

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ
«АЛМАТЫ ЭНЕРГЕТИКА ЖӘНЕ БАЙЛАНЫС УНИВЕРСИТЕТІ»
коммерциялық емес акционерлік қоғамы
IT-инжиниринг кафедрасы

ҚОРҒАУҒА ЖІБЕРІЛДІ

Кафедра меңгерушісі

PhD, доцент

Т.С. Картбаев

« » 2018 ж.

ДИПЛОМДЫҚ ЖОБА

Тақырыбы: Жасанды нейрондық желілер әзірлеу

Мамандығы 5B070300 – «Ақпараттық жүйелер»

Орындаған Қожабеков М.Т. Тобы ИСК-14-1
Ғылыми жетекші аға оқытушы Абсатарова Б.Р.

Кеңесшілер:

Экономикалық бөлім: э.ғ.к., доцент А.И. Бекишева
«21» 05 2018 ж.

Өміртіршілік қауіпсіздігі: аға оқытушы Н.С. Бекмуратова
«21» 05 2018 ж.

Есептеу техникасын қолдану: аға оқытушы Ж.С. Айткулов
«23» 05 2018 ж.

Норма бақылаушы: аға оқытушы К. Мукапил
«31» 05 2018 ж.

Сын-пікір беруші: т.ғ.к, ассис. профессор С.Т Аманжолова
« » 2018 ж.

Алматы 2018

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ
«АЛМАТЫ ЭНЕРГЕТИКА ЖӘНЕ БАЙЛАНЫС УНИВЕРСИТЕТІ»
коммерциялық емес акционерлік қоғамы

Басқару жүйелері және ақпараттық технологиялар институты

IT-инжиниринг кафедрасы

Мамандығы 5В070300 – «Ақпараттық жүйелер»

Дипломдық жобаны орындауға берілген
ТАПСЫРМА

Білім алушы Қожабеков Мадияр Талғатұлы

Жобаның тақырыбы: Жасанды нейрондық желілер әзірлеу

2017 жылғы «23» қазан № 155 университет бұйрығымен бекітілген.

Аяқталған жобаны тапсыру мерзімі: «01» маусым 2018 ж.

Дипломдық жобаның бастапқы мәліметтері (зерттеу (жоба) нәтижелерінің талап етілген параметрлері мен объектінің бастапқы мәліметтері): Microsoft Visual Studio 2015 ортасындағы C# программалау тілін қолданып, бейнелерді таныйтын нейрондық желі құру.

Дипломдық жобада қарастырылған мәселелер тізімі немесе диплом жобаның қысқаша мазмұны:

- а) нейрондық желілер түсінігі;
- б) жобалау бөлімі;
- в) желілік компоненттер;
- г) экономикалық бөлім;
- д) өміртіршілік қауіпсіздігі;
- е) А қосымшасы. Программа мәтіні.

Графикалық материалдар тізімі (міндетті сызбалар дәл көрсетілуі тиіс):
7 кесте, 46 сурет ұсынылған.

Ұсынылатын негізгі әдебиеттер:

1 Головкин В.А. Нейроинтеллект: Теория и применения. Книга 1. Организация и обучение нейронных сетей с прямыми и обратными связями - Брест:БПИ, 2014. 160 б.

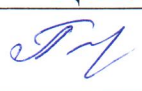



2 Головкин В.А. Нейроинтеллект: Теория и применения. Книга 2. Самоорганизация, отказоустойчивость и применение нейронных сетей - Брест:БПИ, 2014. 128 б.

3 Уоссермен Ф. Нейрокомпьютерная техника: Теория и практика. 2014. 84 б.

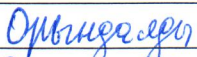


4 Самаль Д.И., Старовойтов В.В. - Подходы и методы распознавания людей по фотопортретам. - Минск, ИТК НАНБ, 2013. 4 б.

5 Нейрокомпьютерная техника: теория и практика. Перевод на русский язык. Ю. А. Зуев, В. А. Точенов, 2015


Дипломдық жобаның бөлімдеріне қатысты белгіленген кеңестер

Бөлімдер	Кеңесшілер	Мерзімі	Қолы
Экономикалық бөлім	Бекишева А.И.	05.05.2018- 21.05.2018	
Өміртіршілік қауіпсіздігі	Бекмуратова Н.С.	05.05.2018- 21.05.2018	
Программалық қамтама	Айтқулов Ж.С.	05.05.2018- 23.05.2018	
Норма бақылау	Мукапил К.	05.05.2018- 23.05.2018	

Дипломдық жобаны дайындау
КЕСТЕСІ

Бөлімдер атауы, қарастырылатын мәселелер тізімі	Ғылыми жетекшіге ұсыну мерзімдері	Ескерту
Нейрондық желілер түсінігі	15.11.2017-25.12.2017	
Жобалау бөлімі	26.12.2017-12.02.2018	
Желілік компоненттер	12.02.2018-21.04.2018	

Тапсырманың берілген уақыты «25» қазан 2017 ж.

Кафедра меңгерушісі  Т.С. Картбаев

Жобаның ғылыми жетекшісі  Б.Р. Абсатарова

Тапсырманы орындауға алған білім алушы  М.Т. Қожабеков

Аңдатпа

Дипломдық жобаны құру алдында нейрондық желілер туралы мәліметтер жиналды. Экспоненттік функция арқылы есептеулер жүргізіліп, нейрондық желінің негізі ретінде массив алынды. Windows жүйесінде 50 программа тандалынды. Программалау тілі ретінде Microsoft Visual Studio 2015 программалау ортасындағы C# тілі негізге алынды. Жобада кері қатені табу әдісімен нейрондық желі құрылды. Негізгі батырмалы форма, класстар жасалынды. Программаны реализациялау және тестілеу Windows 10-да жүргізілді.

Мәндер ретінде Windows жүйесіндегі программалар және сыртқы программалардың толық жолы көрсетілген, сол арқылы тану жүреді. Жаңа мән қолданушымен оқытылады. Программа Windows жүйелерін оңтайлы қолдану үшін қолданылады.

Аннотация

Перед созданием дипломного проекта, были собраны материалы на тему нейронной сети. Расчеты выполнялись экспоненциальной функцией, а в качестве основы нейронной сети был выбран массив. В качестве значения в Windows было выбрано 50 программ. Язык программирования был основан на C#, а в качестве среды программирования был использован Microsoft Visual Studio 2015. Нейронная сеть основана на обратном обнаружении ошибки. Созданы базовые условные формы, классы. Программа была реализована и протестирована на Windows 10.

В качестве значения берется абсолютная директория программ в Windows и внешних программ. С помощью этих значений и образов делается сравнения. Новые значения вводятся пользователем. Программа предназначена для удобного использования системы Windows.

Annotation

The first step of creation this diploma project was to collect all the materials on the "neural network" theme. The calculations made by an expression function, and the array was chosen as the basis of the neural network. There were selected 50 programs in Windows as values. The programming language based on C#, and Microsoft Visual Studio 2015 used as the programming environment.

The neural network based on the reverse error detection. Also basic conditional forms and classes were created. The program was implemented and tested on Windows 10. As a value there have been taken the absolute directory of programs in Windows and external programs. Then with the help of these meanings and images the program makes comparisons. The user enters new values. The program is designed for convenient use of Windows system.

Мазмұны

	Кіріспе	8
1	Нейрондық желілер түсінігі	26
1.1	Нейрондық желіге шолу	12
1.2	Жасанды нейрондық желінің құрылымы	17
1.3	Нейрондық желідегі синаптикалық байланыстар	22
1.4	Нейрондық желілер	25
1.5	Тесттік сұрақтардың маңыздылығын бағалау	26
2	Жобалау бөлімі	47
2.1	Мәселені қалыптастыру	27
2.2	Аналитикалық шешім	29
2.3	Нейрондық желілер идеологиясына шешім шығару	31
2.4	Алгоритмдік бөлім	37
2.5	Селективті Липшиц тұрақтығын пайдалана отырып, нейрондық желінің ақпараттық сыйымдылығын бағалау	38
2.6	Тесттік сұрақтардың маңыздылығымен желіні сәйкестендіру	40
2.7	Контрастты желілермен эксперименттердің нәтижелері	47
3	Нейрондық желілік компоненттер	66
3.1	Нейрондық желінің жалпы элементі	48
3.2	Нейрондық желінің кіруі	49
3.3	Нейрондық желіні шығару	50
3.4	Тривиальды жалдаушы	50
3.5	Нейрондық желілік ағын	51
3.6	Композициялық бір жарым қабатты желі	60
3.7	Нейрондық желіде құрылған бағдарламаның алгоритмі	66
4	Экономикалық бөлім	75
4.1	Экономикалық қызметтің пайдалылығы	68
4.2	Бағдарламалық қамтамасыздандыруды әзірлеуге қатысты ұсыныстардың техникалық-экономикалық негіздемесі	68
4.3	ПП дамуының күрделілігін анықтау	71
4.4	Бағдарламалық жасақтама әзірлеу шығындарын есептеу	74
4.5	ПП-ның ықтимал (шарттық) бағасын анықтау	75
4.6	ПП-ның әлеуметтік-экономикалық көрсеткіштерін бағалау	75
5	Өміртіршілік қауіпсіздігі	82
5.1	Құрылыстың желдету жүйесі	76
5.2	Серверлік бөлмелер үшін климаттық технологияның негізгі түрлері	78
5.3	Серверлік бөлменің желдету жүйесін жобалау және есептеу	82
	Қорытынды	83
	Әдебиеттер тізімі	85
	А қосымшасы. Программа мәтіні	93

Кіріспе

Нейрондар адамдардағы жүйке жүйесін ұйымдастырылу құрылысын түсіндіреді;

Жүйкенің жасушалық бөлігі – жүйкелік ұлпасындағы басты морфологты және функцияналдық бөлігі.

Жасандық нейронды желілер – мәліметтерді тану программалары.

Нейрондық желіде қатар өңдеу мен нейрондардың берік байланыстар болады. Соған орай есептеуіш құралдар шектік байланыстары болмайды.

Нейрон желілері қолдан істелген интеллект, инженерлік жүйелері сезім және өндірістік процестерді бақылау проблемаларын шешу үшін пайдаланылады. Адаптивті Хопфилд сақинасы кедергіге төзімді байланыс жүйелерін жасау үшін пайдаланылады. эксперименттік әзірлемелер (мысалы, Siemens компаниясы зертханаларында) аппараттық өнегесі бола алады.

Үйрету барысында қос функцияны қолданып, алынған параметрлерде қателік функциясының градиентін өте жылдам есептеу үшін алгоритм қолданылады. Қате функциясының градиентін жоғары параллельде (сәйкес аппараттық құралдардың бар болған кезде) алуына мүмкіндік беретін әдістердің болуы нейрон желілерін үйрету үшін көп өлшемді функцияларды шартсыз оңтайландыруға арналған кеңейтілген аппаратты пайдалануға мүмкіндік береді.

Жобаның мақсаты:

- нейрондық желі құрып, символды тану программасын құру;
- нейрондық желіні оқыту режимін асыру;
- символдарға берілген мәндерді пайдаланып процесстерді жүзеге

асыру.

Бұл нысаналар орындалуы үшін:

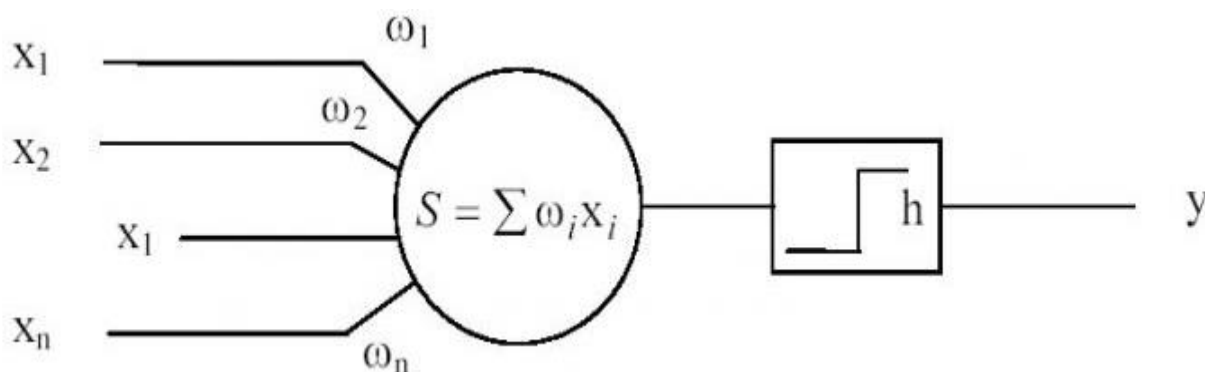
- нейрондық желілердің қолданылу принципін бейнені тануда бағалау;
- нейрондық желі сипаттама шындықты жасауға жол бермейтін ұсынымдар берсе, интуитивті тәсілдің қолданылуын бағалау;
- нейроимитатордың бағдарламалық жасақтама моделі және осы класс проблемаларын шешу үшін қолайлы бағдарламалық жасақтаманың логикалық құрылымын жасау;
- бұл программаны реализациялау үшін кері таралу қатесі әдісін пайдалану.

1 Нейрондық желілер түсінігі

1.1 Нейрондық желіге шолу

Нейрондық желілердегі білім нейротрансмиттердің (немесе жай нейрондардың) элементтерінің жиынтығында және олардың арасында сақталады.

МакКаллокПиттс нейроны әлгі уақытқа дейін де бөлек нейрондарды түсіндіру үшін қолданылып жүрген ресми моделі төменде 1.1-суретте көрсетілген.



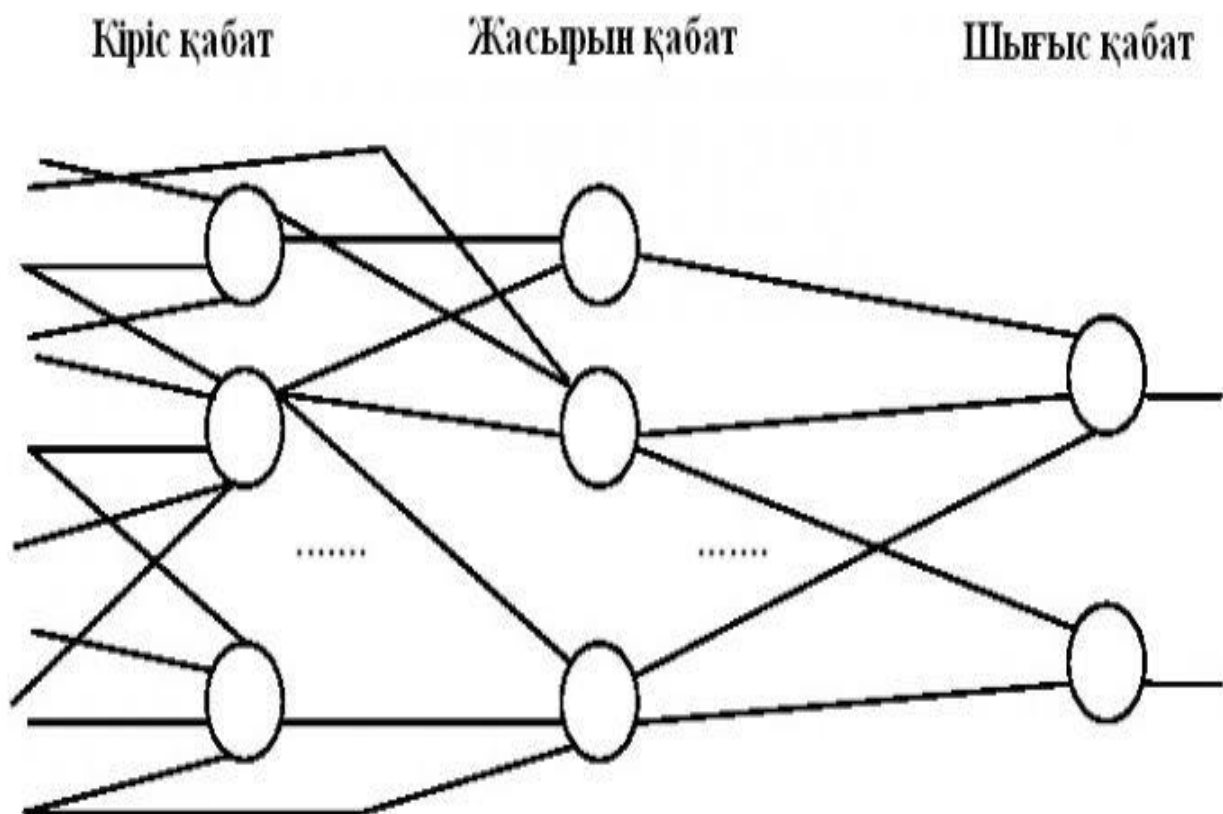
1.1-сурет – МакКаллокПиттс нейронның ресми моделі

мұндағы, x_i – нейрондағы i кірісіндегі сигналы;
 w_i – нейрондағы i кірісіндегі салмағы;
 y – нейрон шығысы;
 h – нейрон жұмыс істеу шегі.

Модельде нейронды енгізу кезінде сигналдар шекті шектімен салыстырылады және сигнал шығыс мәнінен асып кетсе, шығу кезінде шығарылады. Нейрондардың қазіргі заманғы модельдерінде табалдырық функциясы көбінесе трансфер функциясы немесе нейрондық активтендіру функциясы деп аталатын сызықтық емес $y = \langle f \rangle$ (S) функциямен ауыстырылады.

Кейде нейрондық нейрондар пайда болады, олар кейбір кірістер мен шығыстарға байланысты. Әдетте үш түрлі тораптар (кіріс) (кірістіру қабаты немесе кіріс қабаты), шығу (сыртқы қабат немесе шығыс қабаты) және жасырылған қабаттар (Hiddenlayers) бөлінеді. (1.2-суретті қараңыз)[1].

Жасырын қабатта басты таңдау жүргізіледі. Ол 1 немесе 0-ды таңдайды. Егер ол бүтін мән болмаса, яғни 0,5 жоғары немесе кем болса, онда ол мәнлі жуықтайды және шығыс қабатына жібереді. Бұндай нейрондардан нейрондық желі жиналып, олар бір нейронға соңғы қорытынды мәнді жібереді.



1.2-сурет – Тура байланыс нейрондық желісі

Нейронды желінің жұмыс істеуі екі этаптан тұрады: желіні кіру ақпаратқа (кіру векторы) «дұрыс» немесе барабар жауап беруге үйрету және үйретілген желіні кіру векторларын анықтау үшін қолданады. Соңғы этапты көбінесе тестілеу дейді. Басқаша айтқанда, желі кіру векторларды анықтауға, яғни кіру векторлардың танылған класстарына сай келетін шығу векторларды жасауға үйретіледі. Бұл кезеңде шығыс векторларына кіру векторларының сәйкестігі туралы ақпарат нейрондардың синапстар мен шектерінің салмағымен сақталады. Кәдімгі тәсіл (конгоция) контекстінде ол синапстарда сақталады. Кейде кіріс және шығыс векторларын кіріс векторы ретінде сәйкестігін түсінеміз және осы вектордың барлық деңгейлерін желіні үйрену және тестілеу үшін қолдануға болмайды. Нейрондық желілердің кейбір үлгілерінде (Hopfield моделі) кіріс және шығыс сигналдары өшірілмейді және сәйкесінше, желінің жұмысында олардың рөлдерімен алмасуға болады.

Семантикалық нейрондық желілер. 1957 жылы Джон фон Нейманның нейрондық желісінің архитектурасы басқаша рецептрондық нейрондық желі болып табылады. Оның қасиеттері үшін маңызды болып табылатын жұмысы:

- бастапқы конъюнктурада, ажыратылуда және инверсияда қолданылатын кез келген құрылғының кіріс сигналдарын қарапайым нейрондық өңдеу;
- нейрондар орындалатын функцияны динамикалық түрде өзгерте алады;
- бір нейрондық әртүрлі функцияларды түрлі уақытта атқара алады;
- бір рет - тек қана орындалатын функциялардың бірінде;

- нейрон деректерді басқа нейрондармен динамикалық өзгерістермен алмастыра алады;

- нейрондық желінің бір бөлігі желідегі басқа бөліктердің жай-күйін талдай алады;

- нейрондық желінің бөлігі желілік топологияның қалған бөлігін өзгерте алады.

Математикалық функциялардың толық жиынтығының күрделілігі оларды есептеу кезінде ғана көрінеді. Фон Нейман өз жұмысында мұндай тораптық торап Тьюринг машинасына тең екендігін көрсетеді [2].

Бұл ретте, тек деңгейі metaötilim, осындай нейрондық желінің нәтижесі машиналар Тьюринг тең ретінде тікелей таспа үшін ақырлы автомат және бағдарламалық қамтамасыз ету арқылы жүзеге асырылады.

Нейман нейрондық желінің маңызды ерекшелігі болып табылады. Ең бастысы - өзін-өзі рефлексия және өзін-өзі өзгерту мүмкіндігі.

Айталық, кез келген желілік құрылым. Желінің нейрондық желісінің сипатталған бөлімі құрылымды екінші тараптан талдай алады. Содан кейін, осы талдау негізінде қандай да бір шешімдер қабылдау немесе сілтемелер нейрон түрлеріне өзгерістер енгізе алады.

динамикалық қарым-қатынас жасау нейрондық желілер, жеке жүйке жасушаларының басқа бөлігіне нейрондық желі, олардың мәртебесін өзгерту немесе жад банк *paudalanaaladi* ретінде оқып.

ДНҚ молекуласы және мұндай автоматтагі таспааналогия қызықты. Сонымен қатар, «оригинатор» жеңдер жылғы нейрондық желінің бөлігі жеке нейрондардың функцияларын кейбір орындауға, құрылғыны жинап алады.

Қазіргі кезде, ДНҚ-ның кез-келген бөлігінен тұратын, нейрондық желі ретінде пайдаланылатын жадтың «таспасы» негізінде.

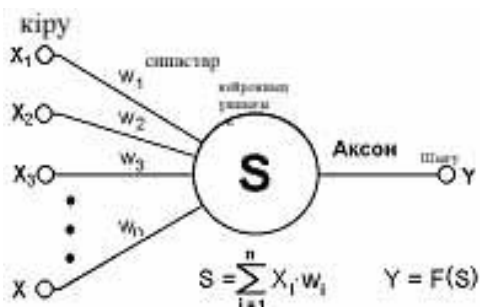
Вон Нейман семантикалық желілер мен желілерді бұл идеяларды қолданудан туындайтын зардаптарды ескереді.

Вон Нейманның семантикалық желісінің топологиясы шектеулі болады және тек логикалық мәндер өңделеді. Фон Нейман семантикалық нейрондық өңдеу желілік қосылымдар және нақты құндылықтар негізінде желі топологиясы бойынша ешқандай шектеулер жоқ. барлық Neuron желісі синхрондалған тахтилмен болып табылады. Семантикалық нейрондық желілер синхрондалған және синхрондалған нейрон болып табылады. нейрондардың нейрондық желілерді семантикалық желі топологиясы айырмашылығы фон Нейман, фон Нейман *adrestelwin* жоқ, нейрондық қатысуға шектелмейді. Бұл жағдайда абсолютті адресті енгізу керек.

Өткен онжылдықта қолданбалы математиканың жаңа өрісі жасанды нейрондық жүйе ретінде дамиды. Куксу кестилиги зерттеуі әртүрлі жүйке жүйелерінің бағытын көрсетеді. Ол үлгіні тану процесін автоматтандырады, адаптивті басқару, болжау, сараптама жүйелерін және көптеген басқа.

1.2 Жасанды нейрондық желінің құрылымы

Біз бейнені екі аллегорияға бөлеміз. Бұл объективті жіктеу, тізбектер алгоритмдерін жіктеу. Объектілерді жіктеуді бір деңгейде жүргізу керек. Кескіндердің объективті жіктелуі объектілерді топтарға жатқызып, олардың сәйкестігін табу болып табылады (1.3-сурет).



1.3-сурет – Жасанды нейрондық жүйенің құрылым

Кескінді тану жүйесінің негізгі сипаттамасы - сурет кейбір сыныптарға жататындығын тексеру. Бұл мәселені шешу үшін бөлінген функцияларды пайдалануға болады.

Бұл символдық бейнені тану үшін негізгі талдау компоненті

- оқу циклдерінің саны;
- жасырылған нейрондардың саны;
- бейне мүмкіндіктері: түпнұсқа өлшем, түпкілікті өлшеу;
- сынақ және тәжірибелік бөлімдерге бөлу.

Әдепкі бейне өлшемі кез келген бейнеге сәйкес қабылданады. Рециркуляциялық нейрондық желі - қысу үшін деректер жинағының жиынтығынан бір бейнені шығаратын процесс [3].

Бейресми қадам нейрондық желіні қайта айналдыру үшін қолданылады. Адаптивтік қадамды пайдалану желінің ажыратымдылығына жылдам жету және қателерді азайту үшін қолданылады. Оның алгоритмі келесідей:

Нейрондық желінің шығысы келесі формула бойынша есептеледі (1):

$$y_{ki} = x_i, k = 0, \quad (1)$$

$$y_{ki} = \tanh\left(\sum_{j=1}^p y_{k-1, j} w_{kij}\right), k = 1..L,$$

мұндағы, k – 0-ден L -ға дейінгі үлкейетін ағымдық бөлік;

P - $(k-1)$ алғашқы бөліктің нейрондар саны;

I – ағымдағы бөліктің нейрон индексі;

J – алдындағы бөліктің нейрон индексі;

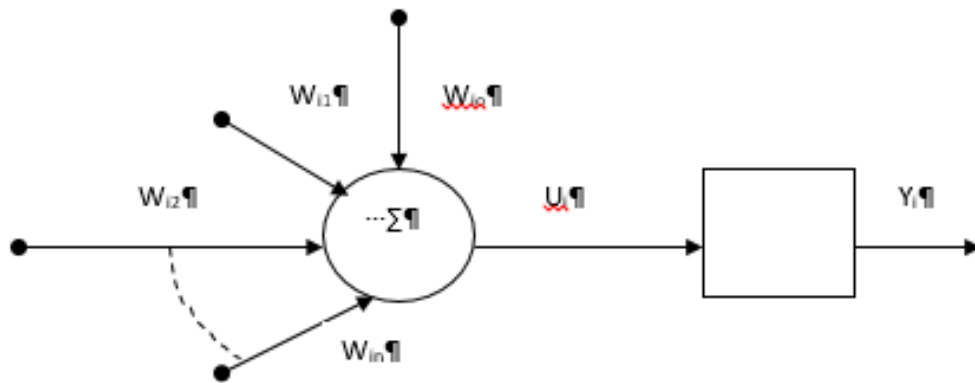
x_i – кіру бейнесінің пикселі;

y_{ki} – k қабығының шығу мәні;

$w_{ij} - j_{k-1}$ және j_k нейроны байланыстыратын салмақ.

Белсенді функция гиперболалық танген болуы мүмкін. Оның ауқымы $[-1;1]$ арасында жатады. Бұл бейне мен пикселдің орташа мәні $[-0.01; +0.01]$ бейнені жақсартуға байланысты төмендейді.

Әрбір нейронды жеке процессор ретінде қарастыруға болады. Ол тиісті сигнал салмағын есептейді және басқа нейрондық нейрондық желілердің шешуші функциясын анықтайды. Ең қарапайым модельдерде шығыс (шығу) сигналы екілік болып табылады; Ол 0 және 1 қабылдайды. 1 мәні - жоғары мәнге ие нейрон, ал 0 - мән бойынша жасалатын ең кіші мән. Нейрондық үлгілердің алғашқысы МакКаллохПитс болды. Мұнда нейрон екілік элемент [4].



