### **КУРСОВА РОБОТА**

КР.ІП –16.00.00.000 ПЗ

#### Група ІП-18-1

Іванов Віктор

2020

### **КУРСОВА РОБОТА**

КР.ІП –16.00.00.000 ПЗ

#### Група ІП-18-1

Гринів Софія

2019

### **КУРСОВА РОБОТА**

КР.ІП –16.00.00.000 ПЗ

#### Група ІП-18-1

Іванов Віктор

2020

### **КУРСОВА РОБОТА**

КР.ІП –16.00.00.000 ПЗ

#### Група ІП-18-1

Гринів Софія

2019

Івано-Франківський національний технічний університет нафти і газу

**(повне найменування вищого навчального закладу)**

Кафедра програмного забезпечення автоматизованих систем

**(повна назва кафедри)**

## **КУРСОВА РОБОТА**

з дисципліни «Інтелектуальні системи аналізу даних»

#### на тему: Алгоритмізація видобування знань для визначення типу бурового розчину по нормі витрати хімреагентів

**Студента (ки) 3 курсу групи ІП-18-1**

**напряму підготовки** 121 **-** інженерія

програмного забезпечення

Іванов В. С.

**(прізвище та ініціали)**

**Керівник\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

доцент, к.т.н., Яцишин М.М

**(посада, вчене звання, науковий ступінь, прізвище та ініціали)**

**Національна шкала \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Кількість балів: \_\_\_\_\_\_\_ Оцінка: ECTS \_\_\_\_\_\_**

**Члени комісії \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**(підпис) (прізвище та ініціали)**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**(підпис) (прізвище та ініціали)**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**(підпис) (прізвище та ініціали)**

м. Івано-Франківськ - 2020 р.

Івано-Франківський національний технічний університет нафти і газу

**Кафедра** програмного забезпечення автоматизованих систем

**Дисципліна** інтелектуальні системи аналізу даних

**Спеціальність** 121 – Інженерія програмного забезпечення

**Курс 3 Група ПІ-18-1 Семестр 5**

ЗАВДАННЯ

НА КУРСОВУ РОБОТУ

**Студенту** Іванову Віктору Сергійовичу

**1. Тема роботи** Видобуток знань на основі набору даних для визначення типу пласта (колектор, покришка)

**2. Термін здачі студентом роботи** 25.12.2020

**3. Вихідні дані до роботи** типове завдання II, типи нейромереж: нейрона мережа – квантування начального вектора та нейрона мережа Машина Больцмана, ГШЗ: метод співвідношень

**4. Зміст пояснювальної записки (перелік питань, що їх належить розробити)**

Вступ, теоретична частина, практична частина, висновки, список використаної літератури

**5. Перелік графічного матеріалу (з точним зазначенням обов’язкових креслень)**

**6. Дата видачі завдання** 2 вересня 2020 р.

### **КАЛЕНДАРНИЙ ПЛАН**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № п/п | Найменування етапів | Термін виконання етапів роботи | Об'єм  роботи  (%) | Відмітка про виконання |
| 1 | Одержання завдання, ознайомлення з ним, вибір літератури | 10.09.20 | – | виконано |
| 2 | Огляд літератури і вирішення поставлених задач, обґрунтування необхідності розробки | 10.09.20 | 10 | виконано |
| 3 | Розробка структури вхідних і вихідних даних, уточнення методів рішення завдання | 01.10.20 | 15 | виконано |
| 4 | Опис і розробка алгоритму рішення завдання | 01.11.20 | 15 | виконано |
| 5 | Розробка структури програми та остаточне визначення конфігурації технічних засобів | 16.11.20 | 20 | виконано |
| 6 | Розробка програми, тестування і налагодження | 01.12.20 | 15 | виконано |
| 7 | Оформлення пояснювальної записки і практичної частини | - | 15 | виконано |
| 8 | Підготовка роботи до захисту | - | 5 | виконано |

**Студент** Іванов В. С.

розшифровка підпису

(підпис)

**Керівник** Яцишин М.М.

розшифровка підпису

(підпис)

**“ 02 ” вересня 2020 р.**

**ЗМІСТ**

Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

5

КР.ІП - 16.00.00.000 ПЗ

Розроб.

Іванов В. С.

Перевір.

Яцишин М.М.

Реценз.

Н. Контр.

Затверд.

Алгоритмізація видобування знань для визначення типу бурового розчину по нормі витрати хімреагентів

Літ.

Акрушів

ІФНТУНГ, ІП-18-1

1. Вступ......................................................................................6
2. Теоретична частина..............................................................7
3. Практична частина...............................................................13

3.1 Типове завдання.............................................................13

3.2 Виконання.......................................................................18

3.2.1 Квантування навчального вектора....................18

3.2.2 Машина Больцмана..............................................22

3.2.3 Порівняння двох мереж.......................................27

1. Висновок...............................................................................28
2. Список використаної літератури.........................................29
3. Додаток А – повний текст програми...................................30

**1. Вступ**

Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

6

КР.ІП - 16.00.00.000 ПЗ

Що ж таке нейронна мережа? Це штучний аналог біологічної мережі, який за своїми параметрами максимально наближається до оригіналу. Нейроподібні мережі пройшли довгий шлях становлення і розвитку, від повного заперечення можливості їх застосування до втілення в багато сфер діяльності людини.

Інтелектуальні системи на основі штучних нейронних мереж дозволяють з успіхом вирішувати проблеми розпізнавання образів, виконання прогнозів, оптимізації, асоціативної пам'яті і керування.

Штучні нейромережі є електронними моделями нейронної структури мозку, який, головним чином, навчається з досвіду. Природній аналог доводить, що множина проблем, які поки що не підвладні розв'язуванню наявними комп'ютерами, можуть бути ефективно вирішені блоками нейромереж.

Тривалий період еволюції додав мозку людини багато якостей, що відсутні в сучасних комп'ютерах з архітектурою фон Неймана. До них відносяться:

* розподілене представлення інформації і паралельні обчислення;
* здатність до навчання й узагальнення;
* адаптивність;
* толерантність до помилок;
* низьке енергоспоживання.

Іншими словами, нейронна мережа — це паралельна зв’язна мережа простих адаптивних елементів, яка взаємодіє з об’єктами реального світу аналогічно біологічній нервовій системі. З інженерної точки зору така мережа є динамічною системою, яка сильно розпаралелює, з топологією направленого графа, яка може виконувати переробку інформації за допомогою зміни свого стану у відповідь на постійний або імпульсний вхідний сигнал.

**2. Теоретична частина**

Нейрона мережа Машина Больцмана

Машина Больцмана — вид [стохастичної](https://uk.wikipedia.org/wiki/%D0%92%D0%B8%D0%BF%D0%B0%D0%B4%D0%BA%D0%BE%D0%B2%D0%B8%D0%B9_%D0%BF%D1%80%D0%BE%D1%86%D0%B5%D1%81) [рекурентної нейронної мережі](https://uk.wikipedia.org/wiki/%D0%A0%D0%B5%D0%BA%D1%83%D1%80%D0%B5%D0%BD%D1%82%D0%BD%D0%B0_%D0%BD%D0%B5%D0%B9%D1%80%D0%BE%D0%BD%D0%BD%D0%B0_%D0%BC%D0%B5%D1%80%D0%B5%D0%B6%D0%B0), винайденої Джеффрі Хінтоном і Террі Сейновскі 1985 року. Машина Больцмана може розглядатися як стохастичний породжувальни варіант

[мережі Гопфілда](https://uk.wikipedia.org/wiki/%D0%9D%D0%B5%D0%B9%D1%80%D0%BE%D0%BD%D0%BD%D0%B0_%D0%BC%D0%B5%D1%80%D0%B5%D0%B6%D0%B0_%D0%93%D0%BE%D0%BF%D1%84%D1%96%D0%BB%D0%B4%D0%B0).

