disintegration in the brain system with subsequent violations in other functional systems. Thus, it has been established that the human endocrine, immune and reproductive systems are too sensitive to electromagnetic fields (EMF). Periodic action of EMF can lead to persistent changes in hormonal status, adversely affect genetic structures.

For the first time, a significant comprehensive study of the possible adverse effects of electromagnetic fields on the health of users was conducted in 1994 in Canada.

According to the generalized data, in computer workers from 2 to 6 hours a day functional disorders of the central nervous system occur more often on average 4-6 times than in control groups, diseases of the cardiovascular system - 2 times more often. Diseases of the upper respiratory tract - 1.9 times more often, diseases of the musculoskeletal system - 3.1 times more often. As the duration of work on the computer increases, the ratio of healthy and sick among users increases sharply.

Studies of the functional state of computer users conducted in 1996 at the Center for Electromagnetic Safety showed that even with short-term operation of

45 minutes, the user's body under the influence of electromagnetic radiation of the monitor undergoes significant changes in hormonal status and specific changes in brain biocurrents. These effects are especially bright and persistent in women. It is noticed that only about 20% of people have a negative reaction to the functional state of the body when working on a PC for more than 1 hour. In 80% it happens earlier. Therefore, methods and means should be developed to reduce the negative impact of electromagnetic fields on human health.

Article 3. Basic principles of environmental protection

The main principles of environmental protection are:

a) the priority of environmental safety requirements, the obligation to comply with environmental standards, standards and limits on the use of natural resources in the implementation of economic, managerial and other activities;

b) guaranteeing an environmentally safe environment for human life and health;

c) precautionary nature of measures to protect the environment;

d) greening of material production on the basis of comprehensive solutions in matters of environmental protection, use and reproduction of renewable natural resources, widespread introduction of new technologies;

e) preservation of spatial and species diversity and integrity of natural objects and complexes;

f) scientifically substantiated coordination of ecological, economic and social interests of society on the basis of a combination of interdisciplinary knowledge of ecological, social, natural and technical sciences and forecasting of the state of the natural environment;

g) mandatory environmental impact assessment;

{Item "is" of Article 3 as amended by Laws № 3038-VI of February 17, 2011, № 2059-VIII of May 23, 2017}

h) publicity and democracy in decision-making, the implementation of which affects the state of the environment, the formation of the population's ecological worldview;

i) scientifically substantiated rationing of the impact of economic and other activities on the environment;

j) gratuitousness of general and payment of special use of natural resources for economic activity;

k) compensation for damage caused by violation of environmental legislation;

{Paragraphs "i" of Article 3 as amended by Law № 2756-VI of December 2, 2010}

l) addressing issues of environmental protection and use of natural resources, taking into account the degree of anthropogenic variability of territories, the cumulative effect of factors that adversely affect the ecological situation;

m) a combination of incentives and responsibilities for environmental protection;

n) solving problems of environmental protection on the basis of broad interstate cooperation;

o) establishment of an environmental tax, rent for special use of water, rent for special use of forest resources, rent for subsoil use in accordance with the Tax Code of Ukraine;

{Article 3 is supplemented by item "l" in accordance with Law № 2756-VI of 02.12.2010; with changes made in accordance with the Law № 71-VIII of 28.12.2014}

p) taking into account the results of strategic environmental assessment.

{Article 3 is supplemented by item "m" in accordance with Law № 2354-VIII of March 20, 2018}

{Article 3 as amended by Law № 186/98-BP of 05.03.98}

6.2. Calculation of the impact of neural networks on the environment

The paper especially considers the process of learning a model for natural language processing (NLP), a subfield of artificial intelligence, which is engaged in learning machines for working with human language. Over the past two years, the NLP community has reached several important milestones in machine translation, proposal completion, and other standard evaluation tasks.

The infamous OpenAI GPT-2 model, as an example, has succeeded in writing compelling fake news notes.

But such advances required the training of ever-larger models on stretched datasets from sentences extracted from the Internet. This computational approach is expensive and very energy intensive.

The researchers looked at four models in the area responsible for the biggest performance jumps: Transformer, ELMo, BERT and GPT-2.

They trained each of them on a single GPU during the day to measure power consumption.