1.4-сурет – Бинарлы нейрондық элемент

– кіріс сигналдары x_j ($j = 1, 2, \dots, n$) сәйкес w_{ij} қосу арқылы есептеледі;
 – тотализатор соңғы w_{io} -мен салыстырылады. Шығу нейрондық сигнал келесі қатынастардан есептеледі y_i (2).

$$y_i = f\left(\sum_{j=1}^n w_{ij} x_j(t) + w_{io}\right) \quad (2)$$

Функцияның дәлелі келесі сигнал болып табылады.

$$u_i = \sum_{j=1}^n w_{ij} x_j(t) + w_{io}$$

мұнда $f(u_i)$ - бұл функцияның белсенділігі. Төмендегі МакКаллохПитс бағдарламасындағы шекті функция (3):

$$f(u) = \begin{cases} 1, & u > 0; \\ 0, & u \leq 0; \end{cases} \quad (3)$$

w_{ij} коэффициенттері синапстық қосылымның салмағын білдіреді. Егер w_{ij} оң болса, ол бұзылған синапс болып табылады, ал теріс болса, теріс синапс

болып табылады. Егер $w_{ij} = 0$ болса, онда i және j нейрондары арасындағы байланыс жоқ. МакКаллок-Питс моделі дискреттік үлгі болып табылады. Нейронның мәні $(t + 1)$ алдыңғы кезеңде болатын мәнге ие болады. Бірнеше жылдан кейін Хебб зерттеуі қауымдастырылған жадты зерттеу нәтижесінде нейрондық теорияны ұсынды. Нейрондар арасындағы байланыс нейрондардың белсенділігіне көбеюі керек. Хебб мысалы келесі қағидаға сүйенеді: (4):

$$w_{ij}(k+1) = w_{ij}(k) + \eta u_i(k) u_j(k) \quad (4)$$

мұндағы, k - цикл нөмірі;
 η – оқыту коэффициенті.

Тренинг кезінде, u_i және u_j нейрондарының шығу сигналдары салыстырмалы болып табылады. Есептеу жүйелерінің жылдам дамуы арқасында жаңа технологиялық есептерді шешу, кешенді жүйені басқару және тезірек өңдеу сияқты бейнелерді тану, қабылдау және түсіндіру сияқты үрдістер. Қазіргі уақытта жасанды нейрондық желісі жоғары дамыған білімнің бөлігі болып табылады.

Нейрондық желіні қолдану мүмкіндігі.

Кез келген нейрондық желі автономды жүйе ретінде пайдаланылады. Бұл жүйенің құрамдас бөлігі ретінде оның сапасындағы басқа элементтерге сигнал жасанды нейрондық желімен байланыспайды. Желілік функцияны бірнеше негізгі топтарға бөлуге болады: жуықтау және интерполяция, бейнені тану және жіктеу, деректерді қысу, болжау, сәйкестендіру, басқару, бірлестіктер.

Жоғарыда айтылғандар арасында нейрондық желі бірнеше негізгі айнаымалы функцияны жақындатуда маңызды рөл атқарады. Ол сызықты емес функция арқылы іске асады. (5):

$$y=f(x) \quad (5)$$

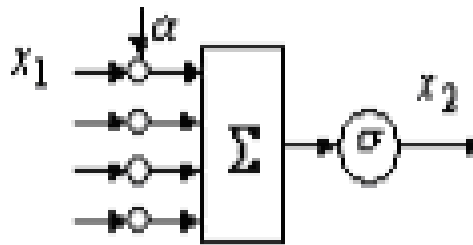
мұндағы, x – кіру векторы;
 y – бірнеше айнаымалылар векторының функциясын тарату.

Бейне тану және жіктеу үшін желінің негізгі ерекшелігі - сурет элементінің орналасу орны, Фурье түрлендіру компоненттері және басқа факторлар кескіннің нүктелік үлгісін геометриялық түрде визуализациялау арқылы. Тиісті сыныпты қабылдау үдерісін зерттеуде базаны құрайтын әр түрлі ерекшеліктер ерекшеленеді.

Желі құрастырған кезде кез-келген архитектураның кез-келген элементтерінен желіні құруға болады. Дегенмен өзіңіздің желіңізді құру кезінде арнайы конфигурацияланған жалпы функцияларды қадағалау маңызды. Өйткені желілер бір-бірінен ажыратылуға тиіс. Сондықтан бейімделгіш көшудің қарапайым өтуі. Дегенмен, желілік кіру циклі желі құрылымына айтарлықтай әсер етеді. Сондықтан бүкіл желіні екі негізгі айырмашылыққа бөлуге болады: циклдық желі және циклдар желісі. Циклдің ортасында желі

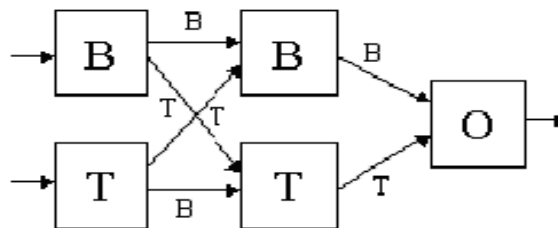
жұмысына айтарлықтай әсер ететін тағы бір бөлім бар: цикл - бұл басқа салмағы мен циклімен цикл.

Барлық қолданылатын желілер ішкі желінің ішкі желісінің шығысқа қалай әсер ететінін анықтай алмайды. 2-сурет желі мысалын көрсетеді. А-параметрін жоғарылату бірегей дыбысқа әкеледі. Ал оң сандар үшін - өседі (1.5-Сурет).



1.5-сурет – Монотонды емес желі фрагменті

Осындай желінің шығатын дыбысы монотонды емес желі α параметріне тәуелді болады (1.6-сурет).



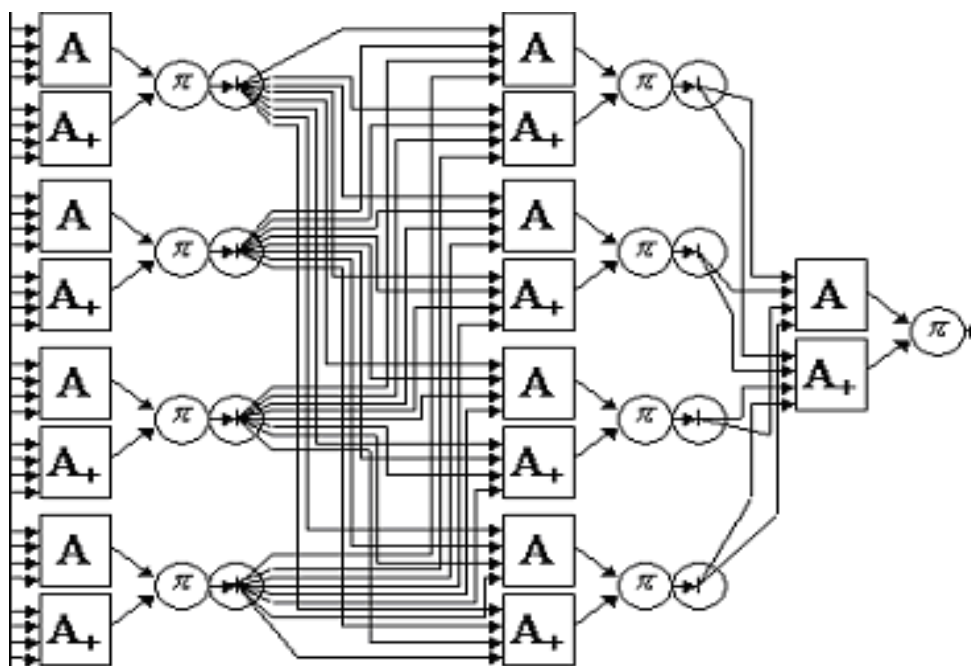
1.6-сурет – Монотонды желінің жалпы кестесі

Параллель - қозғалатын нейронның бөлігі, төменгі жол - тежегіш. Т қарібі - қозғалтқыштың тежеуіне және қарібіне байланысты

Монотонді ішкі қабығына тәуелді болу үшін, нейрондық желілік архитектураның аудио шығыс тәуелділігін қолданыңыз. Төмендегі 1.7 кестеде көрсетілген монотонды іргелі желілік архитектура:

Әр қабықтың біртұтас сызығын құрудың негізгі идеясы екіге бөлінеді. Ол көлікпен жүреді және тежейді. Осыған байланысты қабықтағы желілік элементтерін басқарудың барлық элементтері қабықтың келесі элементін қозғайды, ал келесі элементті тежеу үшін пайдаланылады. Осыған байланысты тежегіш элементтер элементтердің көмегімен іске асырылады және жетек элементтері қабықшаның келесі элементтеріне әкеледі. «Тежегіш» және «қозғалатын» бөлік барлық элементтердің атауы.

Ескерту Біртекті салмақ сызығы үшін барлық қажетті элементтермен байланысу үшін керемет сигмоиальды теріс болмауы керек. Желінің сұранысының архитектурасын өзгертетін желінің біртұтас монотондылық элементі бар.



1.7-сурет – Падеэлементті монотонды емес желі

Жаңа желі параметрлерін шектеудің қажеті жоқ. 1.7-суретте немонотоникалық сызықтың мысалы көрсетілген. 1.6 Pade элементімен біртұтас тізбектің мысалын көрсетеді.

Қауымдастықтағы нейрондық желіні сақтау құрылғысының ассоциативті рөлі маңызды рөл атқарады. Нейрондық желінің басты ерекшелігі - параллельді ақпаратты бір мезгілде қолдану мүмкіндіктері мен барлық нейронды өңдеу мүмкіндігі. Бұл функцияға байланысты нейрондар арасындағы байланыстар саны ақпаратты өңдеу үдерісін жеделдетуге көмектеседі. Көптеген практикалық қосымшаларға қарамастан, жасанды нейрондық желі - сигналдарды қолдану үшін шексіз шешім.

Ықтимал ғана емес сызықтық нейрондық желінің түрін, сонымен қатар жоғары ретті нейрондық желісін, Хопфилда желісін, өзін-өзі ұйымдастыру Коенопа желісін сипаттау үшін. Осы желілерді пайдалану арқылы сіз бірнеше бейне тану үшін негіз жасай аласыз. Көбіне, бейненің күрделілігіне байланысты бірнеше желілер пайдаланылады.

Нейрондық желілер арқылы рәміздерді бейнелеу жүйесін жеделдету және тану сенімділігі артады.

1.3 Нейрондық желідегі синаптикалық байланыстар

Жасанды нейрондық желілер - желілік элементтер өзара іс-қимыл арқылы анықталады қарапайым есептеу элементтерін көптеген элементтері бар жүйе, есептеу және элементтер өздері туралы, немесе тораптар орындалады және параллель жұмыс істейді.

Компьютерлік құрылғы жасанды нейрондық желі ретінде түсініледі.

синапстар немесе контактілер деп аталатын Буындар алмасу ақпаратпен, өзара байланыстырып тұрған - нейрондық Ол қарапайым өңдеу элементтер саны көп тұрады. Нейрон байланыс желісі сыртқы әлеммен және топқа нейрондық желі берілетін сигналдарды қабылдайды шығыс байланыс байланысады топты бар.

Сигнал (лат Сигнум) - таратылатын хабар тарататын физикалық процесс.

Нейрондық желілер классификациялық және жобалаудың әртүрлі мәселелерін шешу үшін қолданылады.

Неврологияны имитациялау екі жолмен жүзеге асырылады:

- компьютерде неврологияны моделдеу;
- мамандандырылған нейроплата және компьютерлік нейрондық желі жүйелерін жеделдету нейропроцесстер құру. Бірінші әдіс компьютер мен жеңілiske салыстырғанда жылдамдығы болып табылады, және екінші әдісі модель бір моделі нейрожелі нейроплата жылжыту немесе үлгісін өзгерту және нейропроцесстерді ауыстырылуы тиіс анықтау үшін рұқсат етілмейді пайдаланады.

Неврологияда проблемаларды дұрыс шешу келесі байланысты:

- пайдаланушының имиджін тану;
- іздеу және ақпараттық моделдеу: өз тілін қалыптастыру;
- математикалық жүйенің күрделі және сызықты емес модельдерін қалыптастыру;
- бақылау жүйесі;
- шешім қабылдау және логикалық шешім.
- неврологиялық эксперименттік деректерді сақтау мен қолданудың табиғи қабілеті бар параллель процессорлар жиынтығы. екі жағдайда да миға ұқсас:
 - білім базасы желілік оқыту процесінде қалыптасады;
 - синаптические салмақ ретінде анықталады интерстициалдық нейрондар күшін, сақтау үшін пайдаланылады.

Салмағы - бетіне жақын орган әсер ауырлық саны: $P = mg$, m - Дене бұқаралық, g - ағынының (немесе ауырлық жеделдету) жеделдету. Дене салмағы тұрақты болып табылады, және g мәні ендік және теңіз деңгейінің биіктігіне байланысты (мысалы, $g = 9804 \text{ м / с}^2$ Алматы үшін) өзгеріп отырады.

Логика (гр Λογική - «Талдау», λόγος - «сөз», «ұсыныс», «ойлау», «ақыл») - ойлау ғылым және оның формалары мен заңдылықтары болып табылады. логикалық пайымдау және бас тарту әдістерін Кейбір теориялық теориялар жиынтығын құрайды.

Деректер (Data) - ыңғайлы, кейде болуы өтінімде, автоматты арқылы өңделеді деректер болып табылады. Деректерді енгізу, шығару, бақылау, проблемалық, сандық, мәтіндік, графикалық және т.б.

Жағдай - бұл адам қызметі, жануарлар өмірі, және қоғамның табиғаты өзгеріп, оқиға, қоршаған ортаға тәуелділігін білдіреді және т.б. D.

философиялық ұғым болып табылады .. Бұл сондай-ақ табиғатта, қоғамда белгілі бір өзгеріс үшін алғышарт болып саналады.

Естеріңізге сала кетейік - еске сақтау. Жадтағы үрдістердің бірі - жаңа ақпаратты енгізуді орнату. Психодинамическую жабдықтар жад динамикасы - жиі адам ақпаратты оқиды және өңдейді қалай тез анықтайды адам жады динамикалық сипаттамаларын анықтау үшін көмектеседі психикалық әдістері, [5].

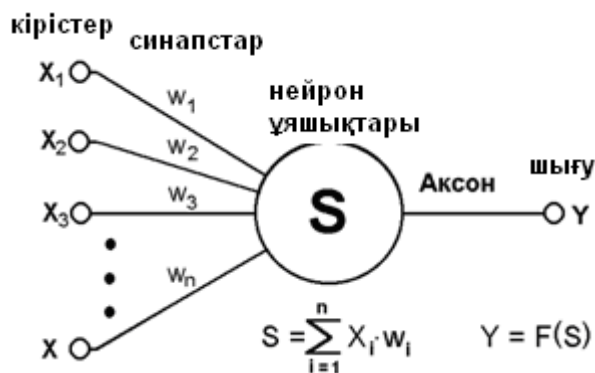
Синаптикалық салмақтар неврологияның параметрлері ретінде қарастырылады. Сонымен қатар, жасанды нейрондық желі жүйелері және нейрондық желі жүйелері олардың алуға, сақтау және эксперименттік білім пайдалануға мүмкіндік береді физикалық ұялы жүйелер болып табылады.

Жасанды нейрондардың құрылымын және оларды нақты мәселелерді шешуде қолдануды қарастырайық.

Физика (грекше: φύσις - Nature) әлемді және оның қозғалысын зерттейді. Дене тұрғысынан энергетика, энергия, массасы, фильмі және т.б., сондай-ақ тұжырымдамалар.

Структура - баланстық құрылым құрылымының тұтастығын, объектінің негізгі функцияларын (Structura латын тілі, орналасуы, тәртібі) қамтамасыз ететін қатынастар мен қатынастардың жиынтығы. Құрылым жүйе мен элементтердің тұжырымдамасымен тығыз байланысты.

Неврологияның кең ауқымына қарамастан, оларда ортақ тіл бар. Олардың барлығы адамдар ретінде біріктірілген қарапайым нейрондық жиынтығынан тұрады. (1.8-сурет).



1.8-сурет – Нейронның схемасы

Неврологиясы нейронның кіруі мен ядросын байланыстыратын синапстардан тұрады. Нейронды ядро кіріс сигналдарын және аксон өңдеуін орындайды және нейронды нейронды келесі қабатына жалғайды. Әрбір синапс салмағы бар. Ол өзінің күйінің нейронына әсер ету дәрежесін анықтайды. Нейронның жағдайы (6) формула бойынша анықталады.

$$S = \sum_{i=1}^n x_i w_i \quad (6)$$

мұндағы, n – нейронның енгізілу саны;
 x_i – нейронның i -інші енгізілу мәні;
 w_i – i -інші синапс салмағы.

Одан кейін нейрон аксоны мәні мына формула бойынша анықталады:

$$y = f(s)$$

мұндағы, f – белсенді функция. Белсенді функция ретінде сигмоид жиі қолданылады, ол мына (7) түрде болады.

$$f(x) = \frac{1}{1+e^{-\alpha x}} \quad (7)$$

Осьтің осі бойынша сараланған және қарапайым өнім болып табылатын осы функцияның жетістігі.

$$f'(x) = \alpha f(x)(1 - f(x)) \quad (8)$$

Айналмалы нейрон. Артқы невродииттың жоғары сапалы дизайнының заңдылығын табудың қуатты құралы. Реверсивтік нейрон бірнеше нейрондық қабаттан тұрады, i -нейрон әр қабаты, $i - 1$ нейронмен байланысты. Жалпы алғанда, оқыту нейрондары функционалдық тәуелділікті тудыруы мүмкін $y = f(x)$. Мұнда x - кіріс векторы, y - шығу векторы. Жалпы, мұндай есептер шектеулі жиынтықтар жиынтығында шексіз шешімдер жиынтығын ұсынады. Бөлінген кеңістікте нейродинамика мәселесі объективті функциямен азайтылады.

Функционалдық - бір немесе бірнеше функцияларға байланысты айнымалы математикалық көрініс. Ол вариацияларды есептеуде алғаш рет пайда болды. Жабық қисық жабылған аумақ белгілі бір жолда жұмыс істейді және тағы басқалар.

Ғарыш - философия, математика және физика сияқты салаларды қолданатын күрделі ұғым. Күнделікті өмірде іс-қимыл саласы бар, бәрі бар жалпы контейнер, ыңғайлы сезінетін жүйе бар [6].

Ол ең кіші квадраттар әдісі (9) бойынша табылады:

$$E(w) = \frac{1}{2} \sum_{j=1}^p (y_j - d_j) \quad (9)$$

мұндағы, y_j - j -ші шығатын нейрожелі мәні;
 d_j - j -ші шығудың мақсатты мәні;
 p – шығу қабатындағы нейрон саны.

Нейрожеліні оқыту градиентті түсу әдісімен жүргізіледі, мұнда әрбір итерациядағы салмақтың өзгеруі (10-11) формула бойынша жүргізіледі.

$$\Delta w_{ij} = -n \frac{\partial E}{\partial w_{ij}} \quad (10)$$

мұндағы, η - оқыту жылдамдығын анықтайтын параметр

$$\frac{\partial E}{\partial w_{ij}} = \frac{\partial E}{\partial y_j} * \frac{dy_j}{ds_j} * \frac{\partial s_j}{\partial w_{ij}} \quad (11)$$

мұндағы, y_j - нейронның j -ші шығу мәні;

S_j – (12) формула бойынша анықталатын енгізу сигналдарының қосындысы.

Мұндағы жиын:

$$\frac{\partial s_j}{\partial w_{ij}} = x_i \quad (12)$$

Әрі қарай (13) формуланың бірінші жиынын анықтауды қарастырайық:

$$\frac{\partial E}{\partial y_j} = \sum_k \frac{\partial E}{\partial y_k} * \frac{dy_k}{ds_k} * \frac{\partial s_k}{\partial y_j} \quad (13)$$

мұндағы, k – $n-1$ қабаттағы нейрон саны;

Қосымша айнымалы енгіземіз (14):

$$\delta_j^{(n)} = \frac{\partial E}{\partial y_j} * \frac{dy_j}{ds_j} \quad (14)$$

мұндағы, n –қабаттағы δ анықтау дарекурсивті формуланы табамыз.

Рекурсия (лат. recursio кейін қайту) – дыбыс артикуляциясының үшінші соңғы фазасы, яғни сөйлеу мүшелерінің дыбыс жасау позициясынан айрылып, бастапқы бейтарап калпына қайта оралуы. Жалпы мағынасында рекурсия – элементтерін өзіне (15-16) ұқсас түрде қайталайтын процестер [7].

$$\delta_j^{(n)} = \left[\sum_k \delta_k^{(n+1)} * w_{jk}^{(n+1)} \right] * \frac{dy_j}{ds_j} \quad (15)$$

мұндағы, δ - анықтау қиынға соғады, себебі бізге тек қана мақсатты вектор белгілі.

$$\delta_j^{(n)} = (y_i^{(N)} - d_i) * \frac{dy_j}{ds_j} \quad (16)$$

Енді (17) формуланы ашып жазааламыз:

$$\Delta w_{ij}^{(n)} = -n * \delta_j^{(n)} * x_i^n \quad (17)$$

Нейрожеліні оқытудың толық алгоритмін қарастырайық:

- талап ететін үлгінің бірін нейрожеліге ұсыну және нейрожелідегі шығатын нейрондардың мәнін анықтау;
- (16) – формула бойынша нейрожелінің шығатын қабаттарын және (13) формула бойынша шығатын қабаттың салмағының өзгеруін есептеу;
- (15) және (17) формула бойынша сәйкесті нейрожелі қабаттарын есептеу;
- (18) – барлық нейрожелі салмағын түзету;

$$\Delta w_{ij}^{(n)} = -n * \delta_j^{(n)} * x_i^n \quad (18)$$

– егер қате табылса, онда 1-қадамға өту керек.

Әртүрлі салмақта туындылары қатты айрықшаланғандықтан қарапайым градиентті түсу әдісі өте тиімді бола бермейді. Бұл әдісті жетілдіру үшін уақыт бойынша салмақтың өзгеруін енгізу керек. Онда (17) формула (17.1) түрге келтіріледі.

$$\Delta w_{ij}^{(n)}(t) = -n * \delta_j^{(n)} * x_j^n + \mu \Delta w_{ij}^{(n)}(t - 1) \quad (17.1)$$

1.4 Нейрондық желілер

Осы мақалада үлгілерді тану және тәуелділіктерді қалпына келтіру проблемалары нейрондық желілерді пайдалану арқылы шешілетін болады. Бұл тақырыпты [8],[9] шолуда негізделген.

1.4.1 Негізгі элементтер

Нейрондық желі келесі негізгі элементтерден тұратын өзара байланысқан ұялы автоматтың құрылымы болып табылады:

Нейрон кіріс сигналын функцияларға түрлендіретін элемент (19):

$$f(x) = \frac{x}{c + |x|} + -m \quad (19)$$

мұндағы, x - кіріс сигналы;

c - шекті функция графигінің көлбеуін анықтайтын параметр;

m - нейронның спонтанды белсенді параметрі.

Тұтқыш – оның кірісіне келген сигналдардың жиынтығын жүзеге асыратын элемент (20):

$$f = \sum_{i=1}^N xi \quad (20)$$

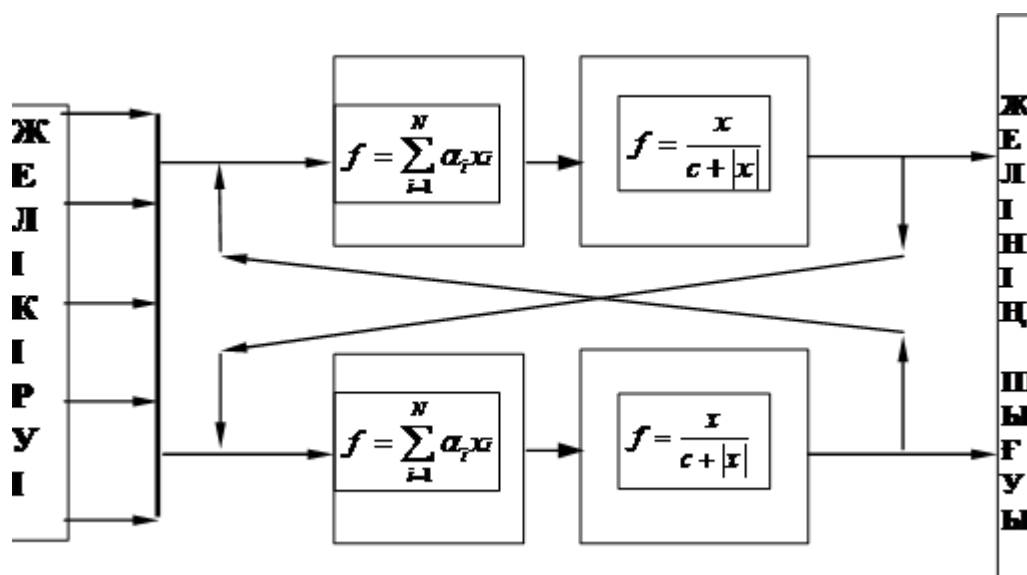
Синапс – сызықтық сигнал беруді жүзеге асыратын элемент (21):

$$f(x) = w \cdot x \quad (21)$$

мұндағы, w – сәйкес синапстардың «салмағы».

1.4.2 Желі құрылымы

Желі синапстармен байланыстырылған нейрондардан келесі схемаға сәйкес 1.9-суретте жиынтықтан тұрады:



1.9-сурет – Нейрондық желінің құрылымы

1.4.3 Тікелей желінің жұмысы

Желі уақытында (циклдарда) дискретті функцияларды атқарады. Содан кейін синапстар синтезге бөлінеді, ол сигналдарды белгілі бір сағаттар циклі бойынша жібереді және нейронды шығудан сигналды келесі операциялық циклге «жады синапсаттары» жібереді. Желі жұмысында пайда болатын сигналдар тікелей (нәтиже желі болған кезде пайдаланылады) және қосарланған (жаттығуда пайдаланылады) болып бөлінеді және келесі формулалар арқылы берілуі мүмкін:

Уақыт сағатта i -ші нейрон үшін (22):

$$\begin{aligned}
A_{i1} &= m_{i0} + x_{i1} \\
A_{iT} &= \alpha M_i \cdot A_{T_{i-1}} + \sum_j \alpha_{ji} \cdot f_{jT} - 1 \\
f_{iT} &= \frac{A_{iT}}{\alpha i1 + |A_{iT}|} + \alpha i2 \\
f_{iT,A} &= \frac{\alpha i1}{(\alpha i1 + |A_{iT}|)^2}
\end{aligned} \tag{22}$$

мұндағы, m_{i0} – бастапқы желінің параметрі;
 x_{i1} – белсенді нейрондық кіріс желілік сигналдар;
 f_{iT} – T цикл уақыттағы шығыс нейрон сигналы;
 A_{i1} – i -ші кіріс нейрондық желісінің бірінші параметрі;
 A_{iT} – кіріс i -ші нейроның салмағы [8].
 i -ші нейроннан j -ға дейінгі байланыс синапсіне (23) арналған:

$$\begin{aligned}
\sigma_{ijT} &= f_{iT} \\
s_{ijT} &= \alpha_{ij} \cdot f_{iT}
\end{aligned} \tag{23}$$

мұндағы, s_{ijT} – i -ші нейроннан бастап j -ге дейінгі синапсты енгізу;
 f_{iT} – i -ші нейронның шығу сигналы;
 α_{ij} – бұл синапстардың салмағы;
 s_{ijT} – сағатта синапс шығару.
 i -ші нейронды есте сақтау синапсіне арналған (24):

$$m_{iT} = \alpha M_i \cdot A_{iT} \tag{24}$$

1.4.4 Желіні оқыту

Бұл міндетте оқыту «байланысшы» моделімен, яғни синапстардың салмағын реттеу арқылы жүзеге асырылады. Оқытудың мәні – қате функциясын азайту, яғни W – синапстардың салмағы. Минимизация мәселесін шешу үшін функцияның градиентін реттелетін параметрлерден (25) есептеу қажет:

$$\overline{grad(H)} = \frac{\partial H}{\partial w_i} \tag{25}$$

Кері операция. Градиенттің есептеуі келесі цикл бойынша сағат циклдерінің кері санымен жүзеге асырылады:
Байланыс синапсіне (26) арналған:

$$\mu_{ij\Theta} = 0$$

$$\mu_{jiT} = \mu_{jT+1} \cdot f_{jT+1, A} + \mu_{mjT+1} \cdot \alpha_{jm}(T=\Theta-1, \Theta-2, \dots) \quad (26)$$

Ақыр соңында, синапстар сағаттардан өтіп болған соң, синапстардың салмағының ішінара туындылары тиісінше жады синапстарының және байланыс синапстарының формасын (27) құрайды:

$$\begin{aligned} \frac{\partial H}{\partial \alpha_{Mi}} &= \sum_{T=\Theta-1}^1 \mu_{MiT} \cdot A_{iT} \\ \frac{\partial H}{\partial \alpha_{ij}} &= \sum_{T=\Theta-1}^1 \mu_{sijT} \cdot f_{iT} \end{aligned} \quad (27)$$

1.5 Тесттік сұрақтардың маңыздылығын бағалау

Кіріс сигналдарының маңыздылығын бағалау кезінде алынған нәтиже (тиісінше LOBI сұрақтары) қызығушылық тудырады.