Фахівці зі статистики називають такі мережі [випадковою марковськими полями](https://uk.wikipedia.org/wiki/%D0%9C%D0%B0%D1%80%D0%BA%D0%BE%D0%B2%D1%81%D1%8C%D0%BA%D0%B5_%D0%B2%D0%B8%D0%BF%D0%B0%D0%B4%D0%BA%D0%BE%D0%B2%D0%B5_%D0%BF%D0%BE%D0%BB%D0%B5" \o "Марковське випадкове поле). Мережу названо машиною Больцмана на честь австрійського фізика [Людвіга Больцмана](https://uk.wikipedia.org/wiki/%D0%9B%D1%8E%D0%B4%D0%B2%D1%96%D0%B3_%D0%91%D0%BE%D0%BB%D1%8C%D1%86%D0%BC%D0%B0%D0%BD" \o "Людвіг Больцман), одного з творців [статистичної фізики](https://uk.wikipedia.org/wiki/%D0%A1%D1%82%D0%B0%D1%82%D0%B8%D1%81%D1%82%D0%B8%D1%87%D0%BD%D0%B0_%D1%84%D1%96%D0%B7%D0%B8%D0%BA%D0%B0" \o "Статистична фізика).

Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

**7**

КР. ІП – 16.00.00.000 ПЗ

Ця мережа використовує для навчання [алгоритм імітації відпалу](https://uk.wikipedia.org/wiki/%D0%90%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC_%D1%96%D0%BC%D1%96%D1%82%D0%B0%D1%86%D1%96%D1%97_%D0%B2%D1%96%D0%B4%D0%BF%D0%B0%D0%BB%D1%83) і виявилася першою нейронною мережею, здатною навчатися внутрішнім поданням, розв'язувати складні [комбінаторні](https://uk.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%B1%D1%96%D0%BD%D0%B0%D1%82%D0%BE%D1%80%D0%B8%D0%BA%D0%B0" \o "Комбінаторика) завдання.

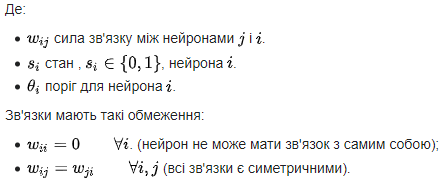
Незважаючи на це, через низку проблем, машини Больцмана з необмеженою [зв'язністю](https://uk.wikipedia.org/wiki/%D0%97%D0%B2%27%D1%8F%D0%B7%D0%BD%D0%B8%D0%B9_%D0%B3%D1%80%D0%B0%D1%84" \o "Зв'язний граф) не можуть застосовуватися для розв'язання практичних задач.

Якщо ж зв'язність обмежено, то навчання може бути достатньо дієвим для застосування на практиці. Зокрема, з каскаду [обмежених машин Больцмана](https://uk.wikipedia.org/wiki/%D0%9E%D0%B1%D0%BC%D0%B5%D0%B6%D0%B5%D0%BD%D0%B0_%D0%BC%D0%B0%D1%88%D0%B8%D0%BD%D0%B0_%D0%91%D0%BE%D0%BB%D1%8C%D1%86%D0%BC%D0%B0%D0%BD%D0%B0" \o "Обмежена машина Больцмана) будується так звана [глибинна мережа переконань](https://uk.wikipedia.org/wiki/%D0%93%D0%BB%D0%B8%D0%B1%D0%B8%D0%BD%D0%BD%D0%B0_%D0%BC%D0%B5%D1%80%D0%B5%D0%B6%D0%B0_%D0%BF%D0%B5%D1%80%D0%B5%D0%BA%D0%BE%D0%BD%D0%B0%D0%BD%D1%8C).

Як і мережа Гопфілда, машина Больцмана є мережею нейронів з визначеним для неї поняттям «енергії». Розрахунок глобальної енергії виконується ідентичним за формою з мережею Гопфілда чином:

{\displaystyle E=-\sum \_{i<j}w\_{ij}\,s\_{i}\,s\_{j}-\sum \_{i}\theta \_{i}\,s\_{i}}





Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

**8**

КР. ІП – 16.00.00.000 ПЗ

Одним з основних недоліків мережі Гопфілда є схильність до «стабілізації» стану мережі в локальному, а не в глобальному мінімумі. Практично бажано, щоб мережа переходила в глибокі мінімуми енергії частіше, ніж неглибокі, і щоб відносна ймовірність переходу мережі в один з двох мінімумів з різною енергією залежала тільки від співвідношення їх глибин.

Це дозволило б керувати можливостями отримання конкретних вихідних векторів стану шляхом зміни профілю енергетичної поверхні системи за рахунок модифікації ваг зв'язків.

На основі цих міркувань і побудовано машину Больцмана.

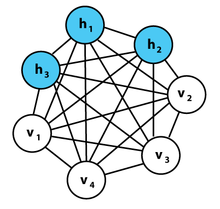
Ідея застосування «теплового шуму» для виходу з локальних мінімумів і підвищення ймовірності попадання в більш глибокі мінімуми належить С. Кірпатріку. На основі цієї ідеї розроблено [алгоритм імітації відпалу](https://uk.wikipedia.org/wiki/%D0%90%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC_%D1%96%D0%BC%D1%96%D1%82%D0%B0%D1%86%D1%96%D1%97_%D0%B2%D1%96%D0%B4%D0%BF%D0%B0%D0%BB%D1%83). Введімо деякий параметр {\displaystyle t}t — аналог рівня теплового шуму.

Тоді ймовірність активності деякого нейрона {\displaystyle k}k визначається на основі ймовірнісної функції Больцмана:



де {\displaystyle t}t — рівень теплового шуму в мережі; {\displaystyle E\_{k}} — сума ваг зв'язків *k*-го нейрона зі всіма активними на даний момент нейронами.

Приклад графічного подання машини Больцмана. В даному прикладі 3 прихованих і 4 видимих нейрона



Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

**9**

КР. ІП – 16.00.00.000 ПЗ

Нейрона мережа – квантування навчального вектора

Алгоритм квантування векторів навчання (або скорочено LVQ) - це штучний алгоритм нейронної мережі, який дозволяє нам вибирати, скільки навчальних примірників потрібно утримувати, і точно знати, як ці екземпляри повинні виглядати.

Прогнози створюються з використанням векторів кодової книги LVQ таким же чином, як і для K-найближчих сусідів.

Прогнози робляться для нового екземпляра (x) шляхом пошуку у всіх векторах кодової книги K найбільш схожих примірників і підсумовування вихідної змінної для цих K примірників.

Для класифікації це значення класу (або найбільш поширене) класу.

Зазвичай передбачення робляться з K = 1, а відповідний вектор кодової книги називається найкращою одиницею відповідності (BMU).

Щоб визначити, які з примірників K в наборі навчальних даних найбільш схожі на новий вхід, використовується міра відстані. Для речових вхідних змінних найбільш популярною мірою відстані є евклідова відстань.

Евклідова відстань розраховується як квадратний корінь з суми квадратів різниць між новою точкою (x) і існуючої точкою (xi) для кожного атрибута j.

EuclideanDistance (x, xi) = sqrt (сума ((xj - xij) ^ 2))

Алгоритм LVQ вивчає вектори кодової книги на основі даних навчання.

Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

**10**

КР. ІП – 16.00.00.000 ПЗ

Ми повинні вибрати кількість векторів кодової книги, наприклад, 20 або 40. Ви можете знайти оптимальну кількість векторів кодової книги, протестувавши різні конфігурації в вашому навчальному наборі даних.

Алгоритм навчання починається з пулу випадкових векторів кодової книги. Це можуть бути випадково вибрані екземпляри з навчальних даних або випадково згенеровані вектори з тим же масштабом, що і навчальні дані. Вектори кодової книги мають таку саму кількість вхідних атрибутів, що і навчальні дані. У них також є змінна вихідного класу.

Примірники у навчальному наборі даних обробляються по одному. Для цього навчального примірника самий релевантний вектор кодової книги вибирається з пулу.

Якщо вектор кодової книги має той же вихідний сигнал, що і навчальний екземпляр, вектор кодової книги переміщається ближче до навчального примірнику. Якщо він не збігається, він переміщається ще далі. Кількість, на яке переміщається вектор, контролюється параметром алгоритму, який називається learning\_rate.

Наприклад, вхідні змінна (x) вектора кодової книги переміщається ближче до вхідного значенням навчання (t) на величину у learning\_rate, якщо класи відповідають наступним чином:

х = х + швидкість навчання \* (т - х)

Протилежний випадок переміщення вхідних змінних змінної кодової книги від навчального примірника обчислюється як:

Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

**11**

КР. ІП – 16.00.00.000 ПЗ

х = х - швидкість навчання \* (т - х)

Це буде повторюватися для кожної вхідної змінної

Оскільки один вектор кодової книги обраний для модифікації для кожного навчального примірника, алгоритм називається «переможець отримує все».

Цей процес повторюється для кожного екземпляра в навчальному наборі даних. Одна ітерація навчального набору даних називається епохою. Процес завершено для ряду епох, які ви повинні вибрати (max\_epoch), наприклад 200.

Ми також повинні вибрати початкову швидкість навчання (наприклад, альфа = 0,3). Швидкість навчання зменшується з епохою, починаючи з великого значення, яке ви вказуєте в епоху 1, який вносить найбільша зміна в вектори кодової книги, і закінчуючи невеликим значенням, близьким до нуля, в останню епоху, вносячи дуже незначні зміни в вектори кодової книги.

Швидкість навчання для кожної епохи розраховується як:

learning\_rate = alpha \* (1 - (epoch / max\_epoch))

Де learning\_rate - це швидкість навчання для поточної епохи (від 0 до max\_epoch-1), alpha - це швидкість навчання, зазначена алгоритму на

початку тренувального циклу, а max\_epoch - це загальна кількість епох для запуску алгоритму, також зазначеного в початок пробігу.

Інтуїція для процесу навчання полягає в тому, що пул векторів кодової книги являє собою стиснення набору навчальних даних до точок, які найкраще характеризують поділ класів.

Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

**12**

КР. ІП – 16.00.00.000 ПЗ

Зазвичай рекомендується підготувати дані для LVQ так само, як ви готували б їх для найближчих сусідів.

класифікація: LVQ - алгоритм класифікації, який працює як для довічних (двокласних), так і для багатокласових алгоритмів класифікації. Техніка була адаптована для регресії.

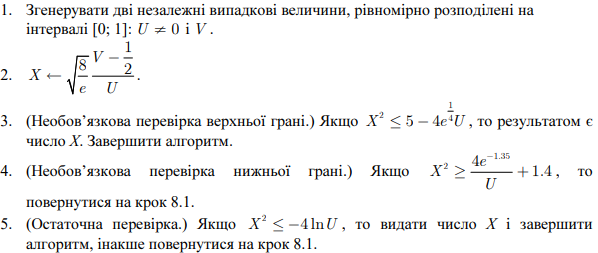
Множинні Перепустки: Хороша техніка з LVQ включає в себе виконання декількох проходів навчального набору даних по векторах кодової книги (наприклад, багаторазове навчання). Перший з більш високою швидкістю навчання для визначення векторів кодової книги пулу, а другий - з невеликою швидкістю навчання для точної настройки векторів.

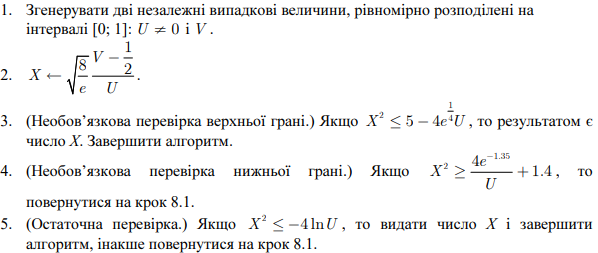
Кілька найкращих матчів: Розширення LVQ вибирають кілька кращих відповідних одиниць для зміни під час навчання, наприклад, один і той же клас і один з іншого класу, які тягнуться до навчальної вибірки і від неї відповідно. Інші розширення використовують власну швидкість навчання для кожного вектора кодової книги. Ці розширення можуть поліпшити процес навчання.

Нормалізувати входи: Традиційно входи нормалізуються (масштабуються) до значень від 0.

Метод співвідношень

алгоритм:





Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

**13**

КР. ІП – 16.00.00.000 ПЗ

**3. Практична частина**

3.1 Типове завдання

Видобуток знань на основі набору даних для визначення типу пласта (колектор, покришка)

Алгоритм видобування знання:

Для вирішення питань порівняння нафтогазогеологічних об’єктів застосовують метод аналогій. Застосування методу передбачає пошук і вивчення природного аналогу, виявлення ступеня подібності об’єкта з аналогом та перенесення ознак аналога на об’єкт вивчення. Для приблизної оцінки подібності складних нафтогазогеологічних об’єктів широко використовують методи розпізнавання образів. Під терміном ”образ“ розуміють сукупність (множину) об’єктів певного класу, що характеризуються загальними ознаками.

Методи розпізнавання образів базуються на принципі аналогій. Суть цих методів полягає у співставленні комплексу ознак досліджуваного об’єкта з комплексом цих же ознак еталонних об’єктів. При розпізнавання образів нафтогазогеологічних об’єктів проводиться їх розподіл по сукупності відомих ознак, які характеризують ці об’єкти, як правило, на два класи з різкопротилежними властивостями.

У нафтогазовій геології методи розпізнавання образів дозволяють одержувати результат з мінімальним ризиком прийняття рішень, ефективно застосовувати їх для вирішення таких задач:

а/ прогнозування продуктивних структур за комплексом геолого-геофізичних ознак;

б/ виділення нафтоносних і водоносних пластів за каротажними даними;

в/ поділ розрізу свердловин на колектори і покришки;

г/ визначення нафтоносності структур за результатами хімічного аналізу пластових вод;

д/ виділення фонових і аномальних ділянок за геохімічними даними;

е/ прогнозу нафтовіддачі нового об’єкта.

Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

**14**

КР. ІП – 16.00.00.000 ПЗ

Для реалізації методу розпізнавання образів необхідно мати такі початкові дані:

- еталонну вибіркову сукупність, для якої поділ на класи /образи/ відомі;

- набір одинакових ознак, що характеризують кожний із елементів обох вибірок;

- прогнозну вибірку, кожний елемент якої потребує прийняття рішення про віднесення його до одного із заданих класів.

Дискримінантний аналіз - один із статистичних методів розпізнавання образів, що базується на підборі так званої дискримінантної функції, яка розділяє багатомірні дані сукупностей на наперед задані групи таким чином, щоб була досягнута максимальна днорідність всередині групи і мінімальна між ними. Для підбору дискримінантної функції використовується масив еталонних об’єктів для кожної завчасно заданих груп.

