Міністерство освіти і науки України
Харківський національний університет імені В. Н. Каразіна
Факультет комп'ютерних наук
Кафедра теоретичної та прикладної системотехніки

«Затверджую»
Зав. кафедри теоретичної та
прикладної системотехніки
_____ д.т.н., проф. С. І. Шматков
«___» _____ 2021 р

Пояснювальна записка

до кваліфікаційної роботи
магістра

на тему: «**МОДЕЛЬ ІНФОРМАЦІЙНОЇ СИСТЕМИ КЛАСИФІКАЦІЇ ПАЦІЄНТІВ ЗА ДОПОМОГОЮ ЙМОВІРНІСНИХ ШТУЧНИХ МЕРЕЖ»**

Захищено на засіданні
Атестаційної комісії № 39
протокол № __ від __.12.2021 р.
Оцінка _____ / _____
Голова Атестаційної комісії
_____Мінухін С.В.

Виконала:

студентка 6 курсу, групи КУ– 61
Галузь знань: 15 – Автоматизація та приладобудування
за спеціальністю 151 – Автоматизація та комп'ютерно-інтегровані технології.
МАКСИМУК Анастасія Родіонівна

Керівник:
к. т. н., доцент; доцент кафедри теоретичної та прикладної системотехніки
БАКУМЕНКО Ніна Станіславівна

Рецензент:
доктор ф.-м. н., проф.,
професор кафедри освітніх та інформаційних технологій Національного фармацевтичного університету
ПОГОРЄЛОВ Станіслав Вікторович

Харків – 2021

АНОТАЦІЯ

Пояснювальна записка до кваліфікаційної роботи магістра складається зі вступу, трьох розділів, висновків, списку використаних джерел і 4 додатків. Загальний обсяг роботи складає 64 сторінок, із яких 52 сторінок основної частини з 14 рисунками та 2 таблицями, 26 найменувань списку використаних джерел на 4 сторінках і 4 додатки на 12 сторінках.

Дана робота присвячена створенню комп'ютерної класифікаційної моделі станів медико-біологічної системи за допомогою ймовірнісних штучних мереж.

Об'єктом дослідження є процес класифікації станів медико-біологічної системи за допомогою методів машинного навчання.

Предметом дослідження є модель розпізнавання станів медико-біологічної системи .

Актуальність даної роботи полягає у необхідності надійного діагностування захворювань шляхом застосування інтелектуальних методів аналізу даних.

Дана робота включає в себе чітко сформульовані цілі та завдання проекту, теоретичні дані, що є необхідними для правильного формулювання знань в даній галузі. Обговорено питання розробки, тестування та аналізу комп'ютерної моделі.

Ключові слова: КЛАСИФІКАЦІЯ, ШТУЧНІ НЕЙРОННІ МЕРЕЖІ, ЙМОВІРНІСНІ НЕЙРОННІ МЕРЕЖІ, АРХІТЕКТУРА НЕЙРОННИХ МЕРЕЖ, СТАТИСТИЧНІ ДАНІ.

ABSTRACT

Thesis consists of an introduction, three chapters, conclusions, list of references and appendices. The total amount of work is 64 pages, 52 pages of which the main part of the 14 figures and 2 tables, four pages list of references with 26 titles, 4 applications at 12 pages.

Thesis is devoted to the creation of a computer classification model of the states of a biomedical system using probabilistic artificial networks.

The object of research is the process of classifying the states of a biomedical system using machine learning methods.

The subject of this research is a model of recognition of the states of a biomedical system.

The relevance of this work lies in the need for reliable diagnosis of diseases through the use of intelligent data analysis methods.

Thesis includes clearly formulated goals and objectives of the project, theoretical data necessary for the correct formulation of knowledge in this area. The issues of development, testing and analysis of a computer model were discussed.

Key words: CLASSIFICATION, ARTIFICIAL NEURAL NETWORKS, PROBABILISTIC NEURAL NETWORKS, ARCHITECTURE OF NEURAL NETWORKS, STATISTICAL DATA.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ, ТЕРМІНІВ

ЙНМ – ймовірнісна нейронна мережа

ЗНМ – згорткові нейронні мережі

МН – машинне навчання

ОВМ – опорно-векторна машина

ШІ – штучний інтелект

ШНМ – штучні нейронні мережі

AIC – Akaike Information Criterion, інформаційний критерій Акаіке.