Нейрондық желінің кейбір функционалды элементі кейбір сигналдардың А сигналдарының кіретін векторын бейімдеу параметрлері векторы түрлендіретін болсын. Н - нейрондық желінің сигналдарына және кіріс сигналдарына және нейрондық желінің параметрлеріне тікелей байланысты тәуелді бағалау функциясы болсын. Қос функциямен V элементінің ішінара туындылары есептеледі. Бұл туындылар бағалаудың сезімталдықты параметрдің өзгеруіне көрсетеді, соғұрлым осы үлгі үшін осы параметрмен көп өзгереді. Сондай-ақ, параметрдің туындысы басқалармен салыстырғанда өте аз, яғни бұл жаттығу кезінде параметр өзгермейді. Осылайша, сіз кем дегенде сезімтал болып нейрондық желі және өзгерту үшін емес, оларды оқыту процесінде өзгерістерге параметрлерін тобын таңдай аласыз. Әрине, ең шағын тобын немесе жоғары сезімталдығы анықтау үшін бірнеше оқу цикл және барлық мысалдар Taskbook үшін параметрлер бағалау функциясы туынды пайдалану қажет. Бағалау функциясы азайту динамикасы оқыту кезінде нейрондық желі оқытудың әр түрлі кезеңдерінде өзгеріп отыр. Оқудың осы сатысында қандай кіріс сигналдарын нейрондық желі үшін маңызы бар екенін анықтау маңызды. Бұл ақпарат кіріс сигналдарының өлшемі жоғары болып табылады және ол оқыту нейрондық желі нашарлатпай кіріс сигналдарының өлшем азайтуға мүмкіндік береді, өйткені оқу уақыты, сондай-ақ, қажет етеді жағдайларда пайдалы болып табылады.

Нейрондық желі маңызы көрсеткіштерін есептеу тәртібін қамтиды, оқыту басталғаннан кейін бірнеше циклінен кейін біз бұл мәселені шешу үшін келесі алгоритмі ұсынылады:

- көрсеткіштердің оң бағыты мен маңыздылығын табу;
- нейрондық желіні оқыту;

– іске қосу бастау.

Алғашқы бірнеше цикл әр түрлі болуы мүмкін, өйткені ол, әр жағдайда жеке (28) таңдалады. Әрі қарай сезгіштіктің кейбір нормада жинақталған оның оқу барысында бірнеше цикл (29), бар.

$$xi = \sum_{k=1}^M |x_k^i| \quad (28)$$

$$xi = \max_{k=1, M} |x_k^i| \quad (29)$$

мұндағы, x – i -ші сигнал үшін норма;

\max – k -ші мысалда i -ші сигнал маңызын бағалау;

M – санау циклдар саны.

Сандар есептелгеннен кейін кіріс сигналдарының санын азайтуға болады. Төмендеуі мәні төмен аз, ол оқу процесін әсер фактісі негізінде жасалуы тиіс [9].

Қорытындылай келгенде, математикалық аппарат жеткіліксіз заманауи талаптарға сай психодиагностика пайдаланылады. Тану және қалпына келтіру шектеулерге байланысты математикалық машина диагностика құралдарын іске асыру үшін қажет.

Бар математикалық әдістер мен алгоритмдерді олардың сарапшылар қолдануға тым күрделі және уақыт тұтынатын болып - субъектісі, психодиагностика қоса алғанда және тікелей прецедент компьютерлік әдістері адам маманы тәжірибесін үйрену мүмкіндік бермейді. Олардың құрушылардың математикалық дайындық жөніндегі талаптар барынша азайту үшін нейрондық желі сараптамалық психологиялық жүйелерін құру, нейрондық желілерді математикалық аппаратты пайдаланады.

2 Жобалау бөлімі

2.1 Мәселені қалыптастыру

R функциясының кеңістіктің кездейсоқ нүктелерінде оның мәндерінің жиыны беріледі $(x_1, y_1), \dots, (x_N, y_N)$.

Біз оның жуықтауын тегіс және үздіксіз дифференциалданатын функциялар жиынтығынан (30) құрастырамыз онда :

$$\begin{aligned} F_1 &= F - f(x, a) \\ F_2 &= F_1 - f_1(x, a) \\ F_3 &= F_2 - f_2(x, a) \end{aligned} \quad (30)$$

мұндағы, $F_1 - F$ функциясының жақындаған қателігі $f_1(\bar{X}, \bar{a})$;

F_2 – жақындатудың алдыңғы қадамының қателігі;

F_3 – қорытынды қателік.

Аппроксимация коэффициенттерді таңдау арқылы ғана емес, сондай-ақ әрбір қадамда функцияларды таңдау арқылы жүзеге асырылуы мүмкін. Осылайша, F кеңейтілімін келесі түрдегі конвергентті сериялардан (31) алуға болады:

$$F = \sum f_i(a, x) \quad (31)$$

Сәйкесінше мәселені шешу ауытқудың квадратына сәйкес келетін сапа функционалын азайту (32) арқылы алынады:

$$H = \sum (F(\bar{a}^i, \bar{x}^i) - y_i)^2 \rightarrow \min \quad (32)$$

Мәселе нейрондардың белгісіз санымен нейрондық нейрондардың түрлендіргішінде қолданылатын нейрондық желілік предикторды пайдалана отырып, нүктелердің бастапқы үлгісімен берілген F функциясына жақындағаннан тұрады. Шешім итеративті процесс ретінде ұсынылуы мүмкін, келесі қадамдардан тұрады:

- жаңа нейронды қосу;
- түрлендіргіш функциясын, оның параметрлері мен синапстардың салмағын таңдап, ағымдағы нейрон үшін берілген нүктелердегі мәндерді болжау қатесін оңтайландыру [10].

2.2 Аналитикалық шешім

Енді Y_i - келесі қабаттың мәні болсын. Сонда Y_0 - экспериментальды үлгідегі нүктелердегі жақындаған функцияның мәні, ал Y_1 және кейінгілері - тиісті кадамда есептеу қателігі.

Оқыту мәселенің кез-келген градиент әдісімен желілік параметрлерді оңтайландыру арқылы жүргізіледі.

Сонда k -ші нейронды (33) оқығанда:

$$Y_k = w_{k1}^{(2)} * F_k * (\alpha_k * \sum_{i=1}^{N_{\%oo}} w_{ik}^{(1)} x_i^j) \quad (33)$$

тиісінше, барлық тапсырмалар кітабы үшін H (қате функциясы) пішінге (34) ие болады:

$$H = \sum_{j=1}^{N \cdot p} \left[w_{k1}^{(2)} * F_k * (\alpha_k * \sum_{i=1}^{N_{\%oo}} w_{ik}^{(1)} * x_i^j) - Y_k^j \right]^2 \quad (34)$$

яғни аппроксимацияланған және жақындату функцияларының жақындығы өлшемі ретінде бүкіл оқу үлгісі бойынша квадраттың қателігі анықталды. Әрбір келесі нейронды оқыту үшін бірінші қабаттың синапстарының салмақтары үшін ішінара туынды функциялар қолданылады:

$$\frac{\partial H_k}{\partial w_{ik}^{(1)}} = \sum_{j=1}^{N \cdot p} 2 \left[w_{k1}^{(2)} * F_k * (\alpha_k * \sum_{i=1}^{N_{\%oo}} w_{ik}^{(1)} * x_i^j) - Y_k^j \right] \left[w_{k1}^{(2)} * \alpha_k * x_i^j * F'_k * (\alpha_k * \sum_{i=1}^{N_{\%oo}} w_{ik}^{(1)} * x_i^j) \right]$$

нейрондық параметр α_k :

$$\frac{\partial H_k}{\partial \alpha_k} = \sum_{j=1}^{N \cdot p} 2 \left[w_{k1}^{(2)} * F_k * (\alpha_k * \sum_{i=1}^{N_{\%oo}} w_{ik}^{(1)} * x_i^j) - Y_k^j \right] \left[w_{k1}^{(2)} * (\sum_{i=1}^{N_{\%oo}} w_{ik}^{(1)} * x_i^j) F'_k * (\alpha_k * \sum_{i=1}^{N_{\%oo}} w_{ik}^{(1)} * x_i^j) \right]$$

және осы нейронға сәйкес екінші (шығу) қабатының синапсалық салмағы (35):

$$\frac{\partial Y_k}{\partial w_{k1}^{(2)}} = \sum_{j=1}^{N \cdot p} 2 \left[w_{k1}^{(2)} * F_k * (\alpha_k * \sum_{i=1}^{N_{\%oo}} w_{ik}^{(1)} * x_i^j) - Y_k^j \right] \left[F_k * (\alpha_k * \sum_{i=1}^{N_{\%oo}} w_{ik}^{(1)} * x_i^j) \right] \quad (35)$$

N_{np} -оқу үлгісі мысалдарының саны.

Дегенмен, егер H функциясының есептеуі процессорлық уақыттың T_C тапсырысының құнымен байланысты болса, онда оның градиентін дәстүрлі жолмен есептеу тәртібін талап етеді [11]:

$$T_{gradH} = nT_H$$

мұндағы, n - функцияның айнымалыларының саны.

Нейрондық желілер дәстүрлі түрде пайдаланылатын мәселелерде n мәндері бірнеше мыңға дейін жетуі мүмкін екенін ескере отырып, қателік функциясының градиентін есептеуге арналған аналитикалық шешім қабылданбайды деп санауға болады.

Дегенмен F автоматтандырылған желісі ретінде F функциясының шешім функциясын сипаттау кезінде, қате функциясының градиентін есептеу бастапқы жүйеге қосарлы жүйенің жұмыс істеуі ретінде ұсынылуы мүмкін.

Бұл тәсілмен:

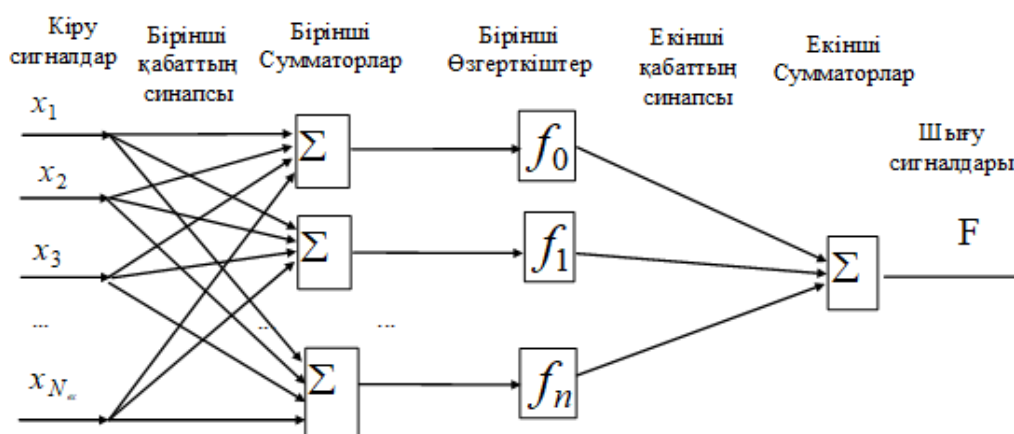
$$T_{gradH} \approx CT_H \quad (36)$$

мұндағы, C – n тәуелсіз, ал көбіне шамамен 3 тең.

Осылайша, біз нейрондық желілердің идеологиясында түпнұсқа мәселені шешу туралы есепке келеміз.

2.3 Нейрондық желілер идеологиясына шешім шығару

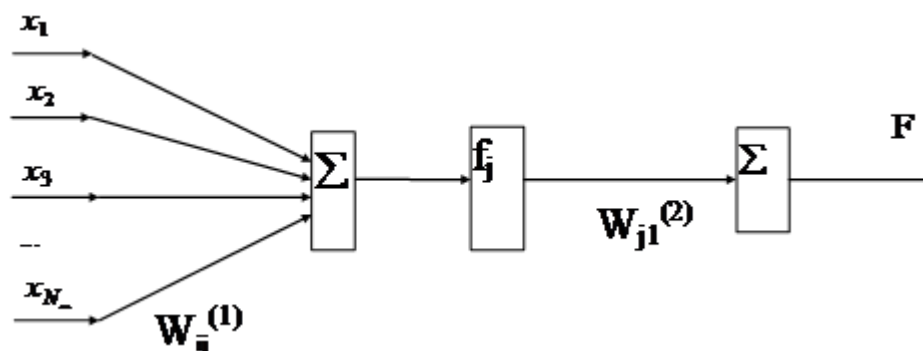
Жоғарыда айтылғандай, жоғарыда көрсетілгендей, түрлер сериясы 2.1-суретте көрсетілген құрылымы бар нейрондық желі ретінде ұсынылуы мүмкін.



2.1-сурет – Ағымдық нейрондық желінің құрылымы

Содан кейін қате функциясы Градиент есептеу H , қос өнер пайдаланып схемасын өндіре алады. мәселені шешу айтарлықтай мынадай құрылғысын пайдалану арқылы жеңілдетуге болады. алдыңғы өзінде оқытылған, және олардың байланыс тіркелген кезде келесі қабаты оқыту басталады бастап, содан кейін, шын мәнінде, әрбір нейрон басқалардан бөлек оқиды, және есептелген құндылық ретінде K -ші нейроны қате есептеу функциясы алдыңғы $K-1$ қабылданады - нейрондық немесе F_i [12].

Бүкіл желісін оқыту процесі, ол 2.2 суретте көрсетілгендей оқыту түрі құрылымын дәйекті процестерді сериясы осы бөлімнен қысқарды.



2.2-сурет – Бір нейронның құрылымы

мұндағы, $x_1 - x_n$ - кіріс желілік сигналдар,

$W_{ij}^{(1)}$ – бірінші қабатының J -ші сумматоры үшін i -ші кіріс бірінші қабатының синапстар;

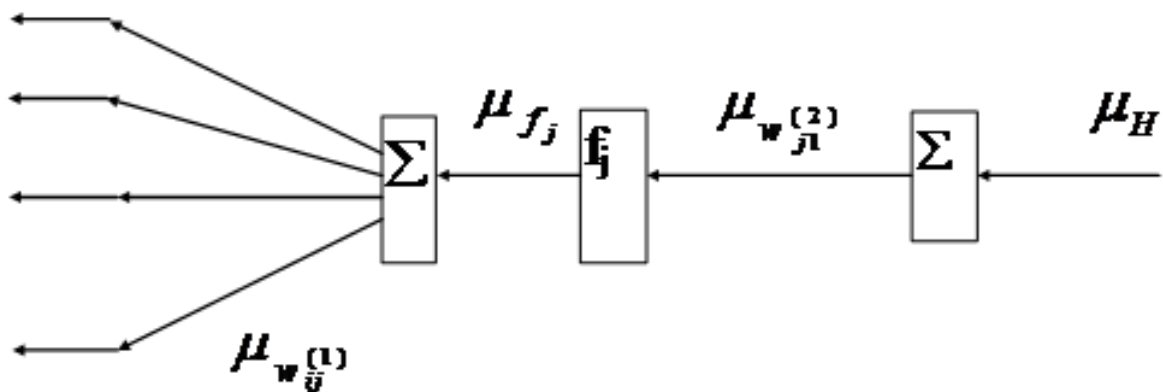
$W_{j1}^{(2)}$ – J - екінші шығыс сумматоры үшін J -ші түрлендіргіш қабаты;

F_j – j -ші синапстар түрлендіргіштің символы, тривиальды көтергішті білдіреді.

Мәтінде мұндай құрылым «желілік ағын» немесе жай ғана «ағын» деп аталады.

Біртұтас «ағын» желісі үшін H (қате функциясы) туындыларының есебі қосарланған функциялардың аппараты және қатені кері тарату үшін алгоритм негізінде жүзеге асырылуы мүмкін. Кері режимде сигнал беру схемасы 2.3-суретте көрсетілген [13].

Мұнда бағалау функциясынан қосарлы сигнал екінші қабаттың тривиальды толтырғышы арқылы өзгеріссіз, екінші қабаттың тиісті синапсінен қос сигнал, j th ағынының түрлендіргіштен қос сигнал, синапстарға тарату үшін синапсиске жіберілді, тиісті синапстардың қос сигналдары бірінші қабат.



2.3-сурет – Екілік сигналдардың бір ағымда таралу құрылымы

2.4 Алгоритмдік бөлім

Алгоритм - бұл мақсатқа жету немесе осы мәселені шешу үшін нақты ережелерде орындаушыны ықшам түрде ұсынатын жүйеліктер тізбегі. Бұл анықтама алгоритм мен оның қасиеттерін байланыстыратын негізгі түсініктерді қамтиды. Сіз оны келесі схемада көрсете аласыз:

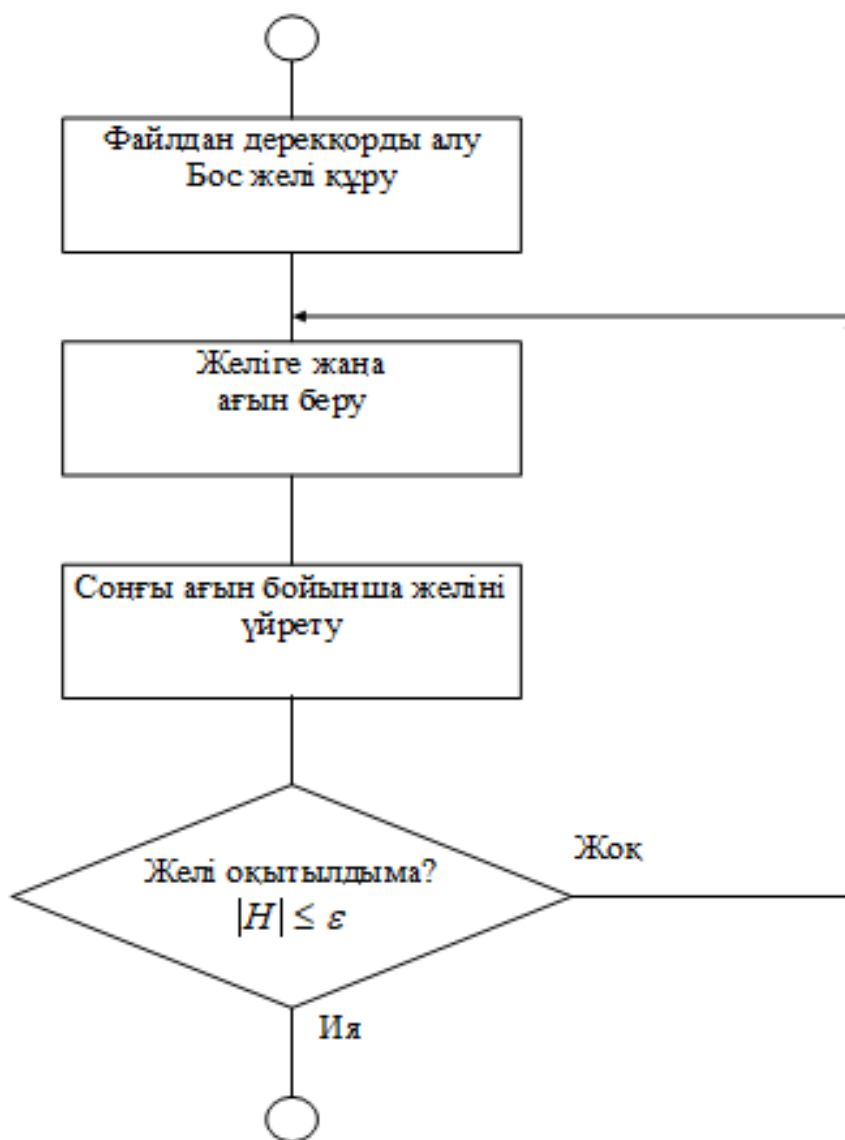
Бұл схемада орталық нысан алгоритмді орындаушы болып табылады. Орындаушы - жасалынған алгоритмді басқаратын объект (немесе объект). Суретшінің басты әрекеті командалық жүйенің басқаруында көрінеді. Ақауларды жоюға арналған кез келген шешімнен шығатын алгоритмдердің нұсқауы бар және нәтиже нәтижесі болады.

«Ағындардан» тұратын желіні дайындау, 13-суретте көрсетілген алгоритмге сәйкес жүргізіледі. Мұнда H - бұл бүкіл проблема бойынша жинақталған желінің бағалау шамасы, t - бұл қателік кішкентай тұрақтысы.

Алгоритм мұнда 5-ші суретте көрсетілген қолданылатын ағындарын Әрбір жаттығу үшін, N_b - карта желі параметрлерінің, h_0 - оқу, ол үшін функцияларды жалпы саны, қадам оңтайландыру ерте цикл, бұл - білім алушылардың параметрлерін бағалау функциясының H градиент.

Егер берілген нақтылыққа қол жеткізілсе, процесс тоқтатылуы мүмкін, әйтпесе - процесс алдымен қайталанады және оқытылған нейрондардың параметрлері бекітіледі, сондықтан әрбір жаңа нейрон бұрынғыдан қалған қателікті есептеуді үйренеді (2.4-сурет).

Ағымдық желіні үйретуде – барлық ағымдардың ең соңғы ағымы таңдалынып, ол класстарға сұраныс жібереді. Класстар желіні оқытады да, желі оқытылғанын немесе оқытылмағанын шығарады. Егер желі оқытылса, программа жұмыс тоқтатады. Кері жағдайда алгоритм басынан жаңа ағынға көшеді.



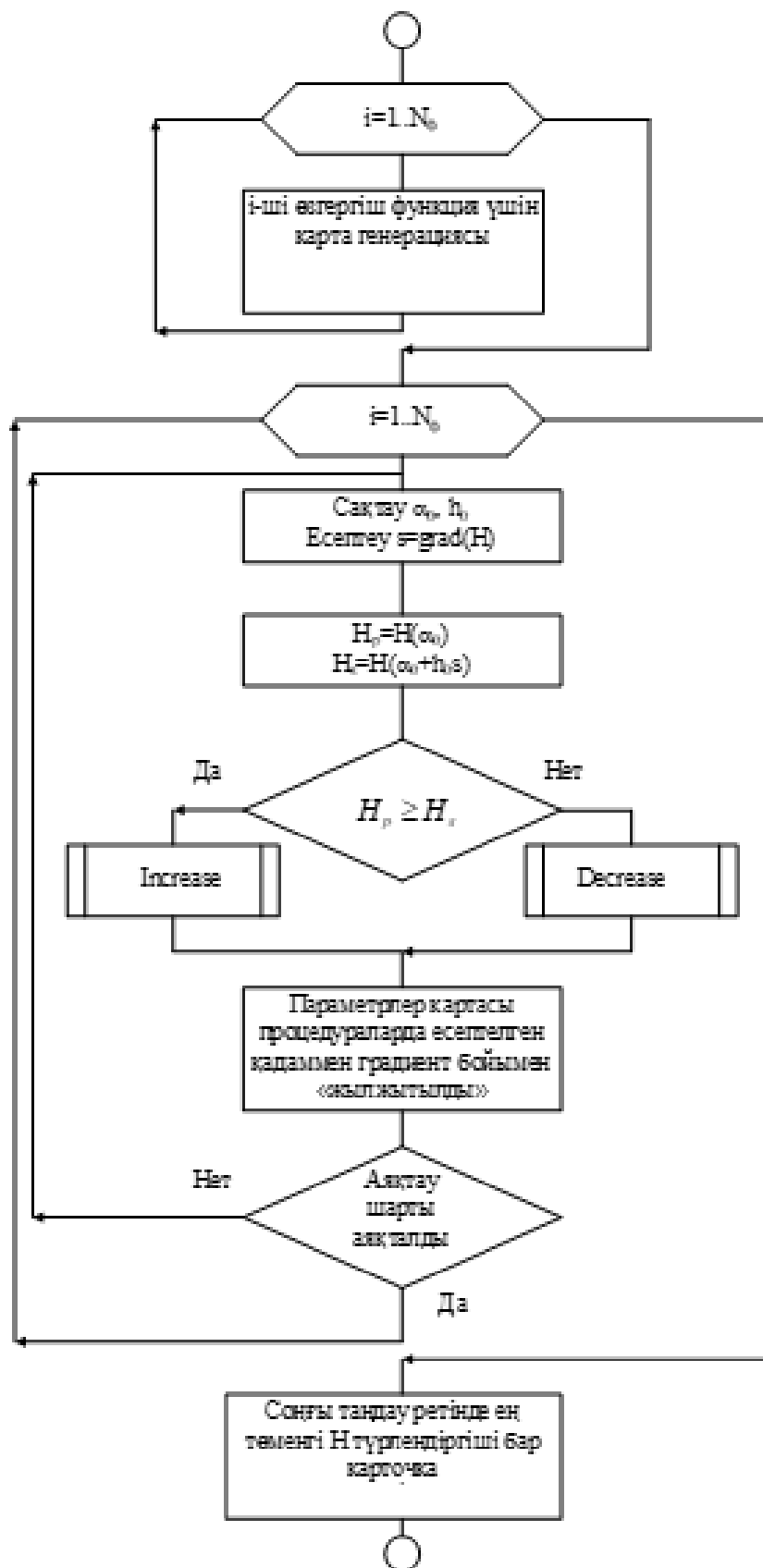
2.4-сурет – Ағымдық желінің үйрену блок-схемасы

Қатерлеу процесінің итерацияларының саны шектелген болуы мүмкін, бұл төменгі шекара Липшица тұрақты нейрондық желі үшін тұрақты Липшица тұрақты үлгідегі жоғарғы шегіне тең

Алгоритмде пайдаланылатын тоқтату күйі «немесе» арқылы біріктірілген екі қосалқы сөздерден тұрады:

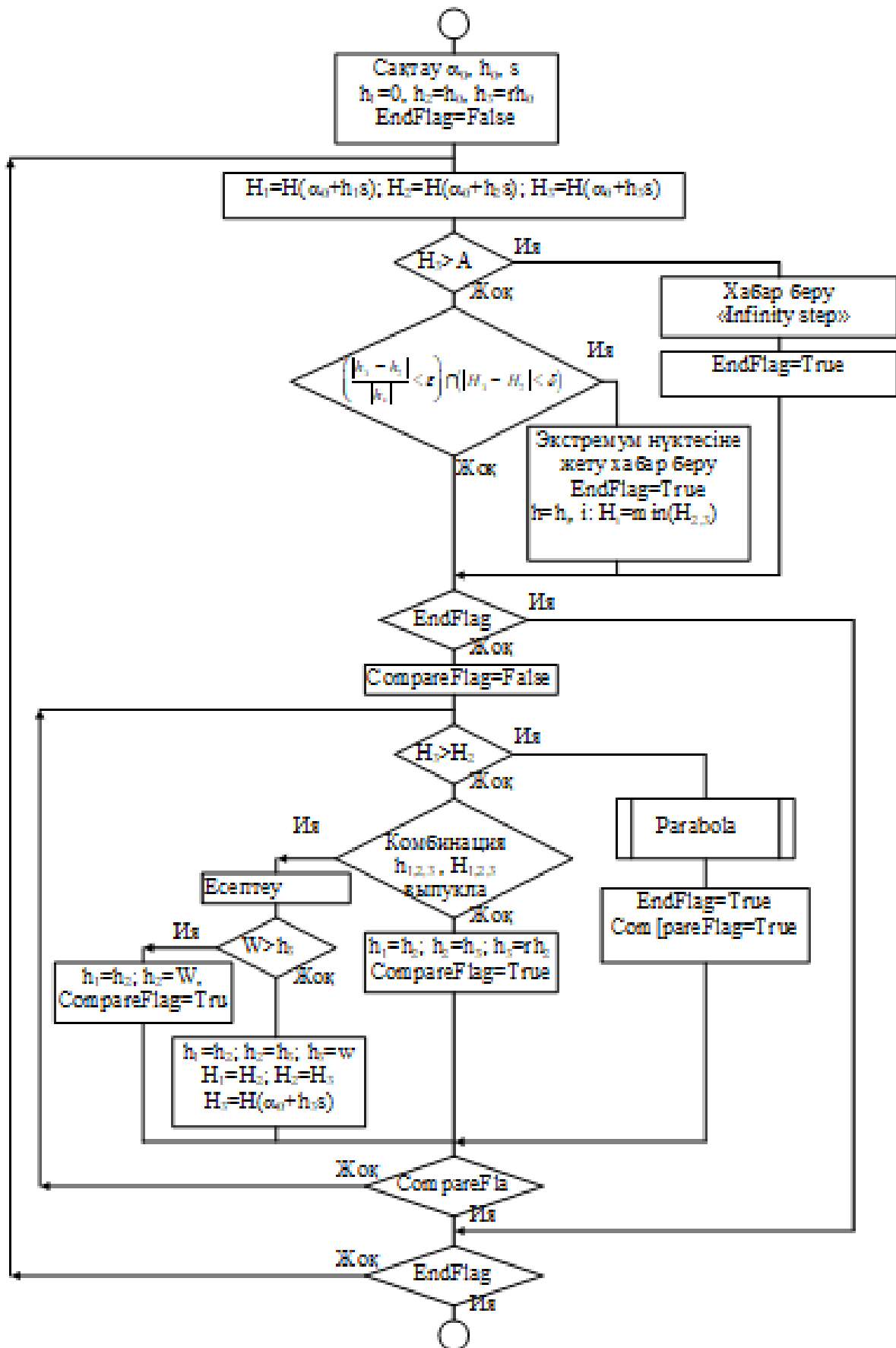
- алдын ала орнатылған мән асып кеткен (жаттығу $N_{sh} = 15$ мәнін пайдаланған) оңтайландыру қадамдарының саны, яғни бұл конфигурациядағы желі бағалауды жақсартпайды (2.5-сурет);

- бағалау функциясының алдын ала белгіленген мәніне, яғни желіге қол жеткізілді (2.6-сурет) [14];

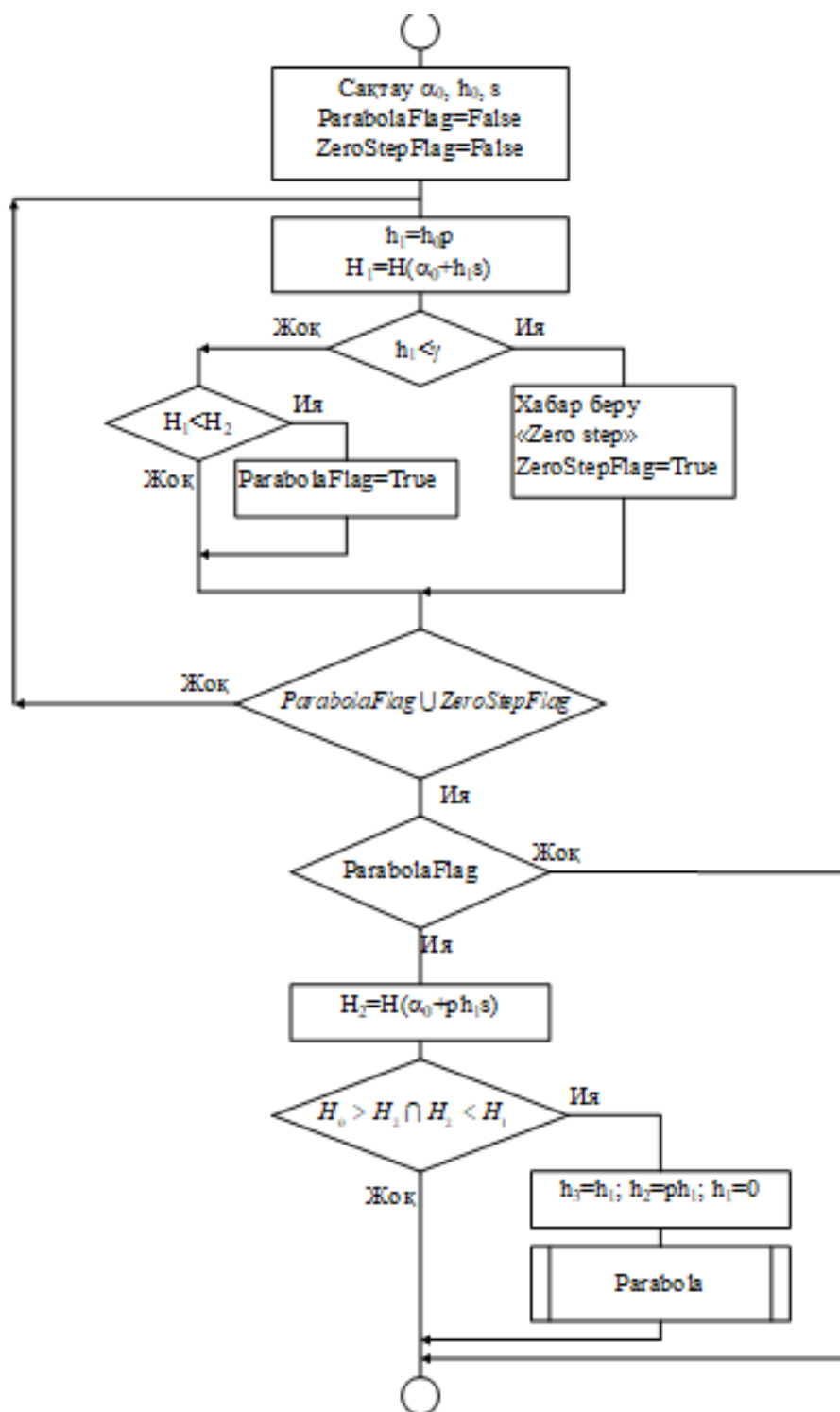


2.5-сурет – Бір ағымның үйрену алгоритмі

– нейрондық желі үйретілген.

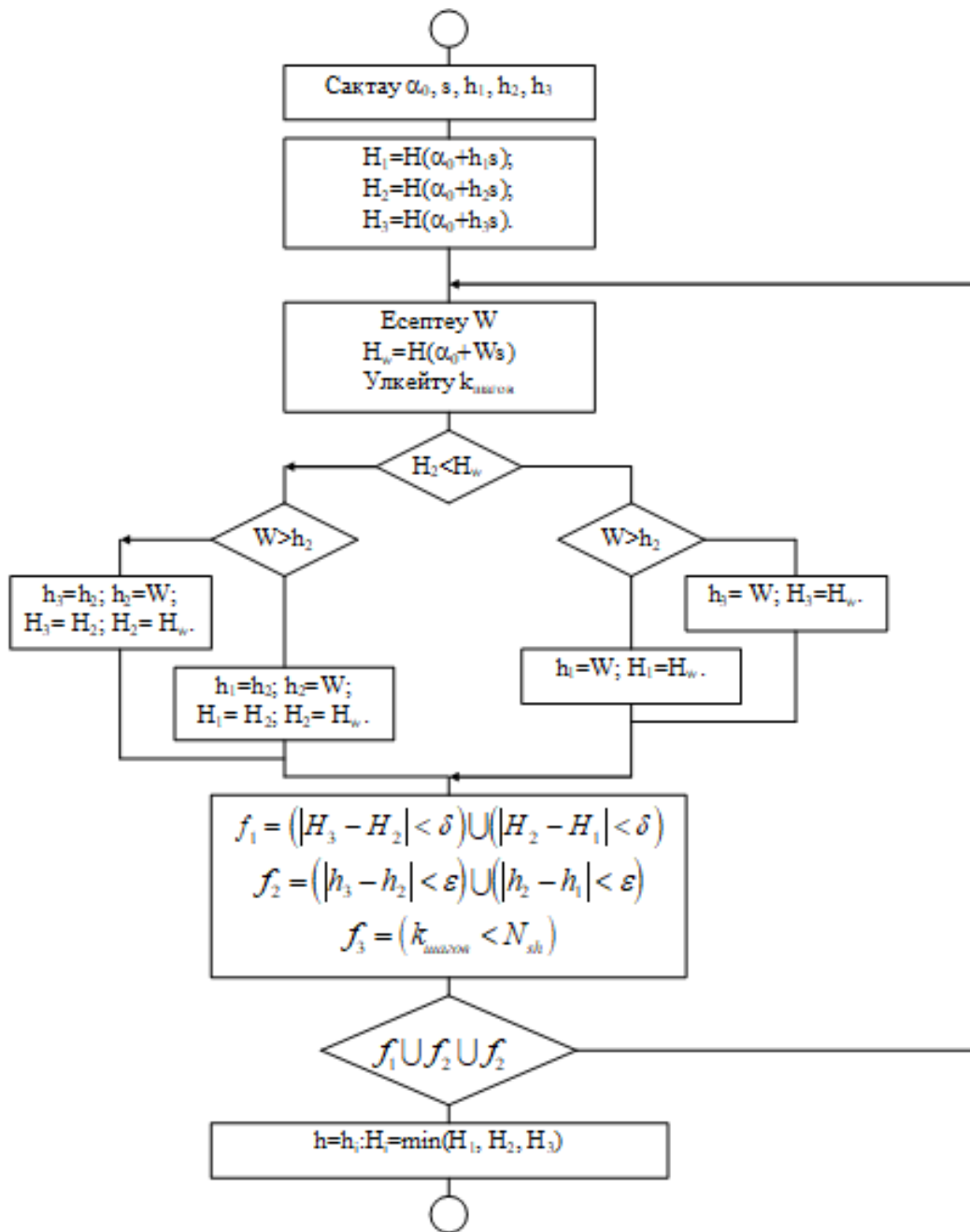


2.6-сурет – Increase процедурасының блок-схемасы



2.7-сурет – Decrease процедурасының блок-схемасы

Бір ағымды үйрену кезінде оңтайландыру қадамын таңдау процедурасы қолданылады (Іздеуді жоғарылату (жоғары қарай қадамды іздеу, 2.6-суреттегі блоктық схема), Төмендеу (төмендегіні іздеу, 2.7-суреттегі блок-схема) және Парабола блок-схема үшін параболикалық іздеу формулаларынан оңтайлы қадам, 2.8-суретті қараңыз) [15].



2.8-сурет – Parabola процедурасы

Төмендегі белгілер рәсімдерде қолданылады:

- $H(\dots)$ - бұл тапсырма кітабында жинақталған желінің бағалау функциясы;
- h_1, h_2, h_3 - оны таңдау кезінде пайдаланылатын оңтайландырудың қадамдық өлшемдерінің әр түрлі мәндері;
- W - параболаның шырлары $(h_1, H_1), (h_2, H_2), (h_3, H_3)$ арқылы өтетін қадам өлшемі. Формула бойынша есептеледі:

$$W = \frac{h_3 + h_1}{2} - \frac{H_3 - H_1}{\frac{H_3 - H_2}{h_3 - h_2} - \frac{H_2 - H_1}{h_2 - h_1}}$$

– H_1, H_2, H_3 - h_1, h_2 және h_3 қадам өлшемдері бойынша градиент бағытында оқу параметрлерін есептеуге сәйкес келетін қателік функциясының мәндері.

$H_{1,2,3}, H_{1,2,3}$ комбинациясының дөңес жағдайы (37) формуламен анықталады:

$$H_2 < \frac{h_3 - h_2}{h_3 - h_1} H_1 + \frac{h_2 - h_1}{h_3 - h_1} H_1 \quad (37)$$

Егер өрнек шын болса, дөңес жағдайды қанағаттандырады.

Енді желінің оқу алгоритмдерін қарастырғаннан кейін, біз компоненттердің сипаттамасына, желінің құрылымына және жұмыс істеуіне жүгінеміз.

2.5 Селективті Липшиц тұрақтығын пайдалана отырып, нейрондық желінің ақпараттық сыйымдылығын бағалау

Қатені кезең-кезеңмен сарқылу процесін тоқтату шарты нейрондық желімен анықталған функцияның толықтығын бағалауға негізделуі мүмкін. Желілік элементтердің саны орнатылса (әрбір қадамдық қадамда «ағынды» бұл да) және оның параметрлері мәндері белгілі бір интервалмен шектелсе (бұл шарт желі параметрлері бойынша шектеулер енгізсе), бұл жағдай Липшицтың тұрақты көмегімен құрастырылуы мүмкін. Дипомадағы Векторлық функцияның Липшиц тұрақтысы ретінде анықталады:

$$L_f = \sup_{x, y \in D, x \neq y} \frac{\|f(x) - f(y)\|}{\|x - y\|}$$

Жоғарғы шекара анықталу доменінен есептелуі мүмкін. Қашықтықты бағалау ретінде евклид норма пайдаланылады [16].

Векторлық функцияларды суперпозиция үшін:

$$(\varphi \circ f)(x) = \varphi(f(x)) \quad L_{\varphi \circ f} = L_\varphi L_f$$

Функциялардың сызықтық тіркесімі үшін Липшицтің тұрақты мәнін бағалау формулаға сәйкес жұмыс істейтін адаптивті жинақтағыш үшін Липшица тұрақтысы $L_\Sigma = \|\alpha\|$ түріне ие:

$$f = \sum_i \alpha_i f_i$$

$$L_f \leq \sum_i |\alpha_i| L_{f_i}$$

$$\alpha_0 + (x, \alpha)$$

Содан кейін кіріс синапстарының матрицасынан тұратын стандартты тіркесім үшін, жалдаушы мен түрлендіргіш - тегіс белсендіру функциясы бар нейрон Липшиц тұрақты мәнін бағалауға болады. Осылайша таңдаулы түрлендіргіштермен нейрондық қабат үшін нейронның кіріс сигналына түсетін синапстардың салмағы векторы, ал i -түрлендіргіш функциясы. Егер D функциясының түпкілікті жиынтығы (тапсырма) бойынша анықтаманың толық доменін ауыстыратын болсақ, онда нейрондық желінің қажетті көлемін анықтайтын жағдайды оқыту үлгісі үшін Липшиц тұрақты мәнімен салыстыру арқылы алуға болады

$$\varphi(y) L_N \leq \|\alpha\| \max_y |\varphi'(y)|$$

$$L_f \leq \left[\sum_i L_{f_i}^2 \right]^{\frac{1}{2}} L_s \leq \left[\sum_i \|\alpha^{(i)}\|^2 \max_y |\varphi_i'(y)|^2 \right]^{\frac{1}{2}}$$

$$L_{\%ohh} \geq \sup_{x, y \in \{x^P\}, x \neq y} \frac{\|f(x) - f(y)\|}{\|x - y\|} \quad (38)$$

$L_{\%ohh}$ жуықтауға болатын функцияның Липшицтер тұрақты мәнінің төменгі шекарасы. Нейрондық желі бұл функцияны тек $L_s > L_{\%ohh}$ болған кезде іске асыра алады.

Осы жағдайға сүйене отырып, біз 2.9-суреттегі блок-схемада көрсетілгендей, қатені кезең-кезеңімен сарқылуымен желіні үйрену алгоритмін құрастыра аламыз.

Желінің құрылымы мен жұмыс істеуіне қатысты мәселелерді баяндау тапсырманы іске асыру бағдарламасымен тығыз байланысты. Сондықтан мәтінде бағдарламаның нақты құрылымына сілтемелер болады [17].

2.6 Тесттік сұрақтардың маңыздылығымен желіні сәйкестендіру

Оқыту нейрондық желісін дәрежесін маңыздылығын бағалау үшін көрсеткіштердің есептеу бірге маңызды сигналдарды топтардың анықтау, ол оқыту мен нейрондық желінің кәсіби даярлау бойынша сигналдар экрандау үшін ерте сатысында мүмкін болады.

Оқыту басында маңызды болып шықты. Алайда, ол осы алгоритм деп опция қарсы сақтандыруға емес екенін атап өту керек, соңғы нейрондық желі

басым бола қоймайды. Осындай үй-жайдан шығып, автор ЛОБИ сұрақнамасының сигналдарының (сұрақтарының) маңыздылығын анықтады.



2.9-сурет – Сәйкесіншебіртіндеп сарқылу алгоритмі

Кезде барынша маңызы бар сигналдар түрін анықтау үшін маңызды болып табылады мәселелерді нөмірлерінің тізімін алынған, және ол фойесінде осы түрі үшін үлгідегі негізгі бірдей елеулі бөлігінде болып табылады. шамалы кірістерін кесімді кезде қызықты нәтижелерге алынды - желілік оқыту сапасы айтарлықтай жақсарды (2 желілері мұндай нәтижеге 95,24% және 90,48%, немесе 20 және 21-тест мысалы 19 дұрыс жауап алынды). Бұл нәтиже қандай қорытынды жасауға мүмкіндік береді?

Эксперименттік үлгілерін қажетті соманың өрескел сметаларын сынақ әдісін жасағанда, ол осы мәселелерді салмақ коэффициенттері іріктелген қажет есептеу үшін, «негізгі реттілігі» өлшемі N сұрақтар, егер мынадай $N * N$ мысалдар тәртібін болуы тиіс. Бұл қатынас туралы ғана (N - шамамен 15, $N * N$ - шамамен 200) сипатталған тәжірибеде орын алды. Алайда, бұл тест сұрақтары көптеген белгілері санын анықтау үшін арналған көп бөлігі үшін әдісі ретінде, Осы мінездің диагностикалау үшін қажетті болып табылады қарағанда әлдеқайда кең болуы, әдетте, бұл есте сақтау қажет. Демек, келесі мәселе туындайды: моделін параметрлерін анықтау үшін $M * M * M$ мысалдар қажет - сұрақтардың жалпы саны. қуатты ғылыми-зерттеу орталығының артында ие емес, бір - мысалдар саны іс жүзінде іс жүзінде мүмкін емес болып табылатын,

26244 жасау керек, ал, мысалы, лобби әдістерін үшін еске кез келген қажеттілігі, $M = 162$ бар.

Үлгіде жеткіліксіз мөлшерде болған жағдайда «жалған корреляция» құбылысы пайда болады – бұл модель шын мәнінде онымен нашар корреляциялаған параметрлердің шығуына әсерін анықтайды. Дәл осы «жалған корреляция», сыныпқа диагноз қойылған түрдің бар немесе жоқтығын сыныптауда қате жібереді.

2.7 Контрастты желілермен эксперименттердің нәтижелері

Келесі эксперименттер сериясында «контрастын» деп аталатын әдіс қолданылған. «MultiNeuron» нейроимитаторында кіріс сигналдарының бір бөлігін ажыратуға болады. Бұған оқу-жаттығу кезінде өзгерте алмайтын мүгедектердің кіруіне жауап беретін синапсқа -0 мәнін тағайындау арқылы қол жеткізіледі. Содан кейін бұл кіріс желідегі оқу үрдісіне әсер етпейді. Бұл экспериментте автор типтік диагностика үшін үйреншікті желілік күйге сәйкес келетін теңшеу параметрлері саласында минималды маңызы бар деректерді енгізуді жалғастырды. Сондықтан оларды өшірсеңіз, енгізу кеңістігінің өлшемі төмендейді, демек, оқу үлгісінің қажетті өлшемі. Қарама-қайшылықтан кейін қол жеткізілген оқу нәтижелерін түбегейлі жетілдіру бұл фактіні растайды, себебі жалпы ұсынымдардан алынған эксперименталды үлгі көлемі контрастты желіні үйренуге жеткілікті түрде дәлелденді.

Мәселен, жоғарыда келтірілген сұраққа жауап бере аламыз: кірістерді талдау және түзетуден кейін оқу нәтижелерін жақсарту контрастты жалған корреляциямен күресудің құралы екенін көрсетеді және, демек, көп өлшемді әдістер үшін эксперименттік үлгілерді азайтуға мүмкіндік береді.

Бұған қоса, емдеудің қосымша өнімі психолог-теоретиктерге тақырыпты бере алатын осы түрге қатысы бар сұрақтар жиынтығы болуы мүмкін.

2.7.1 Байланысты бағалау мәселесі

Персоналдың немесе бұрыннан бар топтардың шеңберінде зерттеуге қарым-қатынас іріктеу айналысатын тәжірибелік психологтар жұмысына (бұрынғы үлгісі кеңесшісі іріктеу, екінші үлгісі болып табылады - бөлімшелерінде қызметкерлерімен жұмыс істеуге офицер, мектепте сынып жетекшісі) үнемі құру мәселесін туындайды және тобында тұлға аралық қарым-қатынастарды болжайды.

Осы қағаз көзқарасы сәйкес (байланыс адамдардың өзге тұлға немесе топ болып табылады), адам психикалық білім пайда болып, оның мәні психологиялық құбылыс, жатады шындық нақты объектінің білім нәтижелерін жинайды, барлық интеграция осы нысанға эмоционалдық жауап, сондай-ақ мінез-жауап өткізді оған. Сонымен қатар, байланыс, әдетте белгілі бір жағдайға орын алады: кез келген эксперименттік факторлардың әсерінен, нақты қызмет

аясында, субъективті маңызы түрлі градус қарым бар басқа адамдардың қатысуымен.

2.7.2 Тәжірибелердің жалпы міндеті

Бұл зерттеу, егер мүмкін болса тест мемлекет пен мінез-құлық негізінде тобында қарым-қатынастардың жүйесін болжау, модельдеу және міндеті адам мақсаттары мен себептер жатқызылған сыртқы түрі ретінде, адамдар арасындағы қарым-қатынастар қалыптастыру, сол аспектілерін жағына қалдырып, болды. Болжам мен бағалау топтарында тұлғааралық «мәртебесі-рөлі» қарым-қатынастар ұшыраған. жеке (әрбір әрбір) және топ (тұлғаға тобының) - «адам-адам» үйлесімділігі бағалау және «тұлғалар тобы» сынақтың күйін бағалау үшін жүргізілді.

Топтың (команда) адамдардың қарым-қатынас модельдеу және болжау міндеті бірыңғай емес - ол шартты түрде мынадай подзадач бөлуге болады:

- зерттеушінің белгіленген ұжымға кіру болжамын;
- екі зерттеуші арасындағы үйлесімділік болжамы.

Сонымен қатар, эксперимент тұлғааралық интуитивті тәсілі сипатталған (сипаттама) шындыққа шығару болжау жасау айналып күтілетін болжау мәселені тексеру үшін.

2.7.3 Тәжірибелерде пайдаланылатын психологиялық әдістер

Зерттеу топтарындағы нақты қарым-қатынастарды анықтау үшін социометриялық әдіснама қолданылады. Бұл әдіс сізге тиесілі топтың тұлғааралық қарым-қатынасының позициясын анықтауға мүмкіндік береді. Топты социометриялық зерттеу әдетте кем дегенде 10 адамнан тұратын және кемінде бір жыл болған кезде жүргізіледі.

Зерттеу тобының барлық мүшелері серіктестерінің әрбір бағалауға сұралады (өзі, оның ішінде - өзін-өзі оқуға мүмкіндік). бағалау стандартты әдісі преференциялар үш сатылы шкаласы бойынша жүзеге асырылады - «объект - немқұрайлы - қабылдамау» Дегенмен, техниканың жоғары ажыратымдылығын алу үшін, шкала он шкаланың шкаласына өзгертілді. «Достарыңызды бағалаңыз өзіңнен:»? Мен бір топта сол адаммен жұмыс істеуге қалар еді қанша «sociometric зерттеулер пайдаланылған нұсқа келесі тапсырманы қолданылады. мынадай жолмен 1 және 10 нүктелер арасындағы тиісті терезесі есебімен орны: 1 - Мен онымен ештеңе, 10 болуы келмейді, - деп адаммен Мен қиын жұмыс істеуге келеді.

Топтағы тақырыптардың әрқайсысы бойынша зерттеудің нәтижесі мәртебені және кеңейтуді қабырғалық бағалау болып табылады. Бағалау қалыпты бөлу арқылы таратылады, сондықтан, әдетте, «үш сигма» жүзеге асырылады деп болжанса металық қалыпқа құрайды. Бұл мәртебе топтың кез-келген мүшесіне жасалатын барлық бағалаудың қабырғалық бағасына қатысты,

кеңею - топтың барлық басқа өкілдері осы тақырыпқа жасалған барлық бағалаудың қабырғалық бағалауы.

Растау (немесе қабылдамау) арналған эксперименттер нейрондық желі тобында қатынастарды модельдеу және топтың жаңа мүшесін енгізу туралы және екі жеке тұлғалар арасындағы қарым-қатынас туралы болжамын беруге халықтың (топтың мүшелері) психологиялық сипаттамалары негізінде береді деп гипотеза барысында. Сондай-ақ, болжамдардың сапасын - ықтимал қателіктерді және олардың таралуын бағалайды [18].

Бұл жобаны құруда кері таралу қатесі әдісі қолданылды. Негізі ретінде $10 \times 10 = 100$ элементі бар массив алынды.

Жобаны MS Project-те күндер бойынша орындалуының көрсетілімі 2.10-суретте берілген.

	Режим задачи	Название задачи	Длительность	Начало	Окончание	Предшествен
1		Материалдарды оқып үйрену	5 дней	Пн 02.04.18	Пт 06.04.18	
2		Форманы құру	3 дней	Пн 09.04.18	Ср 11.04.18	1
3		Интерфейсін жасау	1 день	Пн 09.04.18	Пн 09.04.18	1
4		Форманы кодтау	1 день	Вт 10.04.18	Вт 10.04.18	3
5		Өзгертулер енгізу	1 день	Ср 11.04.18	Ср 11.04.18	4
6		Класстарды құру	9 дней	Чт 12.04.18	Вт 24.04.18	5
7		NeiroGraphUtils.c классын құру	2 дней	Чт 12.04.18	Пт 13.04.18	5
8		Neiron.cs классын құру	2 дней	Пн 16.04.18	Вт 17.04.18	7
9		NeiroWeb.cs классын құру	3 дней	Ср 18.04.18	Пт 20.04.18	8
10		Program.cs классын құру	1 день	Пн 23.04.18	Пн 23.04.18	9
11		ShowMemoryVeig формасын құру	1 день	Вт 24.04.18	Вт 24.04.18	10
12		Өзгертулер енгізу	2 дней	Ср 25.04.18	Чт 26.04.18	11
13		Тестілеу	1 день	Пт 27.04.18	Пт 27.04.18	12
14		Қорытындылау	1 день	Пн 30.04.18	Пн 30.04.18	13

2.10-сурет –Жобаны құруға кеткен уақыт

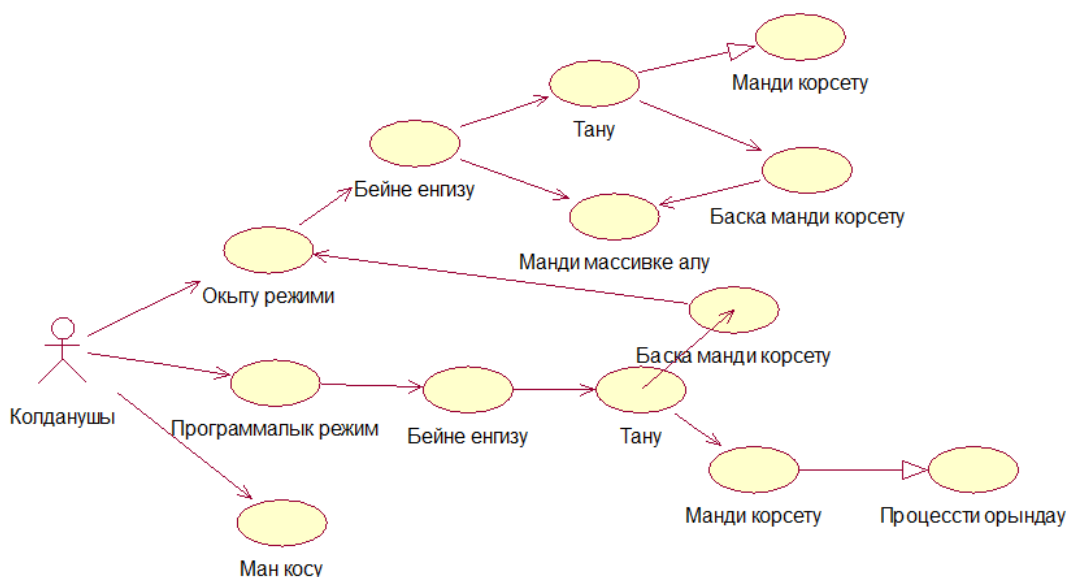
Rational Rose – та жасалған программаның жұмыс істеу алгоритмінің прецеденттер (2.11-сурет) және тізбектер диаграммасы (2.12-сурет) құрастырылған.