Обчислені на еталонних даних параметри /коефіцієнти/ дискримінантної функції використовуються для прогнозу - віднесення нових /невідомих по статусу/ об’єктів до однієї із груп.

Звідси виходить, що задача поділу /дискримінації/ зводиться до підбору

коефіцієнтів дискримінантної функції таким чином, щоб різним групам відповідали різні значення функції.

Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

**15**

КР. ІП – 16.00.00.000 ПЗ

На практиці часто зустрічаються задачі поділу багатомірних об’єктів, що характеризуються не двома, а трьома і більше ознаками. В найбільш загальному випадку геометрична інтерпретація дискримінантної функції являє собою гіперплощину в k - мірному ознаковому просторі, а кожний об’єкт є точка цього ж простору. Необхідно провести в цьому просторі таку гіперплощину, яка забезпечувала б максимальну

відмінність між множинами об’єктів, що належать різним класам /образам/ і приводила б до мінімуму розсіювання всередині кожної множини. Таку гіперплощину і називають дискримінантною.

Аналітично гіперплощина в k - мірному просторі має вигляд:

D = а1·Х1 + а2·Х2 + ··· + аk·Хk,

де D - дискримінантна функція, а1,а2,...,аk – коефіцієнти дискримінантної функції,

Х1,Х1,...,Хk - значення ознак об’єкта.

Отже, задача полягає у знаходженні коефіцієнтів а1,а2,...,аk, які б забезпечили умови поділу, що вимагаються.

Якщо дискримінантна функція знайдена, то множина її значень складається із двох підмножин, кожна із яких належить тільки одному із класів /образів/. Існує таке граничне значення D0, яке ділить область значень дискримінантної функції на дві підплощини. Значення показників об’єкта підставляється як аргументи в побудовану на еталонній виборці дискримінантну функцію.

Якщо значення дискримінантної функції досліджуваного об’єкта більше граничного значення D0, то об’єкт відносять до класу

/образу/ А, в протилежному випадку - до класу В. Для ілюстрації наведемо такий приклад. Досліджуваний регіон перспективний в нафтогазоносному відношенні.

Тут виявлено кілька десятків структур, частина із яких виявились перспективними /клас А/, частина водоносними /клас В/, а частина інших /нерозбурених/ структур має невиявлену природу. Необхідно розділити нові нерозбурені структури і визначити першочергові із них для проведення пошукового буріння.

За результатами вивчення геолого-геофізичних властивостей колекторів та покришок нафтових і газових родовищ нафтогазоносного регіону було встановлено, що колектори характеризуються одним співвідношенням таких параметрів, як пористість, глинистість, карбонатність і амплітуда ВП, а покришки іншим співвідношенням цих параметрів.

Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

**16**

КР. ІП – 16.00.00.000 ПЗ

При вивченні розрізу однієї із перспективних площ були отримані геологогеофізичні характеристики горизонтів (пористість, глинистість, карбонатність і амплітуда ВП).

Перед геологами стоїть завдання використовуючи встановлені геолого-геофізичні характеристики колекторів і покришок розділити горизонти перспективної площі на колектора та покришки.

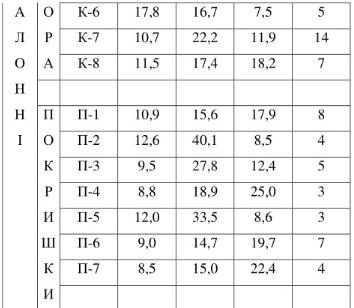
Для вирішення поставленої задачі ми повинні дані, що знаходяться в таблиці 2, виконати роботу в наступній послідовності:

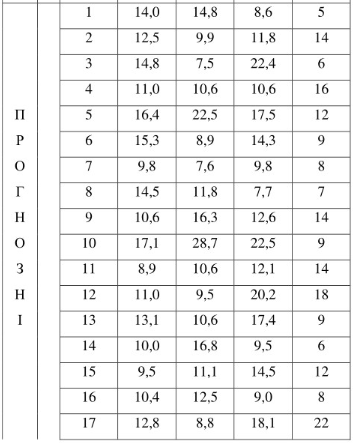
1. Визначити значення дискримінантної функції для кожного із об’єктів.

2. Навчити нейромережу (відповідно до варіант) розрізняти за даними пористості, глинисності, карбонатності і амплітуді - це колектор чи покришка.

3. Перевірити результат навчання на згенерованих штучних даних.

Таблиця 2 - Вихідні дані до проведення дискримінантного аналізу геологопромислових даних





Змн.

Арк.

№ докум.

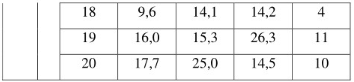
Підпис

Дата

Арк.

**17**

КР. ІП – 16.00.00.000 ПЗ



**3.2 Виконання**

Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

**18**

КР. ІП – 16.00.00.000 ПЗ

3.2.1 Квантування навчального вектора

Алгоритм:

1) перед початком роботи:

* подаємо на вхід вектори
* виділяємо кластери
* формуємо вагову матрицю

2) ініціалізуємо індекс швидкості навчання (learningSpeed)

3) продовжуємо кроки 4-9, якщо немає передумов для зупинки

4) проходимо кроки 5-6 для кожного начального вектора

5) вираховуємо відстань до 1ого і 2ого кластеру

6) виявляємо «переможця» (за рахунок мінімальної відстані)

7) корегуємо ваги

8) зменшуємо коефіцієнт швидкості навчання

9) перериваємо навчання:

якщо досягнуто максимальну кількість ітерацій

Фрагменти коду:

1 а) подаємо на вхід вектори:



б) виділяємо кластери

в) формуємо вагову матрицю



Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

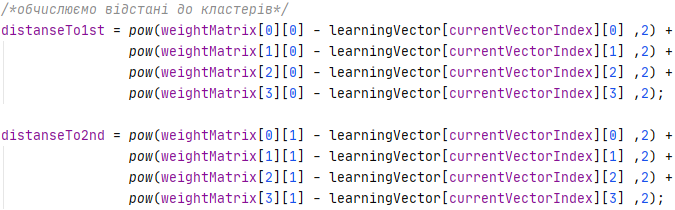
**19**

КР. ІП – 16.00.00.000 ПЗ

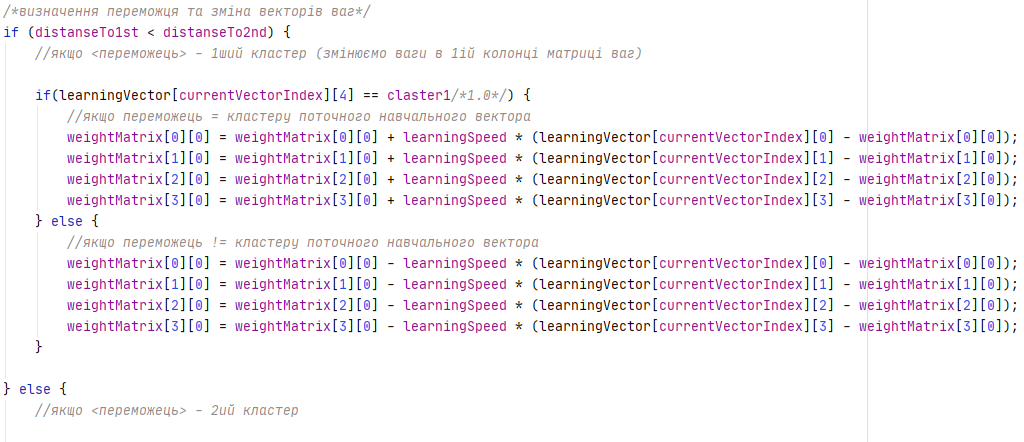
2) ініціалізуємо індекс швидкості навчання



5) вираховуємо відстань до 1ого і 2ого кластеру



6/7) виявляємо «переможця» /змінюємо ваги



Змн.