They then took the number of training hours indicated in the initial documents of the model to calculate the total energy consumed throughout the training process. This amount has been translated into the equivalent of pounds of carbon dioxide, which corresponds to the energy consumption structure of AWS from Amazon, the largest provider of cloud services.

It turned out that the computational and environmental costs of training increased in proportion to the size of the model, and then increased many times as the final accuracy of the model was adjusted. Finding a neural architecture that attempts to optimize the model by gradually changing the structure of the neural network through trial and error is extremely costly with little performance gain. Without it, the most expensive model BERT left a carbon footprint of 1,400 pounds (635 kg), which is close to the Trans-American round trip.

Moreover, these figures should only be considered as baselines.

In total, scientists estimate that the process of creating and testing a final model worthy of publication required the training of 4,789 models in six months. In terms of CO_2 equivalent, it is about 35,000 kg.

The significance of these numbers is enormous, especially given the current trends in artificial intelligence research.

In general, artificial intelligence research is neglected because large neural networks are found to be useful for a variety of tasks, and companies with unlimited computing resources will use them to gain a competitive advantage. But for the climate it will not be very good.

6.3. Calculation of the sanitary protection zone for the source of electromagnetic radiation

The effect of electromagnetic radiation is determined by the field strength, the duration of irradiation and the wavelength. The greater the field strength, the shorter the wavelength and the longer the irradiation time, the stronger the effect.

The creation of a sanitary protection zone (SPZ) around the emitter is one of the measures taken to eliminate the negative impact of EMF. For a powerful energy source, such a zone may consist of a strict regime zone, which is fenced and protected and the stay of people in which it is prohibited, and a restricted use zone, in which it is allowed to place warehouses, workshops, garages and other facilities hours per day. The size of the SPZ is calculated separately for each object, the main condition of such calculation is that at the boundary of the zone the parameters of the field should not exceed their MAL (maximum allowable level).

Calculations are specified by means of instrumental control measurements.

MAL EMF for populated areas have the following values:

Frequency subband	Field frequency	Wavelength	MAL of field
Low	30-300 kGz	10-1 km	25 В/м
Medium	0,3-3 MGz	1-0,1 km	15 В/м
Hight	3-30 MGz	100-10 m	10 В/м
Very hight	30-300 MGz	10-1 m	3 В/м
Ultra high	0,3-3 GGz	100-10 sm	10 мкВт/sm^2
Extremal hight	3-300 GGz	10-0,1 sm	10 мкВт/sm^2

In the low frequency range - very high MAL EMFs are determined by the field strength, and in the ultra high and ultrahigh frequencies are determined by the energy flux density (EFD), which means the amount of energy passing through each cm2 of body surface perpendicular to the propagation of radiation.

The main measure of the EMF is the creation of SPZ around the transmitter antennas, the size of the zones is calculated so that at the boundary of the zone the field strength should not exceed its MAL.

Calculate the radius of the sanitary protection zone according to the following formula:

$$R = \sqrt{\frac{P_{cp} \cdot G}{4\pi\sigma}} = 2,27$$

where P_{cp} is the average radiation power of the source, W;

G is the gain of the antenna, equal to 10 · 103;

σ is the energy flux density, W / m2.

The average radiation power is calculated by the formula:

$$P_{cp} = P_i \cdot \tau_i \cdot F = 65$$

where P_i is the pulsed power of the source radiation, which is equal to 50 kW;

τ_i is the pulse duration (1.3 μs);

F is the pulse repetition frequency (1000 Hz).

The boundaries of the sanitary protection zone are determined by the allowable values of energy flux density (EFD):

a) for service personnel involved in the operation of the radiation source - 10 W / m2;

b) for personnel not connected with the operation of the radiation source - 5 W / m2;

c) for outsiders (population) - 0.15 W / m2.

So, after the calculation, we see that in this case the radius of the sanitary protection zone will be 2.27 m.

CONCLUSIONS

In this paper, the influence of neural networks on medical image processing and image classification was investigated. Considering the relevance of this topic, neuro fuzzy classifiers such as TSK, Wang-Mengel network, ANFIS were considered, but special attention was paid to the NEFCLASS classifier.

The general and most common methods of medical image processing, namely filtering and segmentation, were considered. These methods allow you to get rid of various noises and factors that can adversely affect the classification process.