Класс NeiroGraphUtils – бұл класста кескінді түрлендіру функциялары жиналған. Бұл класстың қызметтері:

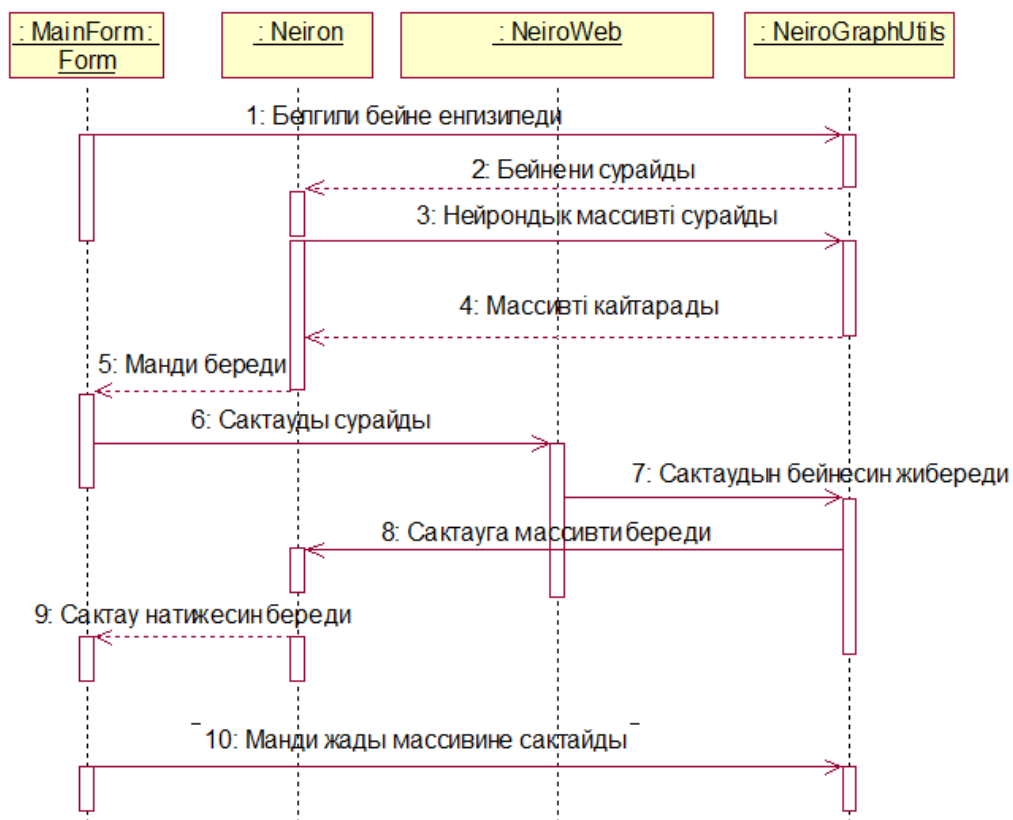
- суретті массивке түрлендіреді, ақ түстен басқа барлық түстер 1, ақ

- О деп жазылады;

- сызбалық суретте жол белгісін басып шығарады;
- массивті сызбаға түрлендіреді;



2.11-сурет – Прецеденттер диаграммасы



2.12-сурет – Тізбектер диаграммасы

- кескінді жиектерге қып, массивке түрлендіреді;

– жиым көзін массивте қайта құрастыру үшін - деректердің массивтерін стандартты өлшемдер жиынынан алу үшін пайдаланылады;

– суретті тазартады.

Класс `NeiroWeb` – бұл, негізінен, нейрондық нейрон массиві үшін контейнер болып табылады. Архив файлдан жасаған кезде оны жүктеп, шығу кезінде сақтайды. Бұл класстың қызметтері:

– мәтіндік файлды ашады және оны нейрондық массивге түрлендіреді;

– деректер құрылымын нейрон класына айналдырады;

– кіріс массивін желідегі әрбір нейронды салыстырады және нейронның атауын оған ұқсас етіп қайтарады, бұл класс кескінді тану үшін жауап береді;

– файлдағы нейрондық массивтерді сақтайды;

– жадыдағы суреттер атауларының тізімін алады;

– деректер бейнесінің жаңа нұсқасын нейрондық жадқа трассалық атаумен енгізеді.

Класс `Neiron` – бұл нейрондық класс, әрбір нейрон белгілі бір кескіннің массасын сақтайды, оны үйренуге және есте сақтау қабілетімен салыстыруға болады. Бұл класстың қызметтері:

– нейронның жадысын өшіріп, оған жаңа ат береді;

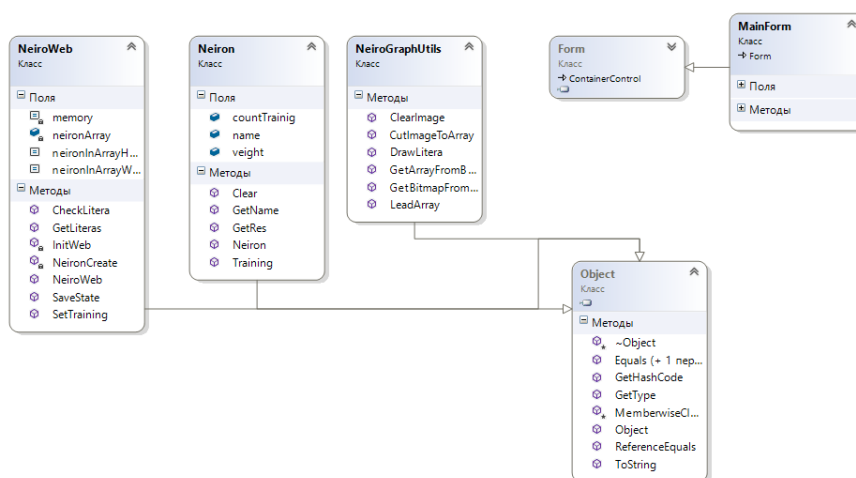
– функциясы кіріс массивінің ауытқу мәндерінің сілтемеден басталатын мәнін қайтарады, яғни нәтиже 1-ге жақын болса, кіріс жиілігі нейронды болады;

– жиым жадына кіріс кескінін қосады;

– зерттеу санын қайтарады.

Класс `MainForm : Form` – басты батырмалы форма. Бұл жерде барлық класстарға ссылақ беріледі сол арқылы процесс жүзеге асады. Процедура мәндердің тізіміне жол береді.

Класстардың байланысы 2.13-суретте көрсетілген.



2.13-сурет – Класстар диаграммасы

Бастапқы форманы құруда PictureBox, Button, MenuStrip, TextBox, Combobox, Label, Calendar қолданылды:

а) PictureBox – Форманың барлық бетін алады және бейнені қабылдайды;

б) Button – 3 негізгі батырма бар:

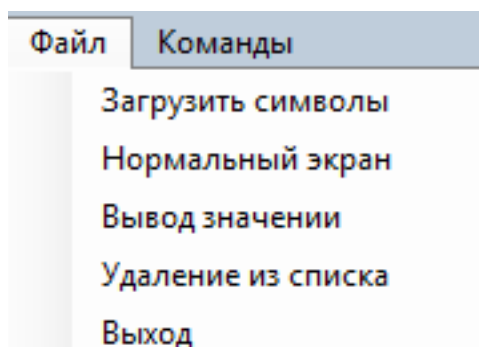
- 1) қосу – жаңа мәнді массивке қосады;
- 2) суретті өшіру – қате терілген бейнені алып тастайды;
- 3) тану – берілген бейнені танып мәнді шығарады (2.14-сурет).



2.14-сурет – Батырмалар

в) MenuStrip – Екі негізгі меню бар Файл және Команды. Файлда:

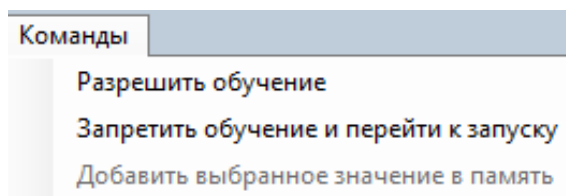
- 1) мәнді жүктеу – мәнді тектілік файлдан алады;
- 2) нормалды экран – picture папкасындағы фондық суреттерді
- 3) өлшемін экранның өлшеміндей қылдырып, picture1 папкасына
- 4) салады;
- 5) мәнді шығару – Combobox-тағы мәнді out.txt файлына шығарады;
- 6) мәнді өшіру – Combobox-тағы мәнді өшіреді;
- 7) шығу – программадан шығады (2.15-сурет).



2.15-сурет – Файл менюі

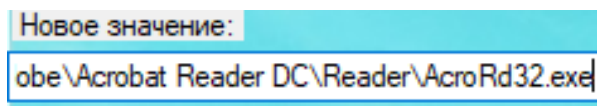
г) команды менюінде келесі батырмалар бар:

- 1) оқыту режимі – жазылған бейнені массивтік жадыға сақтауға
- 2) рұқсат береді;
- 3) программалық режим – программалардың процесстерін;
- 4) орындайды;
- 5) жаңа бейнені қосу – массивтік жадыға мәнді енгізеді (2.16-сурет).



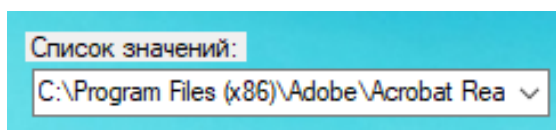
2.16-сурет – Команды меню

д) Textbox – Енгізілген абсолютті жолды Combobox – қа жібереді (2.17-сурет);



2.17-сурет –Textbox жаңа мәні

е) Combobox – Textbox-тан келген мәнді массивке салады, енгізілген бейнені жадыда ұстайды (2.18-сурет);



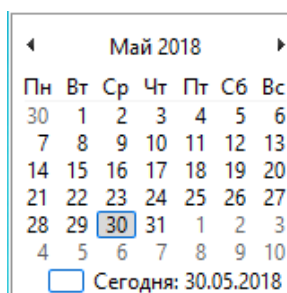
2.18-сурет – Combobox-та мәнің көрсетілімі

ж) Label –негізгі функционалы сағат үшін қолданылды (2.19-сурет);



2.19-сурет – Сағаттың макеті

и) Calendar – Visual Studio – ның өзінің күнтізбесі (2.20-сурет).



2.20-сурет – Күнтізбе

Қорытындылай келгенде, тану нейрондық желінісін құру үшін:

- MS Project пайдалана отырып, жобаны күндер бойынша бөлінді; программаның жұмыс істеуі Case және байланыстар диаграммасымен құрылды;

- программалық тіл ретінде C# және программалау ортасы ретінде Microsoft Visual Studio 2015 таңдалынды;

- класстар диаграмма құрастырылып, 3 класс және басты батырмалы форманың классын кодталды;

- 50-ға жуық мән алынды;

- негізгі интерфейске сағат және күнтізбе енгізілді.

Нейрондық желі бейнелерді тану арқылы белгілі мәндерге сәйкес процесстерді ашу үшін арналған. Оның мұғалімі ретінде қолданушы болады. Сол арқылы әр адам программаны өз ыңғайына сай келтіре алады. Сонымен қатар бір компьютерден екінші компьютерге мәндердің экспорты қарастырылған.

3 Нейрондық желілік компоненттер

Желілік элементтердің дәстүрлі құрамы мынадай элементтерді қамтиды: нейрон, синапс, сіңіргіш. Сонымен қатар, типтік кіріс және шығыс элементтерінің саны енгізілген.

2.9-суретте бағдарламалық модульдің құрылымы мен жұмыс істеуін сипаттайтын диаграммаларда пайдаланылатын желілік элементтердің схемалық суреттері көрсетілген.

3.1 Нейрондық желінің жалпы элементі

Нейрондық желі сипаттау үшін пайдаланылатын базалық түрі элементтері жалпы желілік элементі болып табылады – барлық желілік компоненттері үшін ортақ болып табылады негізгі қасиеттері мен әдістерін, инкапсулирует сынып. Бағдарламаны іске асырудағы осы сыныптың атауы TnetPiece болып табылады.

Осы сыныптың объектілері келесі өрістерді қамтиды:

- NextPiece – келесі желі элементіне көрсеткіш;
- PriorPiece – желінің алдыңғы элементіне көрсеткіш;
- ForwOut – алға сигнал аудару мүшесі құны алға операция;
- BackOut – кері пайдалануға сигнал таратушы элементтің мәні.

Әдістер жиынтығы мыналарды қамтиды:

- жасау – объектіні құру сипаттамасы;
- Destroy – объект жойылған кезде (жойылған) әрекеттер;
- ForwardTact – тікелей әрекет ету тактикасы кезінде элементтің әрекеттері;
- BackwardTact – Өлшем қайтару операциясы барысында іс-қимыл элементі;
- әдістері мен ForwardTact BackwardTact сипаттайтын кезде бетон элементтерді желісінің жұмыс істеуін айтарлықтай ерекшеленеді, олар, бос қалды.

Дегенмен, осы әдістердің енгізілуі TnetPiece сыныбы желі элементтерін сипаттайтын барлық басқа сыныптардың ата-анасы болып табылады және тікелей және кері әрекет ету үшін стандартты процедуралардың болуы объектілерге бағдарланған бағдарламалау үлгісінің қасиеттерін және әдістерін және полиморфизмді мұра ретінде қолдану мүмкіндігін береді. Бұл тезис төменде талқыланады [19].

3.2 Нейрондық желінің кіруі

Желіді тапсырмалар кітабымен қосуға және TnetInput класының объектілерін тасымалдау үшін – желінің кіріс элементі қолданылады.

Бұл сынып TnetPiece-тің ұрпағы болып табылады, сондықтан осы сыныптың өрістері мен әдістерінің жиынтығын иеленеді, сонымен қатар, осы желі кірісі мәнді таңдайтын тапсырма өрісінің санын қамтитын SourceSignal өрісі қосылады.

ForwardTact және BackwardTact әдісі жабық, яғни олардың коды тағайындалған элементтің тағайындалған орнына сәйкес келетін бірімен ауыстырылады.

ForwardTact әдісі тиісті тапсырма элементінен мәнің элементтің шығу сигналына, ForwOut өрісіне өткізілуін орындайды.

BackwardTact әдісі келесі элементтің қос сигналын оның қос сигналына (BackOut өрісі) өтеді [20].

3.3 Нейрондық желіні шығару

Желінің шығу элементі TnetOutput сыныбын, сондай-ақ TnetPiece тобының ұрпағын сипаттайды.

ForwardTact және BackwardTact әдістерін алға және кері операциялық циклдары бар элементтің әрекеттері бар.

ForwardTact әдісі алдыңғы элементтің шығуынан осы элементтің шығуына сигнал беруді жүзеге асырады, сонымен қатар желінің қателігі Y функциясын есептеу кезінде H өрісінде есептеледі.

BackwardTact әдісі қос сигналдың мәнін элементтің қайтару мәніне (BackOut өрісі) өтеді. Екі сигнал $\square H$ – желінің шығуындағы қателік функциясының туындысы:

$$\mu_H = \frac{\partial H}{\partial Y_{-H} \square} = 2(Y_{-H} \square - F \cdot p) \quad (39)$$

мұндағы, Y_{setu} – желі арқылы қайтарылатын функцияның шамамен шамасы;

F_{np} – осы мысалға жақындатылатын функцияның мәні.

TnetSynapse сынып нысандары желі синапстарын сипаттау үшін қолданылады. TnetPiece сыныбының мұрагері ретінде ол барлық өрістер мен әдістерді иеленеді. Сонымен қатар, Альфа өрістердің тізіміне енгізілген, синапс салмағын білдіретін параметр, ал MuAlpha – бұл синапс салмағына қосарлы сигнал [21].

Тікелей жұмыс істеу сағатында ForwardTact әдісі алдыңғы элементтің шығуын жояды, оны синапсаның салмағына көбейтеді және оны осы нысанның шығу сигналына (ForwOut өрісі) өткізеді.

Кері операциялық циклде BackwardTact әдісі BackOut өрісіне қос форматтағы синапс сигналын береді, ол келесі формула (40) бойынша есептеледі:

$$\mu_w = Y_{back} \frac{dW}{dx_w} = Y_{back} \alpha_w \quad (40)$$

3.4 Тривиальды жалдаушы

Тривиальды жалдаушының бағдарламалық моделі – Tsummator класымен қосымша сыныптан мұраға – ата-баба TnetPiece, Tsummator стандартты далалық PriorPiece айырмашылығы алдыңғы элементіне сілтегіш, және осындай элементтерінің жиынтығы көрсеткіштер тізімі емес болып табылатын, оның құрылымы PriorPieces бар.

ForwardTact әдісі тізімі PriorPieces элементтердің шығыс сигналдарының қосылыс түзу ерекшелігі орындайды және ForwOut саласындағы нәтиже орналастырады.

Қос сигнал кері циклі пайдалану бойынша келесі элементі қос BackOut сигнал сумматор беріледі [22].

3.5 Нейрондық желілік ағын

Енгізу қабатынан тұратын желісінің Detail, арифмометр синапстар, және шығыс нейрон синапстардың және атындағы «ағыны» бағдарламалық қамтамасыз ету моделі сынып TnetStream ұсынылған.

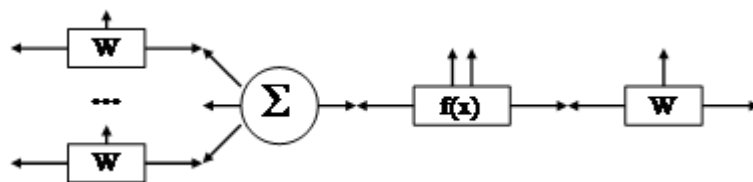
Әдеттегі ұрпағы класс TnetPiece кен NextPiece, PriorPiece, ForwOut және BackOut қатар осы класс FirstLayer кіреді – бірінші қабатының тізімі синапсов сумматор – нейронның тривиальды арифмометр жүзеге асыру нысаны Tsummator класс – сызықты түрлендіргішті іске асыру нысаны Tneuron класс – нейрондық, SecondLayer – шығыс ағыны синапстардың білдіретін Tsynapse класс объектісі төмендегідей әдіс ForwardTact сипатталған Тікелей ағыны цикл болып табылады:

FirstLayer тізімінің элементтері сұрыпталады, олардың әрқайсысы үшін ForwardTact әдісі шақырылады, содан кейін Summator, Neuron және SecondLayer нысандары үшін «триггер» (сол әдісті шақыру) орын алады. Содан кейін нысанның шығу сигналы ForwOut өрісінде – бүкіл ағынның шығуына жазылады.

BackwardTact әдісінде сипатталған кері ағым функциясы мыналарды қамтиды:

Sequentially SecondLayer, Neuron және Summator нысандары үшін BackwardTact әдісін шақырыңыз, содан кейін FirstLayer тізім элементтері арқылы айналдырыңыз және оған қосылған әрбір синапсаға арналған BackwardTact әдісін шақырыңыз.

Ағым элементтері арасындағы қатынастардың құрылымы 3.1-суретте көрсетілген [23].



3.1-сурет – Типтік элементтердің ағын желісіндегі байланыс құрылымы

3.6 Композициялық бір жарым қабатты желі

Бүкіл желі Tnet сыныбы ұсынылған. Желі ортақ элемент, неғұрлым күрделі құрылымын нейрондық желінің жекелеген фрагменттерінің бастап жиналуға жеткілікті оңай полиморфизм сайттар сипатты пайдалану мүмкіндік береді – Осы сыныпқа сондай-ақ TnetPiece сыныбында болып табылады.

Әрі қарай өрістер желі кірулер өрістерін, сондай-ақ сипатталады, TnetPiece мұраға қалған – желілік ағындарын тізімінде, SecondLayer - - кіріс элементтерін, жылғалар тізімін сумматор кіріс қабаты, Жауап – шығыс желі элементін.

Тікелей желі жұмысының тактикасын сипаттау үшін, ForwardTact әдісі, сондай-ақ желінің басқа элементтерінде қолданылады. Әдіс келесі әрекеттерді орындайды:

Кіріс элементтерінің тізімін аударуға және тізімі жіптерді элементтерін әрбір операцияны іздеу және бағыттау, олардың әрқайсысы үшін ForwardTact әдісі қоңырау, содан кейін – SecondLayer және нысандарды жауап.

Желінің кері жұмыс істеуін сипаттайтын BackwardTact әдісі келесі әрекеттер ретімен анықталады:

Жауаптың, SecondLayer элементтерінің кері жұмыс істеуі, одан кейін ағымдар мен кірістер тізімдерінің элементтері үшін BackwardTact әдістерін қайталайды және орындайды.

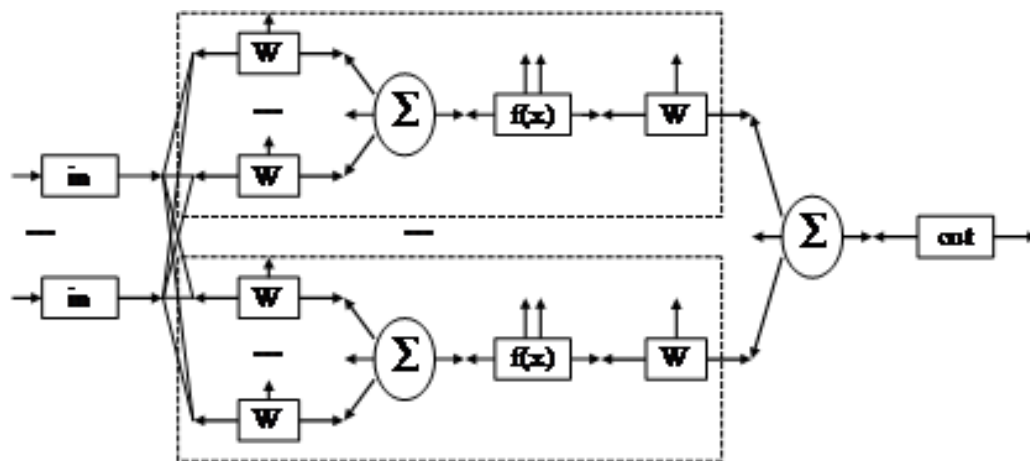
Толық жинақталған нейрондық желінің стандартты элементтері арасындағы байланыстардың құрылымы 3.2-суретте көрсетілген.

Қорытынды суммасы шығыс нейронында белгіленеді. Әр нейрон жеке есептеледі. Алдыңғы нейронның шығысы келесі нейронның кірісі болады. Сол арқылы нейрондық желі барлық нейрондардың мәнін есепке алады [24].

Соңғы тарауда үнемі елеулі жеңілдетулер туралы айттым. Жеңілдетудің себебі – қазіргі заманғы компьютерлердің біздің миымыз сияқты күрделі жүйелерді жылдам модельдеуі мүмкін емес. Сонымен қатар, менің айтқанымдай, біздің миымыз ақпаратты өңдеуге қатысы жоқ түрлі биологиялық механизмдерге толы.

Кіріс сигналын қалаған нәтижеге түрлендіруге арналған үлгі қажет. Барлық басқа нәрсе бізді алаңдатады. Біз жеңілдете бастаймыз.

Биологиялық құрылымы → схемасы



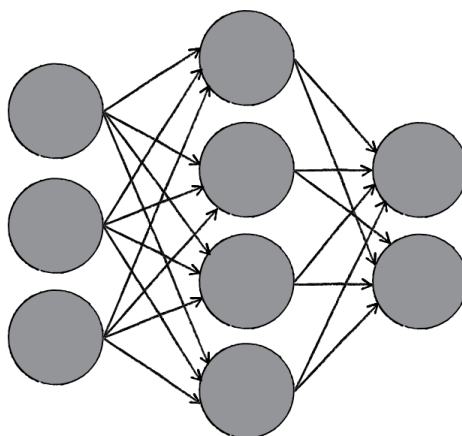
3.2-сурет – Типтік элементтердің нейрондық желідегі құрылымы

Алдыңғы тарауда биологиялық нейрондық желілер мен биологиялық нейрондардың қаншалықты қиын екенін түсіндіңіз. Нейрондарды тентақшылармен бейнелеудің орнына диаграммаларды сызайық.

Жалпы айтқанда, нейрондық желілер мен нейрондарды графикалық ұсынудың бірнеше жолы бар. Мұнда жасанды нейрондарды шеңбер түрінде бейнелейтін боламыз.

Кірістер мен шығындардың күрделі интерполяциясының орнына біз сигналды қозғалыс бағытын көрсету үшін көрсеткілерді қолданамыз.

Осылайша, жасанды нейрондық желі көрсеткілерді қосатын шеңберлердің (жасанды нейрондар) жиынтығы ретінде ұсынылуы мүмкін(3.3-сурет) [25].



3.3-сурет – Электр сигналдары → сандар

Нақты биологиялық нейрондық желіде электрлік сигнал желідегі деректерден шығысқа дейін беріледі. Нейрондық желі арқылы өту кезінде ол өзгеруі мүмкін.

Электр сигнал әрдайым электрлік сигнал болады. Тұжырымдамада ештеңе өзгермейді. Бірақ содан кейін не өзгереді? Бұл электр сигналының

шамасы өзгереді (күшті / әлсіз). Және кез-келген мән әрдайым санмен көрсетілуі мүмкін (көп / аз).

Жасанды нейрондық желінің үлгісінде біз электр сигналының мінез-құлқын жүзеге асырудың қажеті жоқ, өйткені ол оны іске асыруға байланысты болмайды.

Желінің кірісінде біз электр сигналының шамасын білдіретін кейбір нөмірлерді береміз. Бұл сандар желі арқылы қозғалады және қандай да бір түрде өзгереді. Желінің шығуы кезінде біз желінің жауабы болып табылатын кейбір нәтиже санын аламыз.

Ыңғайлы болу үшін, біз желіде, сигналдармен айналысатын нөмірлерге әлі де қоңырау шалдық.

Синапстар → байланыс салмағы

Нейрондар – синапстар арасындағы байланыстар түсімен бейнеленген бірінші тараудағы суретті еске түсірейік. Синапса олар арқылы өтетін электр сигналын күшейтеді немесе азайтады.

Әрбір осындай қосылысты берілген байланыс салмағын атаған белгілі бір сан бойынша сипаттайық. Бұл байланыс арқылы берілетін сигнал тиісті сілтеменің массасына көбейтіледі.

Бұл жасанды нейрондық желілер тұжырымдамасындағы басты мәселе, оны толығырақ түсіндіріп беремін. Төмендегі суретке қараңыз. Енді осы суреттегі әрбір қара көрсеткі (қосылым) үшін белгілі бір w_i (байланыс салмағы) санына сәйкес келеді. Егер сигнал осы байланыс арқылы өтіп кетсе, оның мәні осы қосылымның салмағына көбейтіледі [26].

Жасанды нейрон. Енді жасанды нейронның ішкі құрылымын және оның кіріс сигналын қалай өзгерткенін қарастырамыз.

Кірістер, салмақтар және тартқыштар

Әрбір нейрондық, оның ішінде жасанды нейрон, сигналды алатын кейбір кірістер болуы керек. Біз байланыс арқылы өтетін сигналдар көбейтілген салмақтар тұжырымдамасын енгіздік. Суретте жоғарыдағы салмақтар топтармен ұсынылған.

Кіріс сигналдары олардың салмағымен көбейтіледі. Бірінші кірістің x_1 сигналы осы кіріске сәйкес w_1 салмағына көбейтіледі. Нәтижесінде $x_1 w_1$ аламыз. Міне, n -ші кіреберіс. Нәтижесінде, соңғы енгізу кезінде $x_n w_n$ аламыз.