Арк.

№ докум.

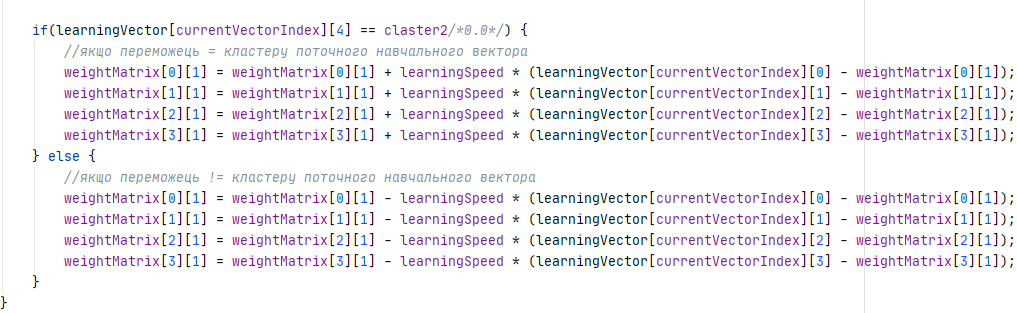
Підпис

Дата

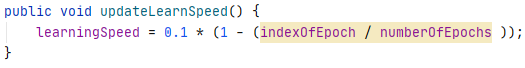
Арк.

**20**

КР. ІП – 16.00.00.000 ПЗ

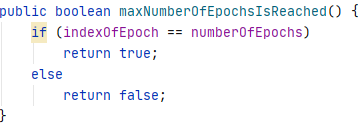


8) зменшуємо коефіцієнт швидкості навчання



9) умова переривання навчання





вивід:

Змн.

Арк.

№ докум.

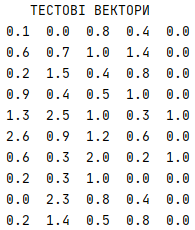
Підпис

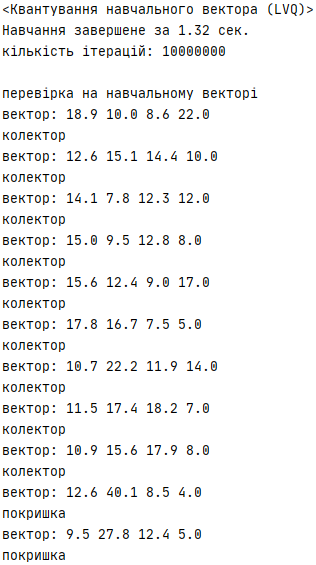
Дата

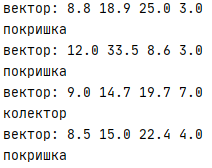
Арк.

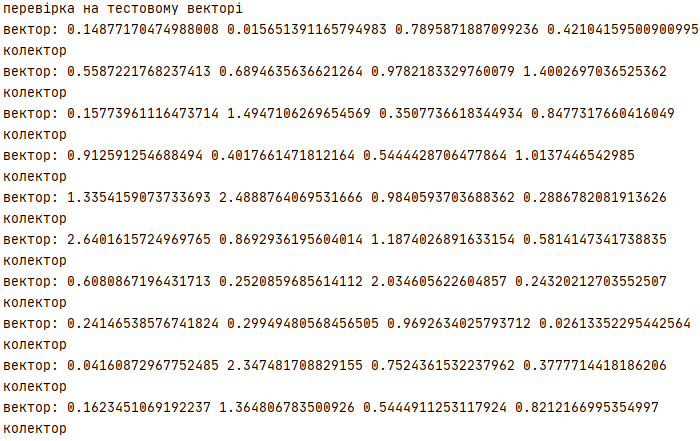
**21**

КР. ІП – 16.00.00.000 ПЗ









Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

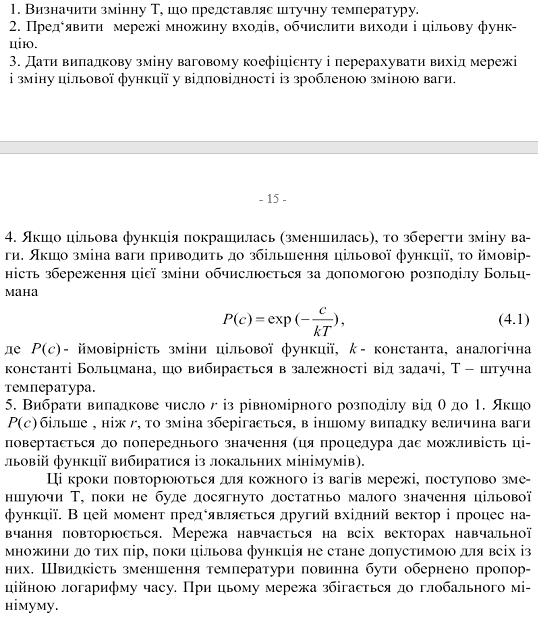
**22**

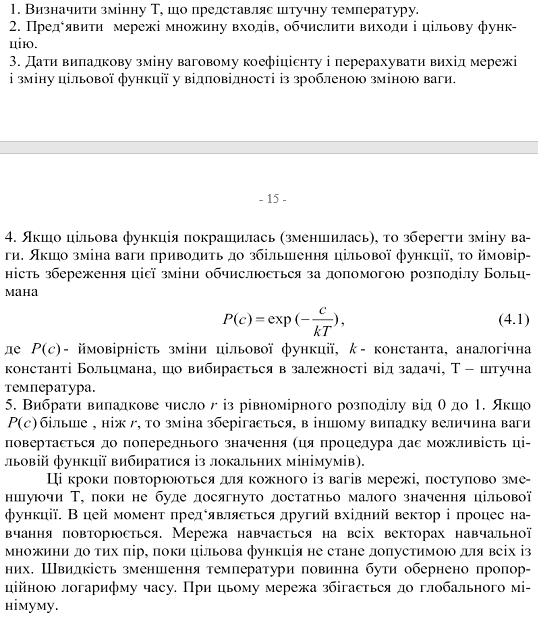
КР. ІП – 16.00.00.000 ПЗ

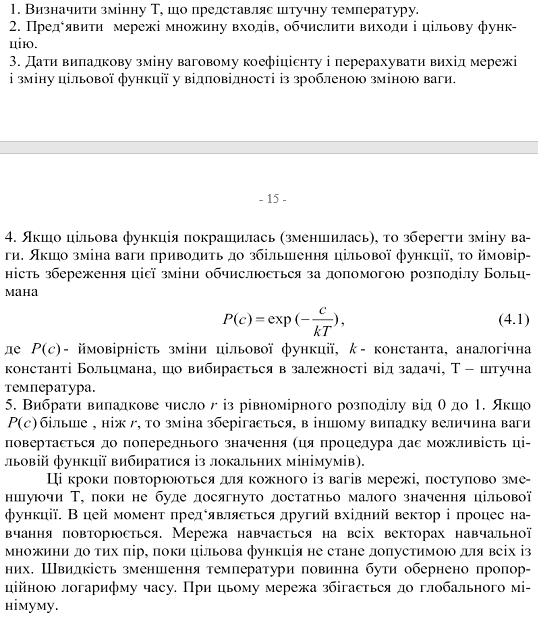
Нейромережа «Квантування навчального вектора» завершила навчання за 1.32 сек та продемонструвала точність на навчальному векторі 86.66%.