The structural-parametric synthesis of the neurofuzzy NEFCLASS classifier was developed in the work. The results obtained from the training and test sample for CT of the lungs showed quite good results for this neural network. It is worth noting that the use of fuzzy logic to classify diseases based on the obtained images is a relevant topic and has the potential to improve the results of diagnosing patients without human intervention, thus simplifying the work of doctors and other health professionals.

References

1. Горюнова В.В. [и др.] Особенности проектирования интегрированных медицинских систем на основе концептуальных спецификаций // Фундаментальные исследования. – 2013 —№11–9 – С. 67–73..
2. Nauk, D., Kruse, R. (1998) How the learning of rule weights affects the interpretability of fuzzy systems. In Proc. IEEE International Conference on Fuzzy Systems 1998 (FUZZ-IEEE’98), Anchorage, AK, May 4-9, pp 1235-1240
3. Doyle, S. Agner, A. Madabhushi, M. Feldman, and J. Tomaszewski, “Automated grading of breast cancer histopathology using spectral clustering with textural and architectural image features,” in Proceedings of the 5th IEEE International Symposium on Biomedical Imaging (ISBI): From Nano to Macro, vol. 61. IEEE, May 2008, pp. 496-499.
4. Y. Le Cun, Y. Bengio, and G. Hinton, “Deep learning,” Nature, vol. 521, pp. 436-444, 2015.
5. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in Advances in NeuralInformation Processing Systems 25, 2012, pp. 1097-1105.
6. Jianpeng Zhang, Yong Xia, Qi Wu, Yutong Xie. Classification of Medical Images and Illustrations in the Biomedical Literature Using Synergic Deep Learning, Xiv:1706.09092v1[cs. CV] 28 Jun 2017.
7. Nilanjan Dey. Classification Technologies Techniques for Medical ImageAnalysis and Computer-aided Diagnosis, Volume 14, 1st Edition.-Academic Press, 201.-218 p.

8. Yin, Xiao-Xia, Hadjiloucas, Sillas, Zhang, Yanchun. Pattern Classification of Medical Images: Computer Aided Diagnosis Springer International. Springer International Publishing. 2017.-218 p. DOI. 10.1007/978-3-319-57027-3.Hardcover ISBN.978-3-319-57026-6.
9. Yuriy Zaychenko, Aghaei Agh Ghamish Ovi Nafas. Medical Images Classification and Diagnostics Using Fuzzy Neural Networks. American Journal of Neural Networks and Applications. Vol. 5, No. 2, 2019, pp. 45-50. doi: 10.11648/j.ajnna.20190502.11
10. K. Malyshevska. The analysis of neural networks' performance for medical image classification/K. Malyshevska//International Journal "Information Content and Processing", Volume 1, Number 2, 2014.-C. 194-199.
11. K. Malyshevska. Analysis of neural networks application for diagnostics of uterus cancer using multispectral images. System research and information technologies.-2010-No2-pp. 64-71. (rus)
12. Detlef Nauck and Rudolf Kruse. Generating classification rules with the neuro-fuzzy system NEFCLASS. In Proc. Biennial Conf. of the Norght American Fuzzy Information Processing Society (NAFIPS'96), Berkeley, 1996.
13. Detlef Nauck and Rudolf Kruse. New learning strategies for NEFCLASS. In Proc. Seventh International Fuzzy Systems Association World Congress IFSA'97, Vol. IV, pp. 50-55, Academia Prague, 1997.
14. Zaychenko Yu. P., Sevaee Fatma, Matsak A. V. Fuzzy neural networks for economic data classification//Vestnik of National Technical University of Ukraine "KPI", section "Informatic, control and c omputer engineering. Vol. 42.-2004.-pp. 121-133. (rus).

15. Chakraborty D., Pal N.R. A neuro-fuzzy scheme for simultaneous feature selection and fuzzy rule-based classification. IEEE Transactions on Neural Networks, 2004, vol. 15, no. 1, pp. 110– 123. 15
16. Jang J.S.R., Sun C.T., Mizutani E. Neuro-fuzzy and soft computing. Upper Saddle River, NJ, Prentice Hall, 1997, pp. 110–123.
17. Grigor'eva K.V., comp. Methods for solving the problem of minimizing a quadratic function. Convergence problems. St. Petersburg State University of Architecture and Civil Engineering. St. Petersburg, 2009. 36 p.