Енді барлық жұмыстар жалдаушыға берілді. Тіпті оның есімінің негізінде сіз не істейтінін түсінесіз. Ол барлық кіріс сигналдарын тиісті салмақтармен көбейтеді (41):

$$x_1 w_1 + x_2 w_2 + \dots + x_n w_n = \sum_{i=1}^n x_i w_i \quad (41)$$

Қайталанатын / бірдей типтік терминдердің жиынтығынан тұратын үлкен өрнекті қысқаша жазу қажет болғанда, сигма белгісін қолданыңыз.

Жазбаның қарапайым нұсқасын қарастырыңыз:

$$\sum_{i=1}^5 i = 1 + 2 + 3 + 4 + 5$$

Осылайша, σ төменгі бөлігінен біз контрперенный i айнымалы мәніне бастапқы мән береміз, ол жоғарғы шекке жеткенше жоғарылайды (жоғарыдағы мысалда бұл 5).

Жоғарғы шама да айнымалы болуы мүмкін. Мен осындай мысал келтіремін.

N дүкендері болсын. Әр дүкеннің өз нөмірі бар: 1-ден бастап n . Әрбір дүкен пайда әкеледі. i -ші дүкенін (қандай да болмасын) алыңыз. Оның пайдасы пиге тең.

Егер біз барлық дүкендерден жалпы пайда табуды қаласақ (біз оны P арқылы белгілейміз), онда ұзақ соманы (42) жазуымыз керек еді:

$$P=p_1+p_2+\dots+p_i+\dots+p_n \quad (42)$$

Көрінетін болсақ, бұл соманың барлық мүшелері бірдей. Содан кейін олар төмендегідей қысқаша жазылуы мүмкін:

$$P=\sum_{i=1}^n p_i$$

«Бірінші дүкеннің пайдасын қорытындылаңыз және біріншіден бастап n -th –пен аяқтаңыз». Формула түрінде бұл әлдеқайда қарапайым, ыңғайлы және әдемі.

Төлеушінің нәтижесі салмақты сома деп саналады.

Салмақты сома (таза) – тиісті салмақтармен көбейтілген кіріс (43) сигналдарының сомасы:

$$Net=\sum_{i=1}^n x_i w_i \quad (43)$$

Тұтқыш рөлі айқын - ол барлық кіріс сигналдарын (көп болуы мүмкін) бірыңғай нөмірге біріктіреді - тұтастай нейрон алған сигналды сипаттайтын салмақты сома. Басқа ауыр салмақты сома нейронның жалпы қозу дәрежесі ретінде ұсынылуы мүмкін.

Мысал:

Жасанды нейронның соңғы компонентінің рөлін түсіну - активациялау функциясы - мен ұқсастықты беремін.

Бір жасанды нейронды қарастырып көрейік. Оның міндеті - теңізде демалысқа бару туралы шешім қабылдау. Ол үшін біз оның деректеріне түрлі деректерді ұсынамыз. Біздің нейронымызда 4 кіріс бар:

- сапардың құны;
- теңіздегі ауа райы қандай?;
- жұмысқа қатысты қазіргі жағдай;
- жағажайда асхана бар.

Барлық осы параметрлер 0 немесе 1-де сипатталады. Демек, егер теңіздегі ауа райы жақсы болса, онда бұл кіріс 1-ге беріледі.

Егер нейрондық төрт кіріс болса, онда төрт салмақ коэффициенті болуы керек. Біздің мысалда салмақ факторлары нейронның жалпы шешіміне әсер ететін әрбір кірістің маңыздылығының көрсеткіші ретінде ұсынылуы мүмкін. Кірістердің салмағы төмендегідей бөлінеді:

1. 5
2. 4
3. 1
4. 1

Теңіздегі қымбат және ауа райы факторлары өте маңызды рөл атқаратындығын (алғашқы екі кіріс) көруге болады. Олар сонымен қатар нейронды қабылдаған кезде шешуші рөл атқарады.

Нейронның кірісіне келесі сигналдарды берейік:

1. 1
2. 0
3. 0
4. 1

Кірістердің салмағын тиісті кіріс сигналдарымен көбейтеміз:

1. 5
2. 0
3. 0
4. 1

Кіріс сигналдарының мұндай жиынтығы (44) үшін өлшенген сома 6:

$$\text{net} = \sum_{i=1}^4 x_i w_i = 5 + 0 + 0 + 1 = 6 \quad (44)$$

Барлығы салқын, бірақ келесі жолы не істеу керек? Нейрон теңізге қалай бару керектігін қалай шешеді? Әлбетте, біз салмақты соманы түрлендіріп, жауап алуымыз керек.

Бұл жерде сахнада активтендіру функциясы пайда болады.

Қосылу функциясы

Шығу үшін салмақты соманы беру дұрыс емес. Нейрон оны қандай да бір түрде өңдеп, тиісті шығу сигналын қалыптастыруы керек. Ол осы мақсат үшін және белсендіру функциясын қолданады.

Ол нейронды шығару (салмақты нейронды шығару айнымалы мәнмен белгіленетін) санының белгілі бір санына айналдырады.

Жасанды нейрондардың әр түрлі түрлері үшін белсендірудің әртүрлі функцияларын қолданады. Жалпы жағдайда олар ϕ (таза) белгісімен белгіленеді. Кірістірілген өлшенген сигналды көрсету активация функциясы параметр ретінде өлшенген соманы алады дегенді білдіреді.

Белсендіру функциясы ($\phi(\text{net})$) – бұл дәлел ретінде салмақты соманы қабылдайтын функция. Бұл функцияның мәні – нейронның шығу (шығу):

$$\text{Out}=\phi(\text{net}) \quad (45)$$

Бұдан әрі біз ең танымал активациялау функцияларына толық шолу жасаймыз.

Ортақ секіру функциясы

Қосылу функциясының қарапайым түрі. Нейронның шығысы тек 0 немесе 1 болуы мүмкін. Егер өлшенген сома белгілі бір шегінен асатын болса, онда нейронның шығу уақыты 1. Егер төмен болса, онда 0.

Ол қалай пайдалануға болады? Теңізге 5-тен жоғары немесе тең болған жағдайда ғана барамыз делік. Демек, шегіміз 5:

$$b=5$$

Біздің мысалда өлшенген сома 6 болды, демек біздің нейронымыздың шығу сигналы 1 болып табылады. Демек, біз теңізге барамыз.

Дегенмен, теңіздегі ауа райы жаман болмас еді, сондай-ақ сапар өте қымбатқа түссе де, қалыпты жұмыс жағдайлары (кірулер: 0011) болатын болса, онда салмақты сома 2 болады, яғни нейронның өнімділігі 0 болады. біз еш жерде жүрмейміз.

Жалпы алғанда, нейрон өлшенген соманы қарастырады және оның шегінен үлкен болса, онда нейрон шығу сигналы 1 тең болады.

Сигмоидалдық функциясы

Шын мәнінде, сигмит функцияларының тұтас отбасы бар, олардың кейбіреулері жасанды нейрондарда белсендіру функциясы ретінде пайдаланылады.

Барлық осы функциялардың нейрондық желілерде қолданылатын өте пайдалы қасиеттері бар. Бұл сипаттар осы функциялардың графиктерін көргеннен кейін анық болады және мұнда аналитикалық түрде жазылған:

$$\text{out}(\text{net})=1/(1+\exp(-a \cdot \text{net})) \quad (46)$$

Параметр дегеніміз не? Бұл функцияның қаттылық дәрежесін сипаттайтын кейбір сандар. Төменде логистикалық функциялары бар, олар басқа параметрімен a .

Гиперболалық тангенс

Алайда, тағы бір sigmoid - гиперболалық тангенс. Бұл жүйке жасушасының неғұрлым шынайы үлгісі үшін биологтардың белсендіру функциясы ретінде пайдаланылады.

Бұл функция әртүрлі таңбалардың шығуын (мысалы, -1-ден 1-ге дейін) алуға мүмкіндік береді, ол бірнеше желілер үшін пайдалы болуы мүмкін.

Функция мынадай түрде жазылады:

$$\text{out}(\text{net})=\tanh(\text{neta}) \quad (47)$$

Жоғарыда келтірілген формулада параметр сондай-ақ осы функцияның графигінің тұрақтылығын анықтайды.

Біз не үйрендік?

Енді жасанды нейронның ішкі құрылымының толық бейнесі бар. Мен оның жұмысының қысқаша сипаттамасын тағы бір рет қайталаймын.

Нейронның кірістері бар. Оларға сандар түрінде сигналдар беріледі. Әр кірістің өз салмағы бар (сонымен қатар сан). Кіріс сигналдары тиісті салмаққа көбейтіледі. Біз «салмақты» кіріс сигналдар жиынтығын аламыз.

Бұдан басқа, бұл жиын барлық кіріс сигналдарын салмаққа көбейтетін жай ғана қосқышқа кіреді. Алынған сан салмақты сома деп аталады.

Содан кейін салмақты сома белсендіру функциясы арқылы өзгереді және біз нейронды шығаруды аламыз.

Нейронның ең қысқа сипаттамасын - оның математикалық моделін құрайық:

Жасанды нейронды математикалық модель n inputs:

$$\text{out}=\phi(\sum_{i=1}^n x_i w_i) \quad (48)$$

мұндағы, ϕ - белсендіру функциясы;

$\sum_{i=1}^n x_i w_i$ - тиісті салмақ бойынша кіріс сигналдарының n өнімдерінің сомасы ретінде өлшенген сома.

АЖ түрлері

Жасанды нейронның құрылымын анықтадық. Жасанды нейрондық желілер жасанды нейрондар жиынтығынан тұрады. Логикалық сұрақ туындайды - бірақ бұл жасанды нейронды бір-бірімен қалай реттеуге болады?

Әдетте, көптеген нейрондық желілерде кіріс қабаты бар, ол тек бір тапсырманы орындайды - кіріс сигналдарын басқа нейрондарға бөлу. Бұл қабаттың нейрондары ешқандай есептеулер жасамайды.

Сонда айырмашылықтар басталады.

Бір қабатты нейрондық желілер.

Бір қабатты нейрондық желілерде кіріс қабатының сигналдары бірден шығу қабатына беріледі. Ол қажетті есептеулерді жүргізеді, оның нәтижелері нәтижеге дереу жеткізіледі.

Нейрондар бір-бірімен көрсеткілермен байланысады. Көрсеткілердің үстінде тиісті сілтемелердің салмағы (салмақ коэффициенттері) болып табылады.

Бір қабатты нейрондық желі - кіріс қабатының сигналдары бірден сигнал түрлендіретін және дереу жауап беретін шығу қабатына жіберілетін желі.

Көп қабатты нейрондық желілер

Нейрондық кіріс және шығыс қабаттарынан басқа, мұндай желілер жасырын қабат (қабаттар) арқылы сипатталады. Олардың орналасуын түсіну оңай - бұл қабаттар кіріс және шығыс қабаттары арасында.

Жасырын қабаттың аты кездейсоқ емес. Мәселе мынада, жасырын қабаттың нейрондарын оқытудың соңғы әдісі әзірленді. Бұған дейін бір қабатты нейрондық желілер басқарылды.

Көп қабатты нейрондық желілер бір қабатты желілерге қарағанда әлдеқайда үлкен мүмкіндіктерге ие. Нейрондық жасырын қабаттардың жұмысы үлкен зауыт жұмысымен салыстыруға болады. Зауытта өнімді (шығу сигналы) сатылар бойынша жинайды. Әр машинадан кейін кейбір аралық нәтиже алынады. Жасырын қабаттар да кіріс сигналдарын кейбір аралық нәтижелерге түрлендіреді.

Тікелей тарату желілері

Жоғарыда келтірілген мысалдарда нейрондық желілердің суреттерінде өте қызықты мәліметтер бар.

Барлық мысалдарда көрсеткілер солдан оңға қарай жүреді, яғни мұндай желілерде сигнал кіріс қабатынан шығысқа қарай өтеді.

Feedforward нейрондық желі (feedforward networks) жасанды нейрондық желілер болып табылады, онда сигнал кіріс қабатының шығыс сигналына қатаң түрде таралады. Сигнал кері бағытта таралмайды. Мұндай желілер кеңінен пайдаланылады және міндеттерді белгілі бір сыныпты табысты шешеді: болжау, кластерлеу және тану.

Дегенмен, сигналдың қарсы бағытқа өтуіне ешкім тыйым салмайды.

Түсініктемелері бар желілер

Осы типтегі желілерде сигнал қарама-қарсы бағытта болуы мүмкін. Қандай артықшылығы бар?

Тікелей тарату желілерінде желінің шығысы сигналмен және жасанды нейрондар үшін салмақ коэффициенттерімен анықталады.

Ал кері байланыспен байланысқан желілерде нейрондардың шығысы кірулерге қайта оралуы мүмкін. Бұл нейронның шығуын оның салмағы мен кіріс сигналымен ғана емес, сонымен бірге алдыңғы шығыс арқылы да анықтайды (олар қайтадан кірулерге оралған).

Нейрондық желіні оқыту

Енді нейрондық желіні үйрену мәселесіне толық шолу жасайық. Бұл не? Және бұл қалай жүзеге асады?

Желіні оқыту дегеніміз не?

Жасанды нейрондық желі жасанды нейрондар жиынтығы. Енді, мысалы, 100 нейронды алайық және оларды бір-бірімен байланыстырамыз. Кіріс сигналын бергенде, шығу кезінде бір нәрсені мағынасыз етеміз.

Сондықтан, желінің кейбір параметрлерін кіріс сигналы қажетті нәтижеге түрлендіргенше өзгерту керек.

Нейрондық желіде не өзгерте аламыз?

Жасанды нейрондардың жалпы санын өзгерту үшін екі себеп бар. Біріншіден, есеп айырысу элементтерінің санының көбеюі жүйені ауыр және артық емес етеді. Екіншіден, 100-ден 100-ге жуық ақсүйектерді жинаған болсаңыз, олар мәселеге дұрыс жауап бере алмайды.

Тұтқаны өзгерте алмайды, себебі ол қатаң анықталған функцияны орындайды - қосу үшін. Егер оны бір нәрсемен ауыстырсақ немесе тіпті алып тастасақ, ол жасанды нейрон болмайды.

Әрбір нейронды активациялау функциясын өзгерткенде, біз әртүрлі және бақылаусыз нейрондық желіні аламыз. Сонымен қатар, көптеген жағдайларда нейрондық нейрондық желілерде бір типті. Яғни, олардың барлығы бірдей белсендіру функциясына ие.

Сілтемелердің салмағын өзгерту үшін тек бір опция қалады.

Нейрондық желіні оқыту (Оқыту) - желіден өткен кіріс сигналы қалаған нәтижеге айналдырылған салмақ жиынтығын іздеу.

«Нейрондық желіні үйрену» терминіне осындай көзқарас биологиялық нейрондық желілерге де сәйкес келеді. Біздің миымыз бір-бірімен байланыстырылған көптеген нейронеттерден тұрады. Олардың әрқайсысы бірдей типті нейрондардан тұрады (активация функциясы бірдей). Біз синапстардың өзгеруін үйренеміз - кіріс сигналын күшейтетін / әлсіреген элементтер.

Алайда, тағы бір маңызды мәселе бар. Егер сіз желіге бір ғана кіріс сигналы арқылы үйренсеңіз, онда желі жай ғана «дұрыс жауапты есте сақтайды». Оның тарапынан ол өте тез үйренетін сияқты. Және сіз сәл өзгертілген сигнал бергеннен кейін, дұрыс жауапты күтуді күте отырып, желі желіден босатылады.

Мұғаліммен сабақ беру

Бұл тәсілдің мәні - сіз кіріс сигналын беріп, желінің жауабын қараңыз және оны қазірдің өзінде дайын, дұрыс жауаппен салыстыру.

Маңызды мәселе. Дұрыс шешімдерді және шешімнің танымал алгоритмін шатастырмаңыз! Бетіңізді саусағыңызбен суретке түсіруіңізге болады (дұрыс жауап), бірақ оны қалай орындағанын айта алмайсыз (белгілі алгоритм). Міне, сол жағдай.

Содан кейін, арнайы алгоритмдер көмегімен, сіз нейрондық желілердің қосылыстарының салмағын өзгертіп, қайтадан кіріс сигналын бересіз. Жауапты дұрыс жауаппен салыстыру және желіні қолайлы нақтылықпен қабылдамайынша, бұл процесті қайталаңыз (1-тарауда айтылғандай, желі нақты жауап бере алмайды).

Қадағалауға үйретілген оқыту - желідегі жаттығулардың түрі, оның салмағы желідегі жауаптардың дайын әрі дұрыс жауаптардан аз болмайтындай өзгереді.

Дұрыс жауап қайдан алынады?

Егер желіні адамдар танитын болса, біз 1000 фотосуреттің (кіріс сигналдары) оқу үлгісін жасай аламыз және оған бет-жүздерді таңдап аламыз (дұрыс жауаптар).

Егер біз желінің баға өсуін / құлдырауын болжауды қаласақ, онда оқыту үлгісі өткен деректерге негізделуі керек. Кіріс сигналдары ретінде сіз белгілі бір күндерді, нарықтың жалпы күйін және басқа параметрлерді қабылдай аласыз. Ал дұрыс жауаптар - сол күндердегі бағалардың өсуі мен құлдырауы.

Тағыда басқа...

Айта кету керек, мұғалім міндетті түрде адам емес. Мәселе, кейде желі мыңдаған және он мыңдаған әрекеттерді жасай отырып, сағаттар мен күндерді үйрету керек. 99% жағдайда, бұл рөл компьютермен, немесе керісінше, арнайы компьютерлік бағдарламамен орындалады.

Мұғалімсіз оқыту

Оқытушысыз оқыту, бізде кіріс сигналдарына дұрыс жауаптар болмаған кезде қолданылады. Бұл жағдайда бүкіл оқу үлгісі кіріс сигналдарының жиынтығынан тұрады.

Желіні осындай оқытумен не болады? Бұл «үйрену» арқылы желі кіріспен берілген сигналдар сыныптарын бөле бастайды. Қысқаша айтқанда, желі кластерлеуді бастайды.

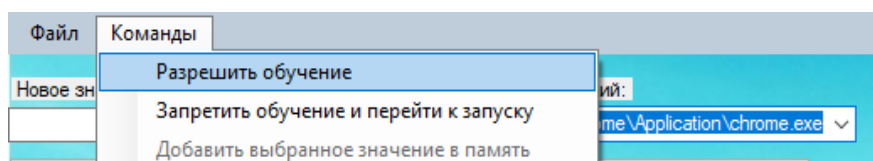
Мысалы, сіз тәттілер, торттар және торттар желісін көрсетесіз. Сіз желіні реттемесеңіз. Сіз нысан туралы деректерді оның енгізулеріне жібересіз. Уақыт өте келе, желі үш түрлі типтегі сигналдарды шығара бастайды, олар кіре берістегі объектілерге жауап береді.

Мұғалімсіз оқыту (бақылаусыз оқыту) желідегі кіріс сигналдарын жіктейтін желілік оқытудың бір түрі болып табылады. Тиісті (анықтамалық) шығыс сигналдары көрсетілмейді [27].

3.7 Нейрондық желіде құрылған бағдарламаның алгоритмі

Бұл программа нейрондық желіде жасалғандықтан, әр енгізуді есте сақтайды және сол арқылы алдыңғы элементтерімен тексереді. Менің программда 10X10 массиві қолданылады:

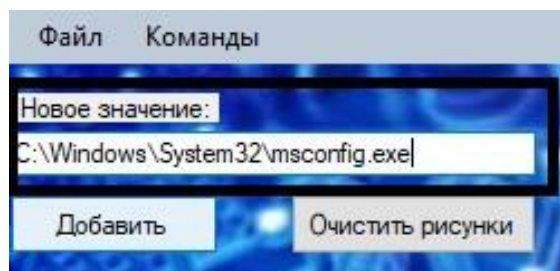
– үйрету режимі қосылады (3.4-сурет):



3.4-сурет – Үйрену режимі

Үйретілу режимінде программа символдарды массивке алады және жадыға жазады. Бұл режимде программалардың процесстері қосылмайды.

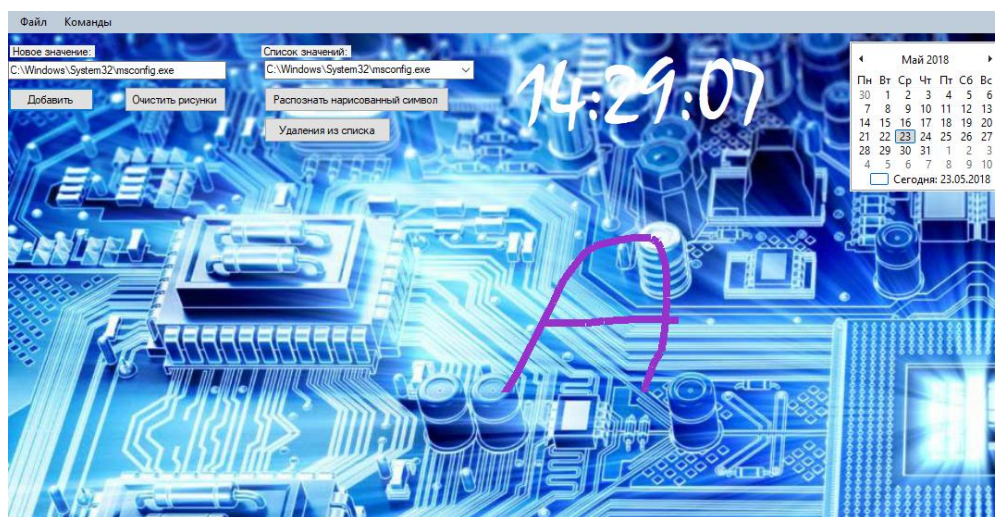
– мән енгізіледі (3.5-сурет):



3.5-сурет – Мәнді енгізу

Мән ретінде абсолютті программалардың жолы белгіленеді. Мәнді қолмен жазудан басқа жүктеу батырмасы қарастырылған. Мән қате жазылса, Microsoft Visual Studio ның қате терезесі шығады.

– сол мәнге байланысты басты формаға символ сызылады (3.6-сурет):



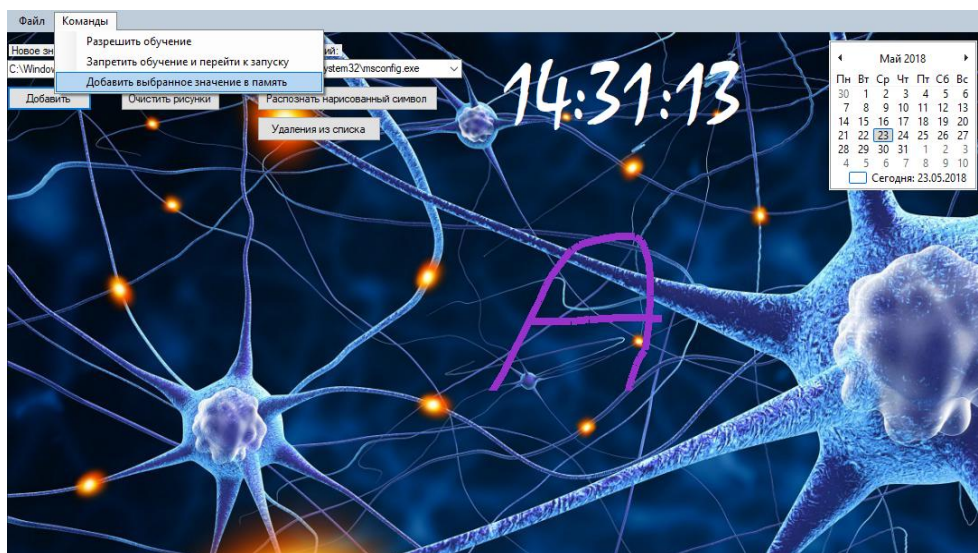
3.6-сурет – Жаңа символ енгізу

Нейрондық желі берілген бейнені 10 да 10 массивіне жазады түсу нүктесі 0,5 тен кем болса ол 0 – ге теңестіреді. Егер 0,5 тен үлкен болса, 1 – ге теңестіреді.

Түсу аймағы экранның бүкіл беті, оны жүзеге асыру үшін PictureBox ты толық бетке қойылды.

Экрандық бетті толық кескіндеу массивтің толуына әкеледі және программа ондай барлық мәнді біреу ретінде қарастырады.

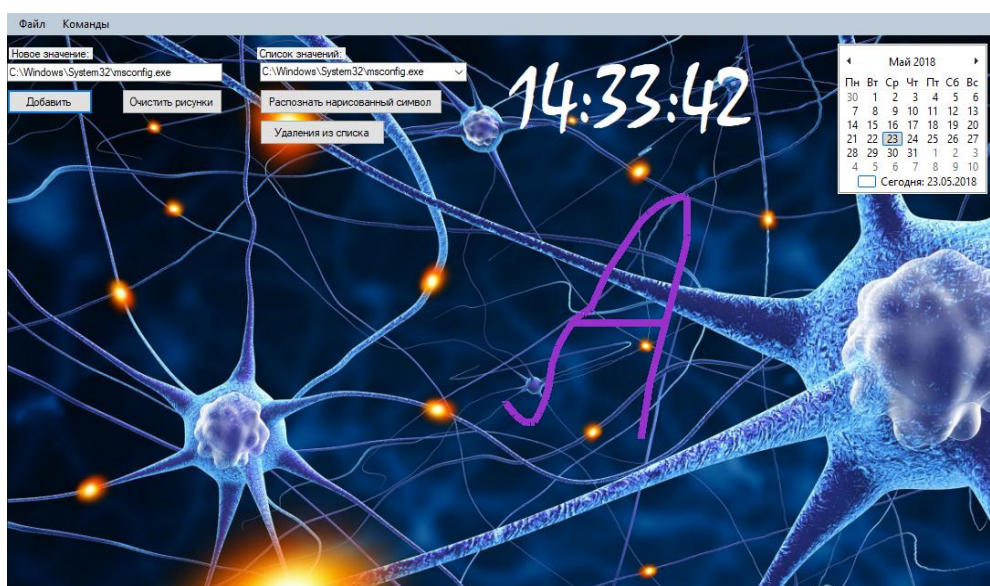
– мән массивке салынады (3.7-сурет):



3.7-сурет – Жаңа мән қабылдау

Массивтік мән салынуы – массивтік класстың орындалуымен түсіндіріледі. Ол үшін бөлек форма құрылған.