3.2.2 Машина Больцмана

алгоритм:







Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

**23**

КР. ІП – 16.00.00.000 ПЗ

Фрагменти коду:

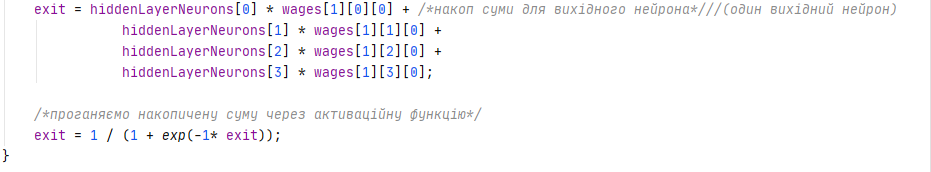
1) визначаємо штучну температуру











Змн.

Арк.

№ докум.

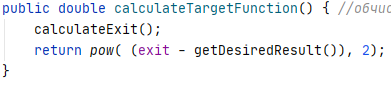
Підпис

Дата

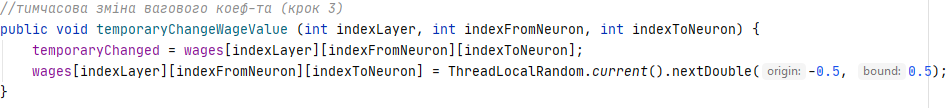
Арк.

**24**

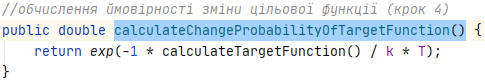
КР. ІП – 16.00.00.000 ПЗ







4. розподіл Больцмана



5. 



вивід:

Змн.

Арк.

№ докум.

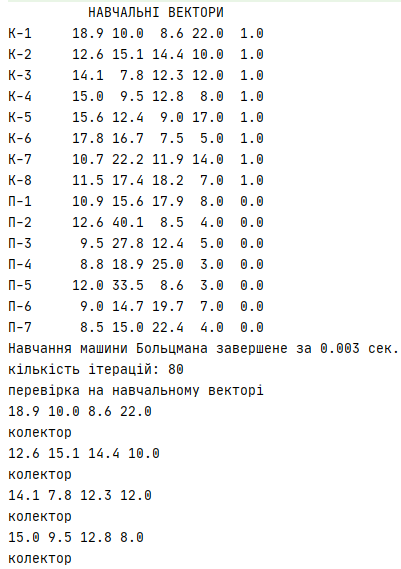
Підпис

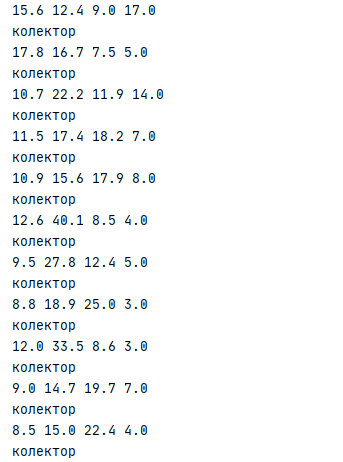
Дата

Арк.

**25**

КР. ІП – 16.00.00.000 ПЗ





Змн.

Арк.

№ докум.

Підпис

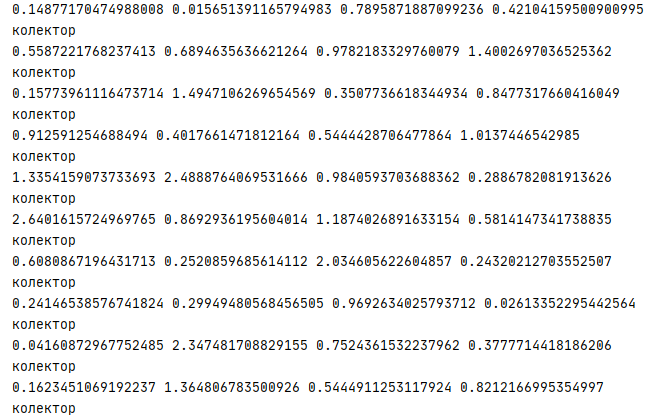
Дата

Арк.

**26**

КР. ІП – 16.00.00.000 ПЗ





Машина Больцмана завершила роботу за 0.003 сек та показала точність при тестуванні на навчальному векторі 53.33%.

3.2.3 Порівняння нейронних мереж

Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

**27**

КР. ІП – 16.00.00.000 ПЗ

|  |  |  |
| --- | --- | --- |
| Критерії | Квантування навчального вектора | Машина Больцмана |
| Швидкість | висока (1.32 сек за 10 млн ітерацій) | середня (0.003 сек за 80 ітерацій) |
| Точність | 86.66% | 53.33% |
| Простота алгоритму | простий | середньої складності |
| Основні причини поганої роботи мережі | не виявлені | коефіцієнт навчання занадто малий чи великий, можливість паралічу мережі, потрапляння мережі в локальний мінімум |
| Кількість шарів нейронів | 2 шари | 3 шари |
| Вагові коефіцієнти | вибираються з навчальних векторів | Можна задавати будь-які вагові коефіцієнти в діапазоні |
| Можливості мережі | необмежені можливості мережі | обмежені можливості мережі |
| Обмеження на образів, що запам'ятовуються | необмежена кількість образів | необмежена кількість образів |
| Навчання (з вчителем/без учителя) | Із учителем | Із учителем |

Зважаючи на наведені переваги та недоліки цих методів, як кращий був обраний метод квантування навчального вектора.

**4. Висновок**

Дані нейронні мережі вцілому впоралися з поставленним завданням.

Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

**28**

КР. ІП – 16.00.00.000 ПЗ

За легкістю проектування нейронної мережі, «Квантування навчального вектора» випереджає машину Больцмана.

«Квантування навчального вектора» показала найкращі результати та продемонструвала кращу швидкість навчання.

Дана курсова робота є підтвердженням того, що нейромережі мають широкий спектр застосування.

**5.Список використаної літератури**

<https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%83%D1%87%D0%B5%D0%BD%D0%B8%D0%B5_%D0%B1%D0%B5%D0%B7_%D1%83%D1%87%D0%B8%D1%82%D0%B5%D0%BB%D1%8F>

<https://habr.com/ru/post/318970/>

<https://wiki.loginom.ru/articles/local-minimum.html>

https://docs.exponenta.ru/deeplearning/ug/learning-vector-quantization-lvq-neural-networks-1.html

https://www.elibrary.ru/item.asp?id=38558208

Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

**29**

КР. ІП – 16.00.00.000 ПЗ

**6. Додаток А**

Змн.

Арк.

№ докум.