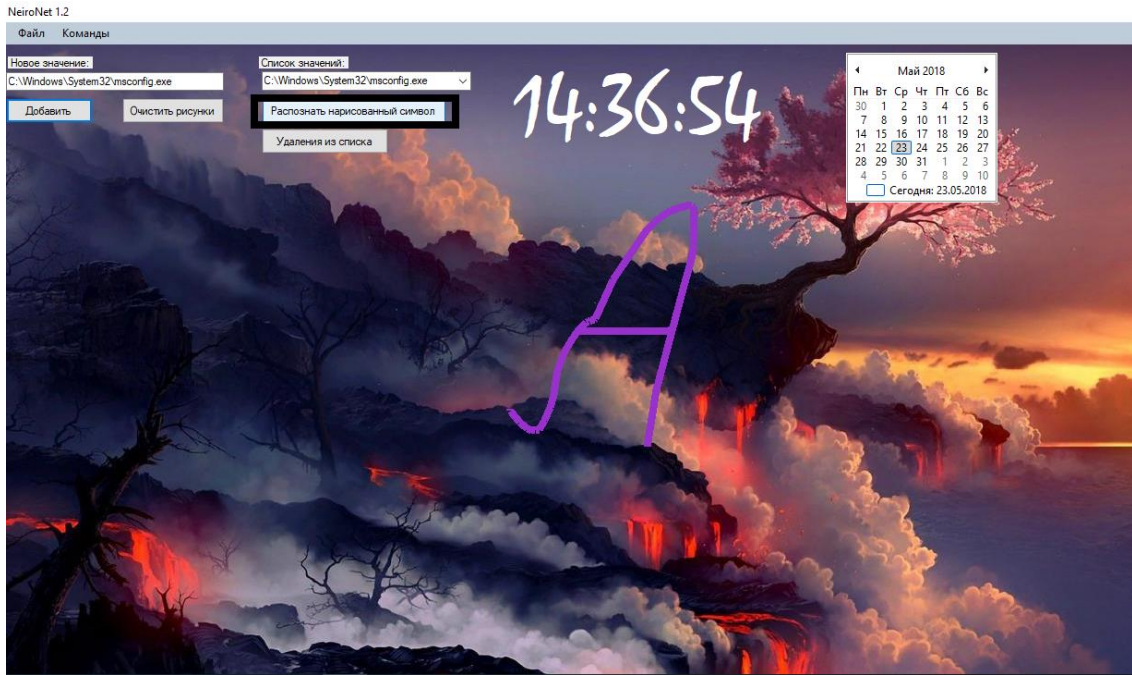
– сол символдың бейнесіндей, тағы жаңа символ енгізіледі (3.8-сурет):



3.8-сурет – Символды қайта енгізу

– алдыңғы символмен салыстырылады (3.9-сурет):

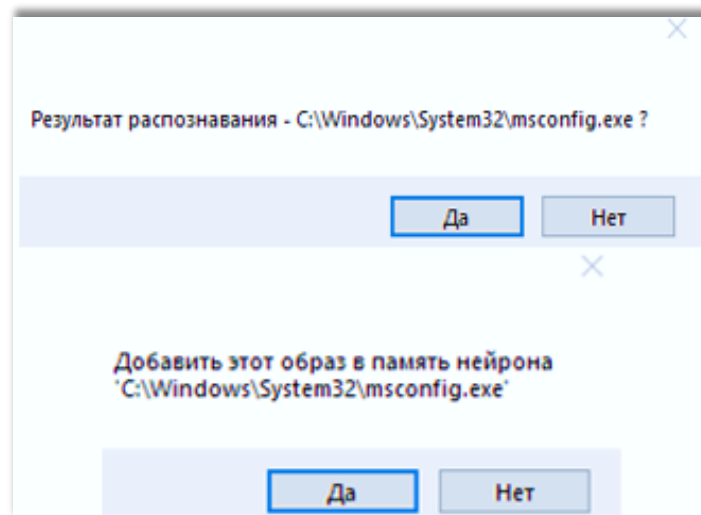
Символды енгізу кезінде бетті толық алмайды. Программа тек нүктелер түсірілген жерлерді қарастырады. Сол арқылы барлық PictureBox-ты емес тек оның бөлігін алады.



3.9-сурет – Салыстыру

Экспоненттік функция бойынша алдыңғы символмен массивта есептеліп тексеріледі 1 – ге жуық мәндер көп болған сайын нәтиже дұрыс мәнді береді.

– егер мән символға сәйкес келсе, мән символды массивке енгізеді (3.10-сурет, 3.11-сурет):



3.10-сурет – Тану кезеңі

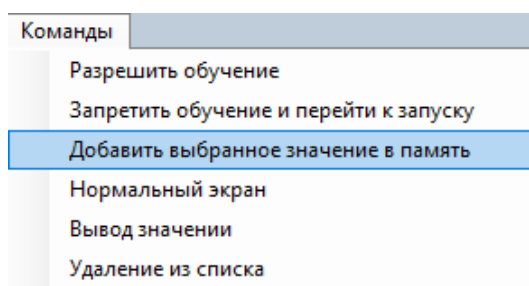
Мән ретінде программалардың соңға жолы алынады және memoгу.txt-да сериязацияланған массивпен мәндерін және түскен нүктелерін салыстырады. Егер түскен нүктелер берілген бейнедегі нүктелердің жартысынан көбін алса, ол сол мәнді шығарады.

Имя образа - C:\Windows\System32\msco...

▶	0	0	0	0	0.7	1	1	0.9...	0	0
	0	0	0	0	1	1	0.6	0.7	0	0
	0	0	0	0.1...	1	0.0...	0.1...	0.9...	0	0
	0	0	0	0.5...	1	0	0.8	0.9...	0	0
	0	0	0.3...	0.5...	1	0.9...	1	0.9...	0.1...	0
	0	0	0.5...	0.7	1	0.6...	1	0.5...	0.2...	0
	0	0	0.3...	0.7	0.6	0.0...	1	0	0	0
	0	0.2...	1	0.3...	0.6	0.1...	0.8	0	0	0
	0	0.2...	0.8	0.4...	0.8	0.6...	0.8...	0	0	0
*	0	0.5...	0.7	0.4...	0.1...	0.6...	0.8...	0	0	0

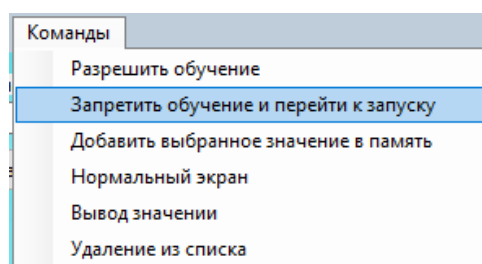
3.11-сурет – Массив 10-да 10

– егер сәйкес келмесе символды қолданушы өзі массивке енгізеді (3.12-сурет):



3.12 – Жаңа мән енгізілу батырмасы

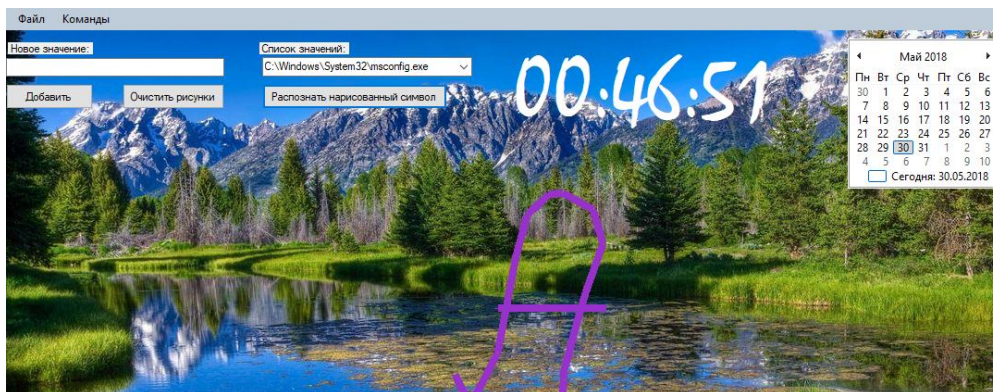
– оқыту режимі тоқтатылады (3.13-сурет):



3.13-сурет – Тану процессінің тоқталуы

Бұл режимде программа берілген бейнелерді тек таниды бірақ массивке енгізбейді және программалардың процесстерін қосады.

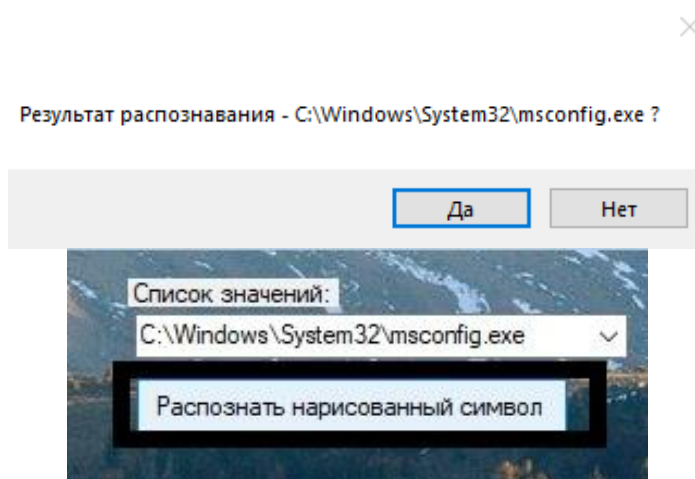
– алдыңғы символдың бейнесі формаға енгізіледі (3.14-сурет):



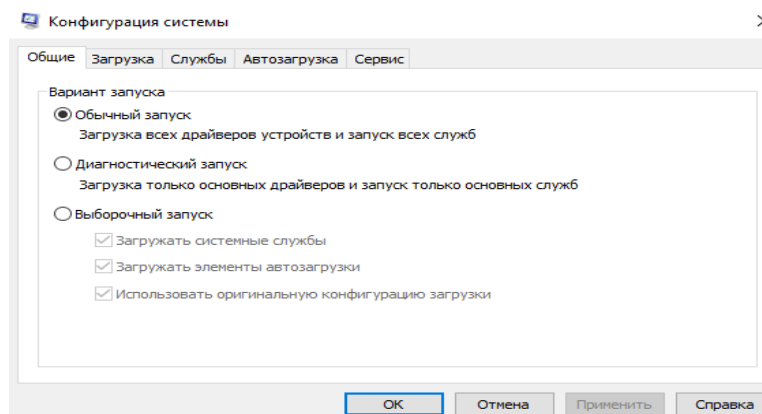
3.14-сурет – Тану процессі

– тану процессі жүреді егер танылса программа сол мәнге байланысты программа ашылады (3.15-сурет және 3.16-сурет);

– кері жағдайда программа орындалмайды. Бұл кезде программаны тоқтатып, мәнді қайта енгізу керек. Сол арқылы программа үйренеді және массив сандары көбейеді. Мәндер көбейген сайын тану дәл жүреді.



3.15-сурет – Тану процессі



3.16-сурет – Программаның процессінің орындалуы

Бағдарламалық қамтамасыз ету симулятор жартықабатты нейрондық желі оңай кеңейтілетін, айқын бар бағдарламаны құрылған және құрылымын кейіннен болды құру объектілі-бағытталған программалаудың принциптерін қолдану.

Қорытындылай келгенде, құрылған бағдарламалық қамтамасыз симулятор жартылай нейрондық желі алдын ала белгіленген оқыту үлгісі ретінде тәуелділікті төмендете отырып, жүйе түрлендіргіштер іске асыру механизмін білдіреді. Ол нейрондық желі емес артық құрылымын жасауға мүмкіндік беретін қателер бағдарламасын нүктелік алгоритмді пайдаланады. Тұрақты салыстыру Липшица бойынша қосымша нүктелік алгоритм сарқылуы қате тексеру селективті Липшица бар нейрондық желі тұрақты пайдаланушы үлгісі көлемі нейрондық желінің функциясын қалпына келтіру үшін қажетті неғұрлым қатаң талаптарына бағдарламалық құралды пайдалануға мүмкіндік береді.

4 Экономикалық бөлім

4.1 Экономикалық қызметтің пайдалылығы

Өнімділік - бұл компанияның қызметінің негізгі экономикалық көрсеткіші. Екі түрі бар:

- негізгі және ротациялық өндіріс үшін кәсіпорын пайдасы
- акциялар құнына қатынасы;
- кәсіпорын өз өндірісінде алатын пайда
- құндылыққа қатынасы;
- кәсіпорынның тиімділігін бағалау үшін алынған
- нәтиже (жалпы табыс) немесе пайдаланылған ресурстар.

Пайда немесе залал сомасы пайда немесе залал болып табылады. Іс жүзінде коммуналдық өлшеу екі жолмен қолданылады. Пайданың ағымдағы шығындарға қатынасы (кәсіпорын құнының шамасы) немесе аванс (негізгі өндірістік активтер және айналым капиталы). Бұл екі өлшемдер ақша ағындарына жұмсалатын шығындардың мөлшеріне байланысты. Іс жүзінде кәсіпорынның қызметінде бірнеше көрсеткіштер пайдаланылады.

Біріншіден, өнімнің табыстылығы барлық сатылған өнімдер мен олардың нақты түрлерін есептеу арқылы ғана есептеледі. Бірінші жағдайда ол өнімнің пайдасын және оның өндірістік және сауда шығындарының көлемін айқындайды. Тауарды сатудан түсетін пайданы сатудан түсетін түсінік үшін сатылатын барлық өнімдердің табыстылығын есептейді; баланстық пайда - өнімді сатудан түсетін түсімдер; Өнімдерді таза пайдаға сатудан түскен табыс. Барлық өнімдерге арналған табыстылық көрсеткіштері кәсіпорынның ағымдағы құнын және өнім сатылымын білдіреді. Екіншіден, өндіріс қорларының кірістілігі - бұл кітаптық пайда мен негізгі құралдардың және айналымның орташа жылдық құнының қатынасы. Бұл көрсеткіш таза пайданың көмегімен есептелуі мүмкін. Үшіншісі - кәсіпорынның кірістілігі. Ол бұл мүліктің құны бойынша есептеледі. Баланс пен тазалау параметрлерін есептеу қолданылады. Компанияның меншікті капиталының рентабельділігі ақша қаражаттарының қозғалысын тудыратын таза активтерінің үлесімен анықталады.

Кірістілік деңгейін көтеретін қарыз бойынша пайыздық мөлшерлеме:

- басқа кәсіпорындардың табыстары;
- бағалы қағаздардан түскен кірістер;
- шамадан тыс айыппұлдар, тұрақсыз төлемдер.

Бұл шығындардың белгілі бір санын - экономикалық басқарудың әртүрлі мәселелерін жоюға қатысты негізгі талап. Өндірістің тиімділігі негізінен негізгі өндірістік активтерді пайдалануды жақсарту есебінен көбейтіледі, екіншіден, роторлық өндірістік активтерді тиімді пайдалану арқылы, сондай-ақ қолданыстағы еңбек шығындарының тиімділігін арттыру. Үлгі әзірленгеннен кейін экономикалық тиімділікті есептеу әдістемесі көрсеткіштердің тиімділігін

анықтайтын негізгі көздердің сипаттамаларын салыстыруды қолданады. Осы әдістің тиімділігі әсерге бағдарланған модель арқылы үздіксіз орындалатын интерактивті процестердің бағытын салыстыруға мүмкіндік береді. Бағдарламалық жасақтаманы әзірлеуден бұрын оның экономикалық тиімділігін есептеу керек, яғни бағдарламалық жасақтаманы құру және енгізу және ақшаны бағдарламалық қамтамасыз етуді енгізу және оны енгізу алдында құнын есептеп шығару қажет.

4.2 Бағдарламалық қамтамасыздандыруды әзірлеуге қатысты ұсыныстардың техникалық-экономикалық негіздемесі

Техникалық-экономикалық негіздемесі мыналарды қамтуы керек:

- ПП дамуының күрделілігін анықтау;
- ПП-ны дамыту шығындарын есептеу;
- әзірленген ПП-ның мүмкін болатын бағасын анықтау;
- ПП жұмысының әлеуметтік-экономикалық нәтижелерін бағалау.

Деректер:

- q - 1600 операторлардың саны;
- c - 1,19 бағдарламасының жаңалығы мен күрделілігін ескере отырып, коэффициент;
- K - біліктілік коэффициенті 1;
- $З_{\text{пoi}}$ - 42 000 теңге;
- $Ц_{\text{р1}}$ - операциялық жүйе жесірлер 30 000 теңге;
- $Ц_{\text{р2}}$ - бағдарламалық қамтамасыз ету ортасы Microsoft Visual Studio 2015 15000 теңге.

4.3 ПП дамуының күрделілігін анықтау

Белгілі бір тапсырма үшін бағдарламалық өнімді әзірлеудің күрделілігі (tr) даму сатыларына арналған еңбек шығындарының сомасы ретінде қарастырылуы мүмкін:

- тапсырманың сипаттамасын дайындау - адамға / сағатқа;
- мәселені шешу алгоритмін әзірлеу - адам, сағат / адам;
- алгоритмнің схемасын жасау - tb , адам / сағат;
- бағдарламалау - tr , адам / сағат;
- компьютерде бағдарламаны, адам / сағатты отладтау;
- тапсырмаға құжаттарды дайындау - td , адам / сағат.

Еңбек қарқындылығын анықтаудың негізгі көрсеткіші - бағдарламалық жасақтамадағы (бағдарламалық қамтамасыз ету) операторлардың командаларының шартты саны. Бағдарламадағы операторлардың шартты саны мына формула бойынша анықталады:

$$Q = q * c = 1600 * 1.19 = 1904 \quad (49)$$

мұндағы, Q - операторлардың шартты саны;
q - мәселенің түріне байланысты операторлардың күтілетін саны;
с - бағдарламаның жаңалығы мен күрделілігін ескеретін коэффициент.

Q коэффициентінің мәні 1-кестеден таңдалуы мүмкін.

1-кесте – q коэффициентінің мәндері

Тапсырма түрі	Коэффициент өзгерістерінің шегі
Бухгалтерлік есеп	1400-1500
Операциялық басқару тапсырмалары	1500-1700
Тапсырмаларды жоспарлау	3000-3500
Көп функциялы тапсырмалар	4500-5000
Кешенді тапсырмалар	5000-5500

Бағдарламалық өнімдер жаңалық дәрежесі бойынша 4 топтың біріне жатқызылуы мүмкін:

- А тобы - түбегейлі жаңа міндеттерді әзірлеу;
- В тобы - бастапқы бағдарламалар әзірлеу;
- В тобы - стандартты шешімдерді қолданатын бағдарламалар әзірлеу;
- D тобы - бір реттік типтік мәселе.

К коэффициенті күрделі топтардың қиылысында және жаңалық дәрежесіндегі 2-кестеден анықталады.

Содан кейін біз бағдарламалық жасақтама өнімін жасаудың әрбір кезеңіне жұмсалатын уақытты анықтаймыз:

– тапсырма сипаттамасын дайындау уақыты факті бойынша қабылданады және (3-5 күннен 8 сағатқа дейін):

$$t = 24 \text{ адам / сағат.}$$

2-кесте – Еңбек қарқындылығын есептеу коэффициенттері

Программалау тілі	Қиындық тобы	Жаңалық дәрежесі				Коэффициент B
		А	Б	В	Г	
Жоғарғы деңгей	1	1,38	1,26	1,15	0,69	1,2
	2	1,30	1,19	1,08	0,65	1,35
	3	1,20	1,10	1,00	0,60	1,5
Төменгі деңгей	1	1,58	1,45	1,32	0,79	1,2
	2	1,49	1,37	1,24	0,74	1,35
	3	1,38	1,26	1,15	0,69	1,5

t_a (мәселені шешу алгоритмін әзірлеу уақыты) формула бойынша есептеледі:

$$t_a = Q / (50 * K) = 1904 / (50 * 1) = 38.08 \text{ адам / сағат} \quad (50)$$

онда K - бағдарламашы біліктілігін ескеретін коэффициент. K коэффициентінің мәні 3-кестеден таңдалуы мүмкін.

3-кесте – бағдарламашы біліктілігінің коэффициенттері

Тәжірибелік жұмысы	Квалификация коэффициенті
2 – жылға дейін	0.8
2-3жыл	1
3-5жыл	1,1–1,2
5-7жыл	1,3–1,4
7жылдан көп	1,5–1,6

– t_b (блоктық схеманы жасау уақыты) 50-формула бойынша анықталады.

$$t_b = 38.08 \text{ адам / сағат}$$

– t_n (бағдарламаны бағдарламалау тілінде жазу уақыты) формула бойынша анықталады:

$$t_n = Q * 1,5 / (50 * K) = 38,08 * 1,5 = 57,12 \text{ адам / сағат} \quad (51)$$

– бағдарлама (бағдарламаны отладтау және тестілеу уақыты) мынадай формула бойынша анықталады:

$$t_{отл} = Q * 4,2 / 50 * K = 1904 * 4,2 / 50 * 1 = 159,936 \text{ адам / сағат} \quad (52)$$

– құжаттың дайындалу уақыты анықталғаннан кейін қабылданады және (3 сағаттан 5 күнге 8 сағатқа дейін):

$$t_d = 24 \text{ адам / сағат}$$

Жалпы еңбекке жұмсалған шығындар формула бойынша адам / сағат:

$$t = t_0 + t_a + t_b + t_n + t_{отл} + t_d = 24 + 38,08 + 38,08 + 57,12 + 159,936 + 24 \approx 342 \text{ адам/сағат} \quad (53)$$

4.4 Бағдарламалық жасақтама әзірлеу шығындарын есептеу

Бағдарламалық өнімді (бағдарламалық жасақтама) әзірлеуді қажет ететін жұмыстарда шығындар

(C_n) мынадай формула бойынша анықталады:

$$C_{\Pi} = Z_{\text{фот}} + Z_{\text{сз}} + M_i + P_{\text{сi}} + P_{\text{мi}} + P_{\text{э}} + P_{\text{нi}}, = 133\ 635 \text{ тг} \quad (54)$$

мұндағы, $Z_{\text{фот}}$ әзірлеушілердің жалақысы, теңге;
 $Z_{\text{сз}}$ – әлеуметтік салық бойынша аударымдар, теңге;
 M – материалдарға арналған шығындар, теңге;
 $P_{\text{сi}}$ – дизайнерлік шешімді әзірлеу үшін қажетті арнайы бағдарламалық қамтамасыз етуге арналған шығындар, теңге;
 $P_{\text{мi}}$ – жабдықтың құнсыздануына жұмсалатын шығындар, теңге;
 $P_{\text{э}}$ – электр энергиясының құны, теңге;
 $P_{\text{нi}}$ – үстеме, теңге.

Әзірлеушілердің жалақы мөлшері ($Z_{\text{фот}}$) формула бойынша есептеледі:

$$Z_{\text{ФОТ}} = Z_{\text{oi}} + Z_{\text{ди}} = 74\ 813 + 7\ 481,3 \approx 82\ 294 \text{ тг} \quad (55)$$

мұндағы, Z_{oi} – негізгі жалақы, теңге;
 $Z_{\text{ди}}$ – қосымша еңбекақы, теңге.

Белгілі бір бағдарламалық қамтамасыз ету үшін орындаушылардың негізгі жалақысы мынадай формула бойынша есептеледі:

$$Z_{\text{oi}} = t_p * Z_{\text{poi}}, = 342 * 218,75 \approx 74\ 813 \text{ тг} \quad (56)$$

мұндағы, t_p – бағдарламалық өнімді, адам-күндерді дамыту күрделілігі;
 Z_{poi} – девелопердің күнделікті жалақысы, i -орындаушы, (теңге).

Күнделікті жалақы әзірлеуші айлық жалақысы мен айдағы жұмыс күндерінің (орта есеппен 22 жұмыс күнін қабылдайтын), 24 тәуліктен кейін анықталады. Дамуға қатысатын қызметкерлер туралы ақпарат 4-кесте түрінде ұсынылуы тиіс.

4-кесте – Жобаға тартылған қызметкерлер туралы ақпарат

Орындаушы	Адамдар саны	Жалақы айына, теңге
Кожобеков М.Т	1	42000 тг
Барлығы	1	42000 тг

Негізгі жалақыны есептеу нәтижелері 5 кесте түрінде ұсынылуы керек.
 Қосымша жалақы базалық жалақының 10% құрайды және келесі формула бойынша есептеледі:

$$Z_{\text{д}} = Z_{\text{о}} * N_{\text{д}} / 100 = 74\ 813 * 10 / 100 = 7\ 481,3 \text{ тг} \quad (57)$$

5-кесте – Негізгі жалақы шығындарын есептеудің жиынтық нәтижесі.

Жұмыстың аты	Орындаушы	Еңбек көлемі норма-сағат	Бір сағатта алатын жалақы тенге	Жалақының қосынды мәні тенге
1	2	3	4	5
Нейрондық желілер арқылы символдарды тану	Кожабеков М.Т	342	42000/24/8≈ 218.75 тг	82 294 тг

мұндағы, Нд – әзірлеушілердің қосымша жалақы коэффициенті.

Әлеуметтік салық қызметкерлердің табысының 9,5% құрайды және келесі формула бойынша есептеледі:

$$З_{сз}=(\text{ФОТ}-\text{ПО})\cdot 9,5\%=(82\,294-8229,4)\cdot 0,095\approx 7\,036\text{тг} \quad (58)$$

мұндағы, ПО – зейнетақы жарналары, 10% құрайды және әлеуметтік салық төленбейді:

$$\text{ПО}=\text{ФОТ}\cdot 10\%=8\,229,4\text{ тг.} \quad (59)$$

Кіріс деректеріне негізделген материалдардың құны мынадай формула бойынша анықталады:

$$M=(З_{\text{осн.}}\cdot N_{\text{мз}})/100\%=(74\,813\cdot 4)/100\approx 18\,703\text{тг} \quad (60)$$

мұндағы, $N_{\text{мз}}$ - негізгі жалақыдан материалдарды жұмсау нормасы (3-5%).

Жобалық шешімді әзірлеуге қажетті арнайы бағдарламалық өнімдерге арналған шығындар:

$$P_c=C_p=C_{p1}+C_{p2}=30000+15000=45000\text{ тг} \quad (61)$$

$$P_{m1}=45000\cdot 0,25\cdot 342/(1200\cdot 24)\approx 134\text{ тг}$$

мұндағы, K_p – арнайы бағдарламалық қамтамасыз етудің бағасы, тенге.

Амортизацияға арналған шығындарға бағдарламалық өнімдерді әзірлеу үшін пайдаланылатын жабдықтың құнын амортизациялық аударымдар кіреді және мынадай формула бойынша есептеледі:

$$P_{m2}=\frac{C_{\text{обор}}\cdot N_a\cdot N}{100\cdot 12\cdot t}=7\,060\,300\cdot 0,25\cdot 342/(1200\cdot 24)\approx 20\,960\text{ тг} \quad (62)$$

$$P_M = P_{M1} + P_{M2} = 134 + 20\,960 = 21\,094 \text{ тг}$$

мұндағы, X_a - амортизация нормасы (25%);

Собор - жабдықтың бастапқы құны;

N - дербес компьютерді пайдалану уақыты;

t - айдың ішінде жұмыс күндерінің саны.

Электр энергиясының құны мына формула бойынша есептеледі:

$$P_3 = M * k_3 * T * C_{kW-h} \quad (63)$$

$$P_3 = 1,1 * 0,8 * 342 * 27,48 \approx 8\,270 \text{ тг}$$

мұндағы, M - компьютердің қуаты, кВт;

k_3 - жүктеме коэффициенті (0,8);

C_{kW-h} - 1 кВт / сағат электр энергиясының құны, теңге / кВтсағ;

T - жұмыс уақыты, сағат.

Электр қуатының шығындарын есептеу нәтижелері 6-кестеде көрсетілген нысан бойынша ұсынылуы тиіс.

«Үстеме шығыстары» (R_n) тармағындағы шығындар еңбек шығындарының 40-60% мөлшерінде (экономикалық кеңес берушімен кеңесу арқылы) қабылданады:

$$R_n = 30i * R_{nH} / 100\% = 74\,813 * 45 / 100 \approx 33\,666 \text{ тг} \quad (64)$$

6-кесте – Энергия шығыны

Құрылғының атауы	Қуаттылығы, кВт	Жүктеу коэффициенті	ІПП құруға кеткен уақыт, час	Бағасы, теңге/кВт-час;	Қосындысы, теңге
Сервер Dell PowerEdge R820 R820	1,1	0,8	342	27,48	8 270
Электр энергиясына кеткен ақша					8270 тг

мұндағы, R_n – нақты бағдарламалық жасақтамаға арналған шығындар (теңге);

R_{nH} – үстеме шығындардың нормасы (%).

Жекелеген элементтер үшін алынған деректер негізінде, 7-кестеде көрсетілген бағдарламалық жасақтама өнімін әзірлеуге жұмсалатын шығын сметасы жасалады.

7-кесте – БҚ-ға кеткен шығындар сметасы

Шығындар аттары	Соммасы,тенге
Еңбек төлеу фонды	82 294
Социалдық салық	7 036
Материалдар	18 703
Амортизация	21 094
Электроэнергия	8 270
Үстеу шығыны	33 666
	171 063
Барлығы:	171 063

4.5 ІІІ-ның ықтимал (шарттық) бағасын анықтау

ІІІ-ның ықтимал (келісімшарттық) бағасының шамасы тұтынушының (тұтынушының) және орындаушының экономикалық мүдделерін қанағаттандыратын деңгейде оның орындалу тиімділігі, сапасы мен мерзімдері ескеріле отырып белгіленуі тиіс.

Келісілген бағасы (CD) келесі формула бойынша есептеледі:

$$Цд = [Cn*(1+P/100)]+НДС=171\ 063*(1+25/100)+ 25\ 659,45 \approx 239\ 488\ \text{тг} \quad (65)$$

мұндағы, Cn - ҚБ-ны дамытуға арналған шығындар, теңге;

P - орташа табыстылық деңгейі (25%);

ҚҚС - қосылған құн салығы,%.

Қазақстан Республикасындағы қосылған құн салығының ставкасы 2018 жылға дейін ҒЗТҚЖ сату бағасының 12% құрайды және келесі формула бойынша есептеледі:

$$НДС = [Cn*(1+P/100)]*0.12=171\ 063*(1+0,25)*0,12=25\ 659,45\ \text{тг} \quad (66)$$

4.6 ІІІ-ның әлеуметтік-экономикалық көрсеткіштерін бағалау

ҚБ жұмысының әлеуметтік-экономикалық нәтижелері төрт негізгі критерийге сәйкес бағалануы керек:

- сапа менеджменті процестер: процесінің автоматты басқару, хост компьютер жадында сақтайды алапта пайдаланушы енгізген деректерді;
- нәтижелердің процесінің мен мерзімдері ұзақтығы: процесі әр ай сайын алып, нәтижелері күн сайын алынады;
- бқ пайдалану бойынша жұмсалған шығындар: 171 063 тг;
- автоматтандырылған үдерісте жұмыс істейтін мамандардың саны: 1.

PP аналогтары: CityWebMaster – 3 491400 тг, Энни – 2 399236 тг. ПП-ның құны 239 488 теңге. PP әлдеқайда арзан оның әріптестерінің және ол аз еске өңдеу үшін пайдаланылады.

Менің PP өмірлік циклі - 5 жыл. 5 жылдан кейін пайдалануды одан әрі дамыту үшін жүйені жаңарту керек.

Қажет болған жағдайда, PP өзгерту және тоқтаусыз базасында деректерді өзгерту, ПП үшін сұрау болып табылады және ол өзі өткен жазбаларды жою немесе түзету нысанын береді.

PP нейрондық желі болып табылады, ол жадта суреттерді сақтайды және қоңырау келесі әрбір уақыт алдыңғы суреттер, ол оқытылған және тұрақты жетілдіріледі тәсіліне сәйкес оларды салыстырады.

5 Өміртіршілік қауіпсіздігі

5.1 Құрылыстың желдету жүйесі

Желдету - бұл құрылыстың ережелеріне сәйкес бөлмелерде және жұмыста ауаны қамтамасыз ететін шаралар мен құрылғылардың жиынтығы. Мұнда біз туралы айтатын таза ауа. Осы мақсатта бөлмелерде желдету жүйесі орнатылған. Желдету түрлері: үй-жайлардың табиғи желдетуі - ауа алмасу физикалық «Сасотоком» есебінен болады. Механикалық желдету ауа желдету жабдығы мен құрылғылармен басқарылады, бұл ауаның ұзақ қашықтыққа тасымалдануына мүмкіндік береді (желдеткіштер, жылытқыштар). Бөлмеден шыққан желдету жүйесі ластанған немесе сіңірілетін ауамен шығарылады. Келе жатқан ауаның желдету жүйесі ауаны кондиционерлеу арқылы қашықтан желдету үшін қажет. Жергілікті желдету ауаның кейбір жерлеріне, желдету (жергілікті желдету) және зиянды заттардан (жергілікті желдеткіш) ауаның ластануы кіреді. ЖҚС-да кішігірім желдету жүйесі орнатылған (мысалы, кішкене бөлмелер). Көп жағдайда бөлменің ауасын шығару үшін желдету жүйесі қолданылады.

Серверде желдету мен ауаны баптау не үшін қажет?

Серверлерде желдету және ауа баптау жүйелерінің негізгі мақсаты компьютерлік және телекоммуникациялық жабдықтың дұрыс жұмыс істеуін қамтамасыз ету үшін оңтайлы микроклиматты жасау болып табылады.

Осы типтегі үй-жайларға арналған желдету және ауа баптау жүйесі мынадай міндеттерді орындайды:

- микроклимат параметрлерін тез және дәл бақылау;
- белгілі бір температура режимін, ылғалдылық деңгейін және шаңдылықты қолдау;
- осы микроклимат параметрлерін тұтастай алғанда тәулігіне 24 сағат бойы тұрақтылық пен үздіксіз қамтамасыз ету;
- салқындату үшін қажетті резервті қалыптастыру;
- өрт кезінде бақылауды және блоктау жүйесін дәл бақылау.

Серверлік бөлмеде жоғары сапалы желдетуді және ауаны баптауды ұйымдастыру тәжірибе, білім және ҚНЖЕ ережелеріне қатаң сәйкестікті талап ететін өте күрделі инженерлік міндет болып табылады.

5.2 Серверлік бөлмелер үшін климаттық технологияның негізгі түрлері

Серверге арналған климаттық технологияның кең таралған және танымал түрлері - бұл қабырға мен төбеге бөлінген сплит жүйелері. Олардың кез-келгенінде өз артықшылықтары, кемшіліктері мен ерекшеліктері бар.

Қабырғаға бөлінетін жүйелер бюджеттік, тиімді және пайдалы кеңістік емес болып бөлінеді. Бұл модельдердің кемшілігі, олар салқындату қабілетінің төмен деңгейін қамтамасыз етеді және фреон трассасының шектеулі

ұзақтығына ие. Ең қуатты модель 10 кВт қуаттылыққа ие, бұл үлкен ауданы бар жоғары сапалы кондиционерлерді қамтамасыз ету үшін жеткіліксіз.

Төбеасты сплит жүйесі төбенің бойында ауа ағынының бағытталуы салдарынан, бөлме бойынша ауа ағынын біркелкі бөлу мүмкіндігін береді. Осындай кондиционерлердің үлкен артықшығы - жабық және ашық блоктардың бір-бірінен елу метр қашықтықта орналасу мүмкіндігі.

Сервер бөлмесінде желдету және ауаны баптауға қойылатын талаптар

Серверлік бөлмелердің желдету және ауаны баптау бойынша жобаларды құру СНиП 2-04*9141-01*2003 нормативтік құжаттар негізінде, сонымен қатар СН 512-78-мен жүзеге асырылады.

Нормативтік-құқықтық актілерге сәйкес жабдықтар үшін қолайлы жылу режимін қамтамасыз ету үшін келесі реттеуші мәндер сақталады:

– температураның режимі: температураның өзгеру жылдамдығы бір сағат ішінде 2°C аспауы керек екенін ескере отырып, $+17^{\circ}\text{C}$ -тан $+24^{\circ}\text{C}$ дейін;

– сервер бөлмесіндегі ылғалдылық индексі: ауа ылғалдылығының өзгеру жылдамдығы бір сағат ішінде 6% аспауы керек екендігін ескере отырып, 40% - 55%;

– бөлменің шаңның деңгейі: 0,001 г / м³;

– атмосфералық қысым деңгейі: 85-105 кПа.

Орташа температура мен ылғалдылық деңгейін өлшеу барлық жабдықтардың тұрақты жұмысы кезінде еденнен 1,5 метрлік белгісімен, жолдар арасындағы кез келген өткелдің ортасында жүзеге асырылады.

Сондай-ақ, желдету талаптары ауа алмасу 1,5-2 қр / с болатын жалпы биржалық жабдықтауды және газдарды шығару жүйесін ұйымдастыруды қарастырады. Сервер бөлмесіне таза ауаны жеткізбес бұрын оны тиісті температураға дейін қыздырыңыз немесе салқындатыңыз.

Барлық шығатын саңылаулар ең үлкен тепловозды жабдықпен жабдықталған.

Серверді салқындату үшін арзан және төмен қуатты кондиционердерді қолдану ұсынылмайды, өйткені олар тазалық пен ылғалдылық параметрлерін дұрыс реттей алмайды. Электрондық жабдықты талап етілмеген параметрлерге сәйкес келмеуіне байланысты істен шыққан жағдайда сізге көп ақша төлеуге тура келеді.

Серверлік бөлмеге арналған кондиционерлерге қойылатын сенімділік талаптары

Сервердің ауа баптау жүйесі ең төменгі резервпен 50% құрастырылған. Бұл жүйелік түйіндердің біреуі істен шыққан жағдайда оңтайлы микроклимат параметрлерін бұзуды болдырмау үшін қажет.

Электрондық және телекоммуникациялық жабдықтармен нысандар тәуелсіз салқындату контуры, тәуелсіз және дербес электрмен жабдықтау жүйесі автоматтандыру, реттеу және бақылау, олардың әрқайсысы кем дегенде екі автономды кондиционерлеу жүйелерін және желдету жүйелерімен жабдықталған болуы тиіс.

Бұдан басқа, бұзылу жағдайында жабдықты дереу автоматты түрде ауыстыруды қамтамасыз ететін сенімді құрылғы болуы керек.

Кондиционерлерді серверлік бөлмелердің сенімділігін арттыру үшін, олардың қоректендіру және рециркуляциялық ауа желілерінде жұптауды қамтамасыз ету қажет. Сонымен қатар компрессор, сорғы, желдеткіш агрегаты сияқты негізгі жүйенің элементтерінің қайталануын қамтамасыз ету қажет.

Кез-келген резервтік жүйелер төтенше жағдайда 48 сағат ішінде барлық жабдықтардың жоғары сапалы және ақаулықсыз жұмысын қамтамасыз етеді.

Серверлік бөлмедегі ылғалдандыру жүйесіне қойылатын талаптар

Жоғары сапалы кондиционерлерді ұйымдастырудың тағы бір маңызды аспектісі ылғалдылықтың оңтайлы деңгейін ұстап тұру болып табылады. Құрғақ ауа массасы электростатикалық әсерді тудырады, бұл жағымсыз ғана емес, сонымен қатар сезімтал электр схемаларына және құрылғыларына зиян келтіреді. Бұған байланысты үзілістер мен шығындардың алдын алу үшін тиімді ылғалдандыру жүйесін қамтамасыз ету қажет.

Төбемен және қабырғадағы кондиционерлермен болатын тізбектер үшін оңтайлы ылғалдылықты сақтау үшін олар бөлек стационарлық жүйені орнатады және оның шашыратқышы кондиционерге жақын орналасады.

512-78 SN деректері бойынша, компьютерлік және телекоммуникациялық жабдықтары бар бөлмелерде мынадай ауа алмасу схемалары мүмкін:

– «жинақтау» схемасы:

Бөлменің 1 шаршы метріне 350 ккал / сағаттан аспайтын жылу жүктемесіне жарайды.

– «төменнен жоғары» схемасы (ол біріктірілген):

Бұл бөлме ауданының 1 шаршы метріне 350 ккал / сағ артық жылу жүктемелері үшін қолайлы.

5.3 Серверлік бөлменің желдету жүйесін жобалау және есептеу

Серверлік бөлменің желдету жүйесінің жобалануы мен есептелуі үшін келесі мәліметтер қажет: бөлменің атауы мен көлемі, жұмыс орындарының саны және олардың мақсаты, қызметкерлердің саны, қуат деңгейі мен жұмыстың категориялары, құрылғылар, машиналардың тізімі мен орналасуы, жұмыс уақыты,

Желдету жүйесі ағын, ауа өткізбейтін, ауаның ағымы қағидаты бойынша жұмыс істей алады.

Ағындық желдету жүйесі өндірістік орындарда зиянды заттардың аз мөлшерімен үйлесімді және қалдықтарды бөлуді шектейді, бірінші кезекте қолдауды нығайту үшін ауа жастығы бар. Көп жағдайларда гидравликалық желдету жүйесі жалпы желдету жүйелерінде ауа ағыны мен ауаны желдетуді өтеу үшін пайдаланылады.

Желдету желдеткіші ауадағы зиянды заттардың концентрациясы төмен болғанда және ауа ағыны аз болған кезде ластанған ауаны тиімді бөлмеден

тазалау үшін қолданылады. Ауа айырбас бағамы келесі формула бойынша анықталады:

$$k = \frac{L}{V} \quad (67)$$

бұл жердегі L – бөлмеге берілетін немесе шығарылатын ауаның көлемі, $\text{м}^3/\text{ч}$; V – бөлменің көлемі, м^3 .

Алдымен бөлменің көлемін тауып аламыз:

$$V = a * b * h \quad (68)$$

мұндағы, $a=12\text{м}$, $b=6\text{м}$, $h=3\text{м}$.

$$V = 12 * 6 * 3 = 216 \text{ м}^3$$

Бөлмеге берілетін немесе шығарылатын ауа көлемін табу үшін келесі нәрселерді білу қажет: егер бөлме ішінде зерттеу жұмыстарын жүргізетін әрбір жұмысшыға бөлінген аудан 20 м^3 кем болса, онда шығарылатын ауа көлемі әр жұмысшыға кем дегенде $30 \text{ м}^3/\text{ч}$ болу керек; егер 20 м^3 көп болса онда ол кем дегенде $20 \text{ м}^3/\text{ч}$ болады. Біздің бөлме көбінесе компьютерлермен жұмыс жасау үшін арналғандықтан, ол жерде көп дегенде орта есеппен 6 адам жүруге мүмкіндігі бар. Сәйкесінше бір жұмысшыға бөлінген аудан 36 м^3 -қа тең

$$V_1 = V/6 = 216 / 6 = 36 \text{ м}^3 \quad (69)$$

Бұдан шығатын нәтиже, бізде бір адам үшін шығарылатын міндетті ауа көлемі $L_1 = 20 \text{ м}^3/\text{ч}$ болады.

Жалпы шығарылатын ауа көлемі мынаған тең:

$$L = 6 * L_1 = 6 * 20 = 120 \text{ м}^3/\text{адам} \quad (70)$$

Қажетті мәндерді тауып алғаннан кейін мәндерін орнына қойып, ауа еселілігін анықтаймыз (5.1-сурет):

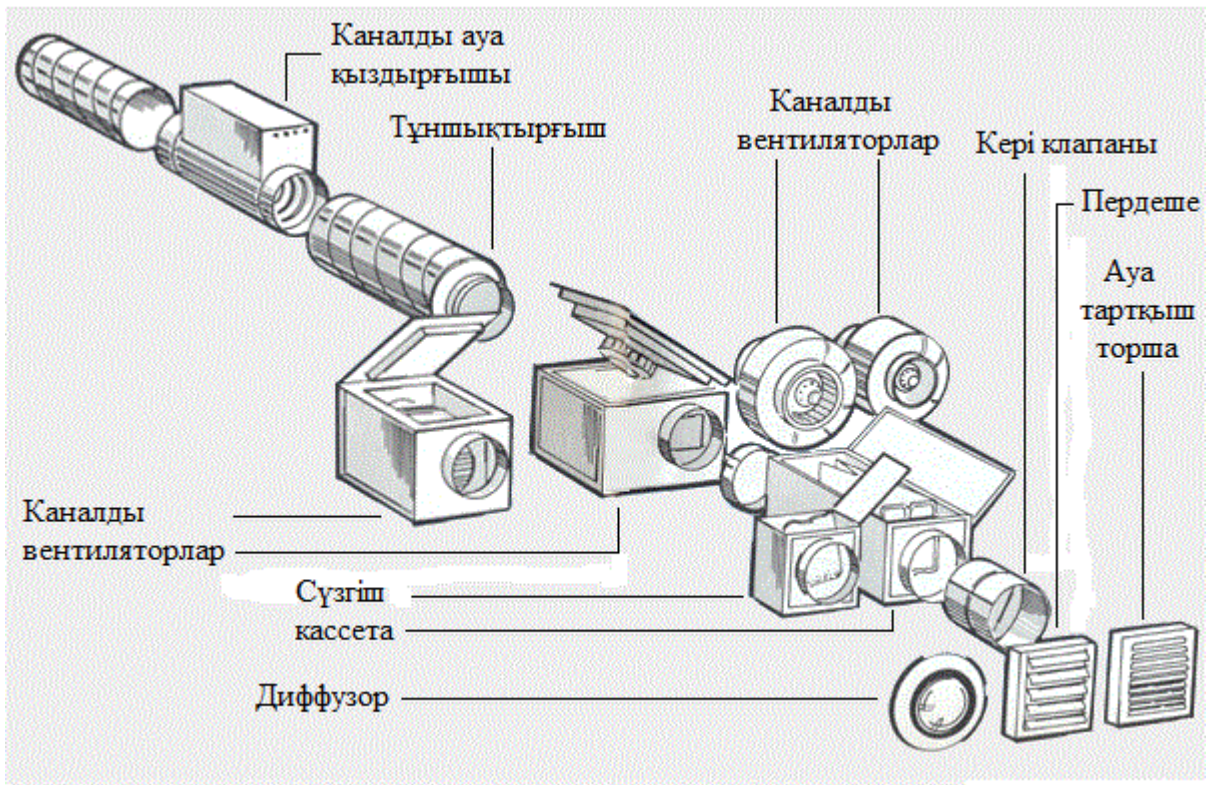
$$k = \frac{120}{216} = 0.5 \text{ адам-1} \quad (71)$$

Бөлмеде зиянды заттардың концентрациясы белгілі болған кезде, берілетін ауа көлемі келесі формула бойынша есептеледі :

$$L_6 = V_{\text{п}} * C_{\text{ф}} / C_{\text{пдк}} = 216 * 0,29 / 3.102 \approx 20 \text{ м}^3/\text{сағ} \quad (72)$$

$$C_{\text{ф}} = m * 1000 / V_{\text{п}} = 0,6264 * 1000 / 216 = 0,29 \text{ кг/м}^3$$

$$m = M * \mu / 100 = 0,06264 \text{ кг}$$



5.1-сурет – Желдету қондырғылардың орналасу сұлбасы

мұндағы, V_p - бөлменің көлемі;
 C_f - зиянды заттардың концентрациясы;
 $C_{пдк}$ – шекті рұқсат етілген концентрациясы;
 m – жоғалған газдың массасы;
 M – бөлмедегі зиянды заттардың массасы;
 μ - жоғалған газдың коэффициенті.

Артық жылуды бөлмеден шығару үшін, келесі көлемді беру қажет:

$$L_{ж} = Q_{изб} / c_p \rho_n (t_{вн} - t_{н,р}) = 5 / (0,99 * 1,225 * (292 - 288)) \approx 1,03 \text{ м}^3 \quad (73)$$

мұндағы, $Q_{изб}$ – артық жылудың қосындысы, 5 кДж/сағ;

c – ауаның жылу сиымдылығы, 0,99 кДж/(кг*К);
 ρ_n – кіретін ауаның тығыздығы, 1,225 кг/м³;
 $t_{вн}$ – бөлмедегі ауаның температурасы, 292 К;
 $t_{н,р}$ – сыртқы ауаның температурасы, 288 К.

Ауасорғыштың шығарған ауа көлемі:

$$L_M = 3600 * F * v_{\text{опт}} * K_3 = 3600 * 1,26 * 0,004 * 1,2 \approx 20 \quad (74)$$

$$F = a * b = (c + 0.8 * h) * (d + 0.8 * h) = (1 + 0.8 * 0.5) * (0.5 + 0.8 * 0.5) = 1.26 \text{ м}^2 \quad (75)$$

мұндағы, F – ауасорғыштың жұмыс ауданы м^2 ;

a және b – ұзындығы мен ені, м ;

c және d – сервердің ұзындығы мен ені, м ;

h – серверден ауасорғышқа дейінгі арақашықтық, м ;

$v_{\text{опт}}$ – зиянды заттарды жою жылдамдығы, м/с ;

K_3 – құрылғының істен шығу коэффициенті $K = 1,2$.

Желдетудің желдеткіш жүйесімен жобаланған бу мен газ тығыздығын ескеру маңызды. Егер газ тығыздығы ауа тығыздығынан төмен болса, қауіпсіздік жастықтары бөлмелердің жоғарғы жағында және бөлменің жоғарғы жағында орналасқан.

Жергілікті шаңсорғыш газдың, будың, шаңның, аэрозольдің кең тараған жерлеріне орнатылады. Бұл желдету жүйесі өндірістік бөлмедегі зиянды заттардың ауаға кіруіне жол бермейді. Жергілікті вакуумды желдету жүйесі газ және электр дәнекерлеу станцияларында, металл кесетін және тегістейтін машиналарда, шойын цехтерінде, техникалық қызмет көрсету станцияларында, көлік құралдары мен тракторларда қолданылады.

Әуе сүзгісі ауаның ластануына ұшырайтын бөлмелерде пайдаланылады, бұл ауа алмасудың жоғары деңгейін талап етеді.

Жұмыс кезінде қауіпті улы және жарылғыш заттарды алу мүмкін болса, апаттық желдету жүйелерін жобалайды. Ауа ағын саңылауларын жобалау және ашу кезінде қашықтан басқару қажеттілігін қарастырып, оны ішкі және сыртқы түрде орнатуға болады. Егер ведомстволық құжаттамада авариялық желдету жүйесі туралы ақпарат болмаса, ол қажет болған жағдайда $k \geq 8 \text{ h}^{-1}$ желдеткішті желдету арқылы ауа айналымын қажет етеді. Төтенше жағдайға қарсы желдеткіш ауа аязына тиіп тұруы керек, мұздату керек. Төтенше желдеткіш үшін қауіпті қалдықтарды жою нүктелерін жою толып жатқан жерлерде немесе басқа желдету жүйелерінде рұқсат етілмейді.

Табиғи желдету жүйелері аэрация немесе булану арналары мен миналармен жүзеге асырылады.

Аэрация - табиғи күштерге негізделген ұйымдастырылған, басқарылатын әуе күштері.

Аэрация масштабты, механикалық желдету және үлкен көлемдегі күрделі салымдар мен пайдалану шығындарында қолданылады. Желдету жүйелері немесе табиғи желдету жүйесі үшін арнайы жобаланған шахталар үшін төмен қуатты ауа ағынының мультипликаторы $k \geq 3 \div 1$ пайдаланылуы керек, табиғи желдету кезінде ауа екі бойлық қабырғаларда орналасқан ойықтары бар болуы керек. Ауа салқындаған кезде салқындатуға жол берілмейді, бірақ арнайы сақтық шаралары берілгеннен кейін ғана рұқсат етіледі.

Қорытынды

Нейрондық жартылай өткізгіштер желісі заманауи компьютерлік техникамен тиімді түрде іске асырылады.

Жартылай өткізгіш нейрондардың желісі Тьюринг машинасына өндірушіні постуландырудың бостандығына байланысты. Тюринг машиналарынан есептелген кез-келген функцияны есептейтін жүйені жасауға болады. Мысалы, осындай желіні көп қабатты перспективалық модель ретінде модельдеуге болады, бұл шешілмейтін шешім болып табылады.

Осылайша, алдын-ала аралық нейрондар мұндай желіні қосу, көбейту және белсендіру функциясын орындайтын жеке нейрондардан тұруы мүмкін.

Perceptron Learning алгоритмі осындай желіде перспективаның үзіндісін талдайтын және өзгертетін бөлек фрагмент ретінде жасалуы мүмкін.

Жартылай өткізгіш нейрондық желі автоматты квадраттың эквивалентіне тең.

Ол шамамен 95% дұрыс жауап классикалық проблема компьютерлік психодиагностика ықтималдығы 0,4 тең түрлендіргіш параметр екі нейрондардың жасанды нейрондық желі пайдалануға болады, бұл шешімдерді көрсетілді. Бұл ретте, сарапшы жүйе бағдарламашы немесе математик қатысуынсыз, тікелей маманның тәжірибесін үйрену мүмкіндігіне ие болады.

Параметр түрлендіргіш интуиция 25-30% қате эксперимент нәтижелері ретінде рәсімделген болжау қарым-қатынас, беруден беретін психологиялық дамуы мүмкін, нейрондық 16 және 0,1 санымен толық жасанды нейрондық желіде.

Әдебиеттер тізімі

- 1 Головки В.А. Нейроинтеллект: Теория и применения. Книга 1. Организация и обучение нейронных сетей с прямыми и обратными связями - Брест:БПИ, 2014. 160 б.
- 2 Головки В.А. Нейроинтеллект: Теория и применения. Книга 2. Самоорганизация, отказоустойчивость и применение нейронных сетей - Брест:БПИ, 2014. 128 б.
- 3 Уоссермен Ф. Нейрокомпьютерная техника: Теория и практика. 2014. 84 б.
- 4 Самаль Д.И., Старовойтов В.В. - Подходы и методы распознавания людей по фотопортретам. - Минск, ИТК НАНБ, 2013. 4 б.
- 5 Нейрокомпьютерная техника: теория и практика. Перевод на русский язык. Ю. А. Зуев, В. А. Точенов, 2015
- 6 Нейронные сети: основные модели. Учебное пособие к курсу "Нейронные сети". Ю.А. Зуев, Ф. Уоссермен, 2012.
- 7 С. Короткий, "Нейронные сети: Основные положения". И. В. Заенцев, 2015
- 8 Amari S. - I. The Brain and Computer // Proceedings of 2014 International Joint Conference on Neural Networks, Nagoya, Japan, October 25-29, 2014. - Nagoya, 2014. - v.1. - p.7-8.
- 9 Asary K.V., Eswaran C.A. Self-organizing Neural Network for Multidimensional Mapping and Classification of Multiple Valued Data // Proceedings of 2014 International Joint Conference on Neural Networks, Nagoya, Japan, October 25-29, 2014. - Nagoya, 2014. - v.2. - p.2488-2491.
- 10 Atamanchuk Z.M., Petrov A.A. Some problems of building and learning of neural networks while creating user's expert systems diagnoses // The RNNS-IEEE Symposium on Neuroinformatics and Neurocomputers, Rostov-on-Don, September 2015 - v.2. - pp 1133-1135.
- 11 Baxt W.G. Complexity, chaos and human physiology: the justification for non-linear neural computational analysis // Cancer Lett. - 2014. - v.77, №2-3. - p.85-93.
- 12 Becraft W.R. Diagnostic applications of artificial neural networks // Proceedings of 1993 International Joint Conference on Neural Networks, Nagoya, Japan, October 25-29, 2013. - Nagoya, 2013. - v.2. - p.2807-2810.
- 13 Bedenbaugh P., Gerstein G.L. Rectification of correlation by a sigmoid non-linearity // Biol. Cybern. - 2014. - v.70, №3. - p.219-225.
- 14 Berrios G.E., Chen E.Y. Recognizing psychiatric symptoms. Relevance to the diagnostic process // Br. J. Psychiatry. - 2013. V.163. - p.308-314.
- 15 Cohen I.L., Sudhalter V., Landon-Jimenez D. et al. A neural network approach to the classification of autism // J. Autism Dev. Disord. - 2013. - v.23, №3. - p.443-466.
- 16 Forrest D.V., Flory M.J., Anderson S. Neural network programming // N.Y.State J. Med. - 2013. - v.91, №12. - p.553.

17 Fu H.C., Shann J.J. A fuzzy neural network for knowledge learning // Int. J. Neural Syst. - 2014. - v.5, №1. - p.13-22.

18 Galushkin A.I., Savushkin S.A. Neural Network expert system // The RNNS-IEEE Symposium on Neuroinformatics and Neurocomputers, Rostov-on-Don, September 2014 - v.2. - pp 1116-1123.

19 Galushkin A.I., Sudarikov V.A., Shabanov E.V. Neuromathematic: the methods of solving problems on neurocomputers // The RNNS-IEEE Symposium on Neuroinformatics and Neurocomputers, Rostov-on-Don, September 2014 - v.2. - pp 1179-1188.

20 Modai I., Stoler M., Inbar-Saban N. et al. Clinical decisions for psychiatric inpatients and their evaluation by a trained neural network // Methods Inf. Med. - 2014 - v.32, №5. - p.396-399.

21 Sima J., Neruda R. Neural networks as expert systems // Neural Network Worl. - 2015 - v.2, №6. - p.775-783.

22 Sitting D.F., Orr J.A. A parallel implementation of the backward error propagation neural network training algorithm: experiments in event identification // Comput. Biomed Res. - 2014. - v.25, №6. - p.547-561.

23 Аванесов В.С. Тесты в социологическом исследовании. - М., 2014 - 199с.

24 Айвазян С.А., Бежаева З.И., Староверов О.В. Классификация многомерных наблюдений. - М.: Статистика, 2013 - с. 240.

25 Айвазян С.А., Бухштабер В.М., Енюков С.И., Мешалкин Л.Д. Прикладная статистика. Классификация и снижение размерности. - М.: Финансы и статистика, 2015 - с.607

26 Айвазян С.А., Енюков И.С., Мешалкин Л.Д. Прикладная статистика. Статистическое оценивание зависимостей. - М.: Финансы и статистика, 2014 - с.471.

27 Анастаси А. Психологическое тестирование. - М. Педагогика, 2014 - кн.1 - с.320, кн.2 - с.360.