Підпис

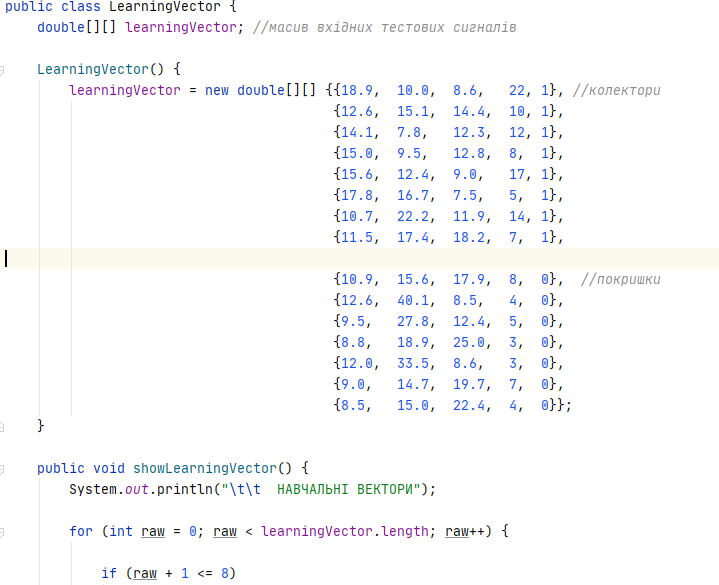
Дата

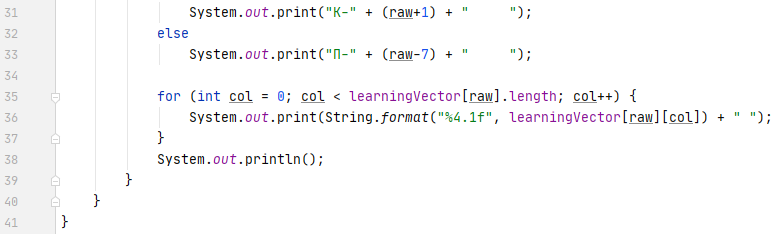
Арк.

**30**

КР. ІП – 16.00.00.000 ПЗ

клас LearningVector





клас TestingVector

Змн.

Арк.

№ докум.

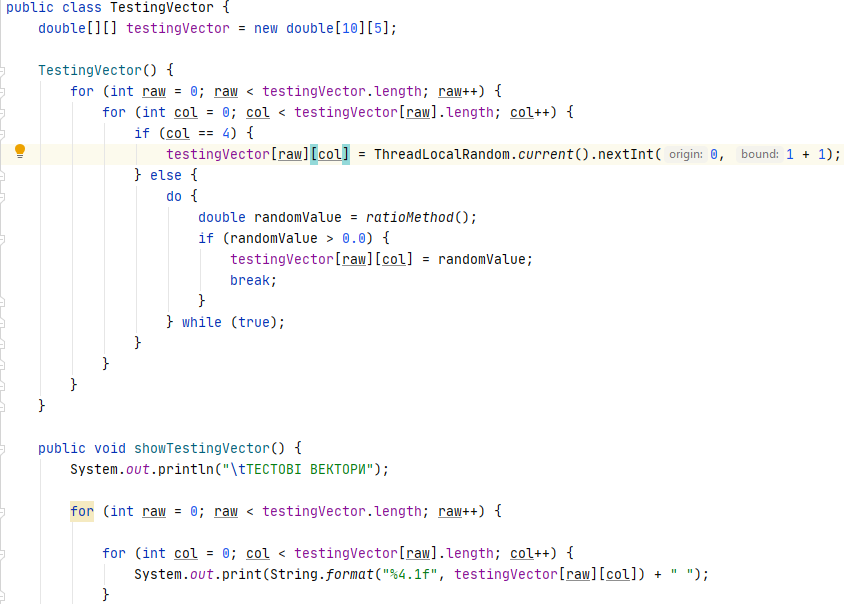
Підпис

Дата

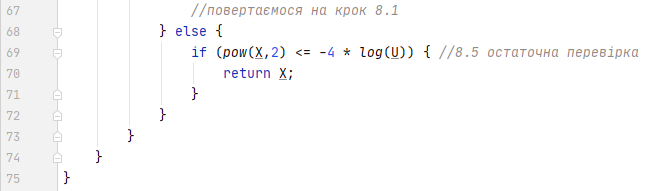
Арк.

**31**

КР. ІП – 16.00.00.000 ПЗ







Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

**32**

КР. ІП – 16.00.00.000 ПЗ

клас BolzmannMachine



Змн.

Арк.

№ докум.

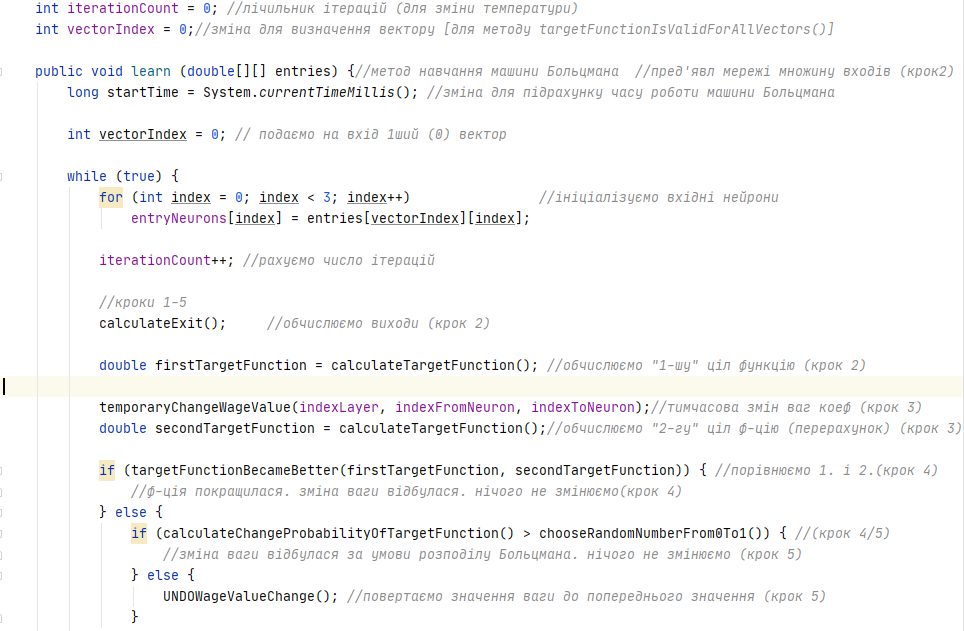
Підпис

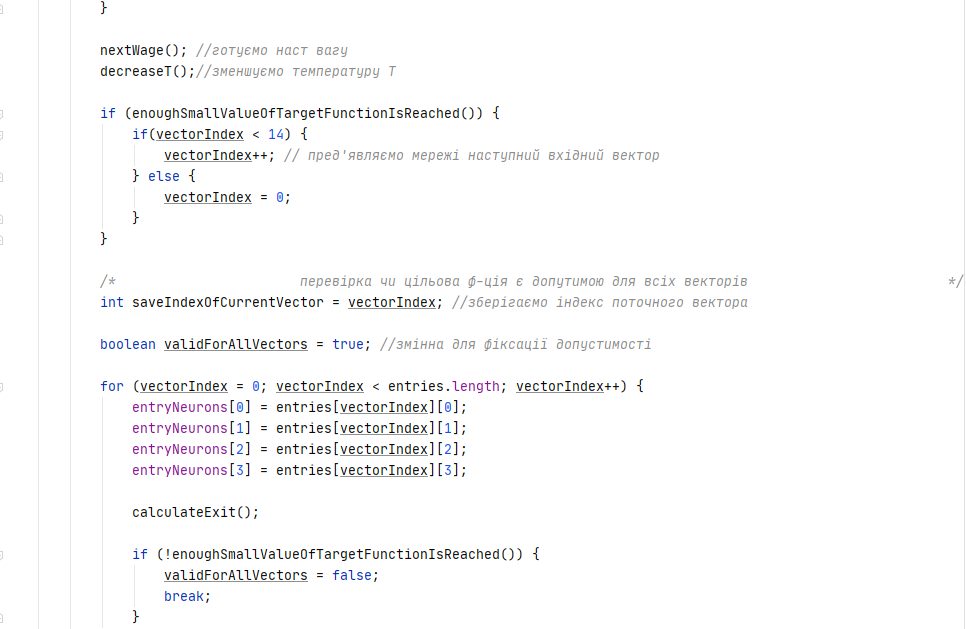
Дата

Арк.

**33**

КР. ІП – 16.00.00.000 ПЗ





Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

**34**

КР. ІП – 16.00.00.000 ПЗ





Змн.

Арк.

№ докум.

Підпис

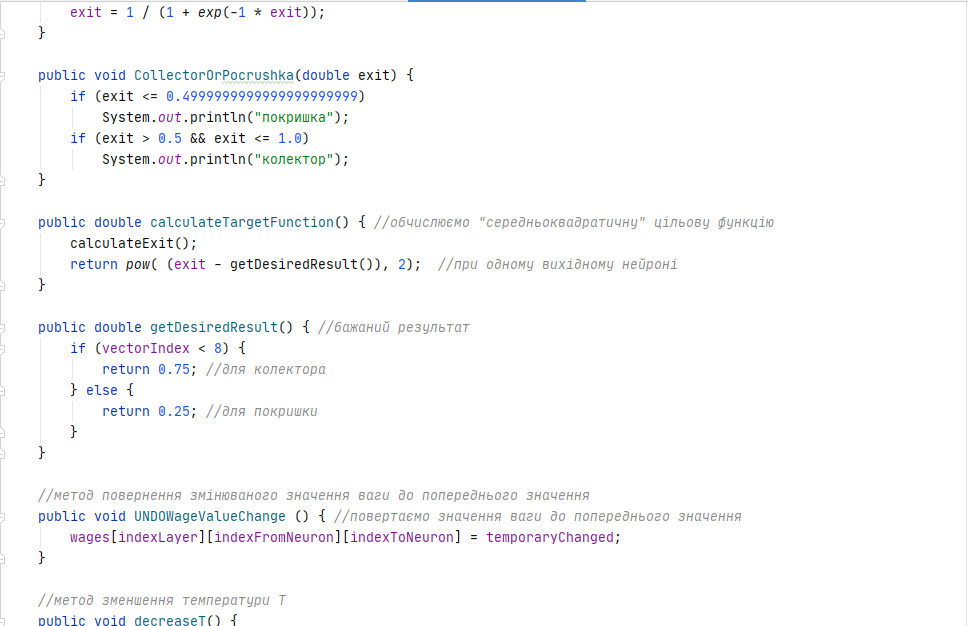
Дата

Арк.

**35**

КР. ІП – 16.00.00.000 ПЗ





Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

**36**

КР. ІП – 16.00.00.000 ПЗ





Змн.

Арк.

№ докум.

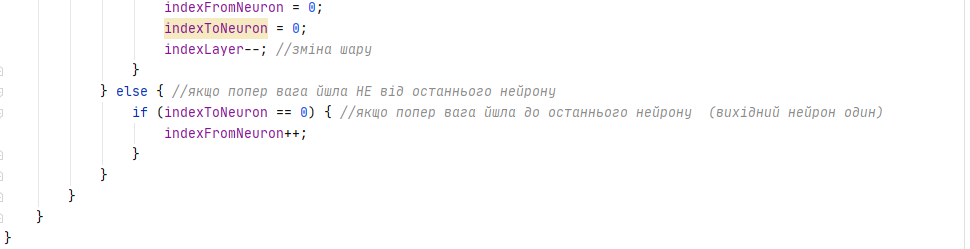
Підпис

Дата

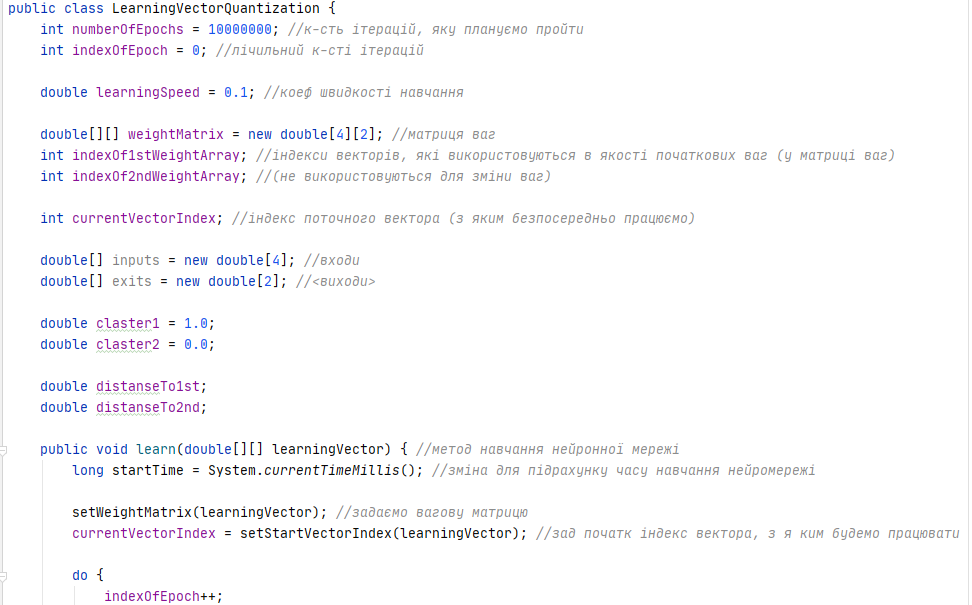
Арк.

**37**

КР. ІП – 16.00.00.000 ПЗ



клас LearningVectorQuantization



Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

**38**

КР. ІП – 16.00.00.000 ПЗ





Змн.

Арк.

№ докум.

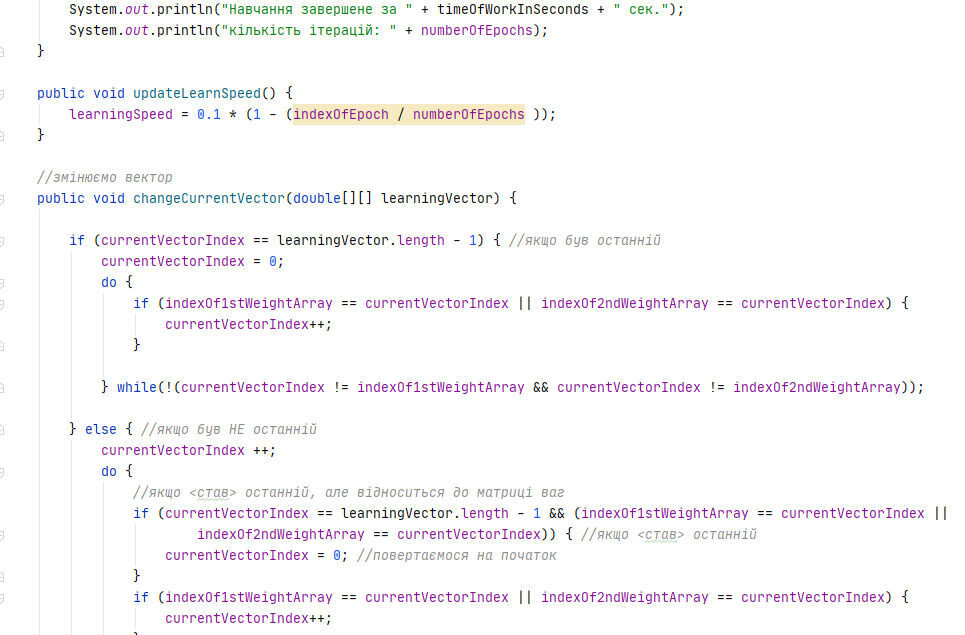
Підпис

Дата

Арк.

**39**

КР. ІП – 16.00.00.000 ПЗ





Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

**40**

КР. ІП – 16.00.00.000 ПЗ





клас Test

Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

**41**

КР. ІП – 16.00.00.000 ПЗ