AUC – area under curve, площа під кривою

GPS – Global Positioning System, система глобального позиціонування

FN – False Negatives, хибно негативні випадки;

FP – False Positives, хибно позитивні випадки

FPR – False Positives Rate, доля хибно-позитивних зразків

PNN – Probabilistic Neural Network, ймовірнісні штучні мережі.

ROC – Receiver Operator Characteristic, робоча характеристика приймача

RF – Random Forest, випадковий ліс.

S_e – Sensitivity, чутливість

S_p – Specificity, специфічність

TN – True Negatives, істинно негативні випадки;

TP – True Positives, істинно-позитивні випадки

TRP – True Positives Rate, доля істинно-позитивних зразків

TV – training-and-validation, навчальна колекція

ВСТУП

З розвитком людської цивілізації накопичувались і знання про навколишнє середовище, явища, дії навколо повсякденного життя. Набуваючи певних знань шляхом експериментів люди впорядковували об'єкти відповідно за їх спільними властивостями, параметрами та ознаками. Подібне сортування, часто, відігравало велику роль навіть в збереженні життя людей того часу. Для прикладу можна розглянути поділ їстівних чи неїстівних предметів, або поділ противників за рівнем небезпеки.

Саме цей процес сортування людьми об'єктів різного типу згідно зі спільними показниками й отримав назву — класифікація. З цього можна зробити висновок, що зі збільшенням знань про світ, різко зростала значущість класифікації .

На примітивних прикладах, наведених вище, все доволі зрозуміло, але в загальному випадку, процес чіткого та ефективного аналізу інформації є досить об’ємним та складним. Коротко кажучи, є непідсильним для людини. Для вирішення цієї проблеми вимагаються нові методи. У зв’язку з автоматизацією різних сфер людської життєдіяльності, логічно використовувати автоматизовані математичні методи і до поняття класифікації.

Дана кваліфікаційна робота робота присвячена найбільш популярній задачі машинного навчання — класифікації [1].

Метою даної роботи є розробка комп’ютерної моделі класифікації станів медико-біологічної системи за допомогою ймовірнісних штучних мереж.

Щоб реалізувати поставлену мету необхідно вирішити наступні **завдання**:

- сформулювати задачу класифікації ознак в медико-біологічних системах,

- провести поширений аналіз існуючих методів класифікації об’єктів,
- зробити детальний огляд обраного методу для вирішення задачі,
- скласти математичну модель вирішення задачі класифікації за допомогою ймовірнісних штучних мереж,
- розробити програмно-алгоритмічну модель системи класифікації,
- виконати тестування моделі та зробити аналіз отриманих результатів.

Об’єктом дослідження є застосування та використання моделі класифікації за допомогою ймовірнісних штучних мереж в медичній діагностиці.

Предметом дослідження є математичні методи класифікації.

Практична значимість результатів полягає у використанні розробленої моделі класифікації в медичній діагностиці.

Актуальність даної роботи полягає у необхідності надійного діагностування захворювань шляхом застосування інтелектуальних методів аналізу даних [2].

РОЗДІЛ 1.
ОГЛЯД МЕТОДІВ КЛАСИФІКАЦІЇ

1.1 Постановка задачі класифікації

Класифікація — один з розділів машинного навчання, присвячений вирішення наступного завдання: є безліч об'єктів (ситуацій), розділених деяким чином на класи.

Основне завдання класифікації полягає в розбитті певної кількості елементів даних на категорії або класи так, щоб всі елементи всередині кожного класу мали достатню кількість загальних ознак, що дозволяє знехтувати їх індивідуальними відмінностями [3].

Об’єкт класифікації — це елемент класифікаційної множини, що має ті чи інші властивості, так звані ознаки класифікації, за якими класифікуються об’єкти.

Формальний запис задачі класифікації має наступний вигляд. Нехай $D = \{d_1, \dots , d_n\}$ — множина об‘єктів, $C = \{c_1, \dots , c_к\}$ - множина категорій, Ф — цільова функція, яка по парі $< d_i, c_j >$ визначає, чи відноситься об’єкт d_i до категорії c_j (1 або True) або ні (0 або False). Задача класифікації полягає в побудові функції Ф', максимально близької до Ф.

Методи машинного навчання, які застосовуються для класифікації, передбачають наявність колекції заздалегідь класифікованих експертами об'єктів, тобто таких, для яких вже точно відомо значення цільової функції. Для того щоб після побудови класифікатора можна було оцінити його ефективність, ця колекція розбивається на дві частини, не обов'язково рівного розміру:

1. Навчальна (training-and-validation, TV) колекція. Класифікатор Ф' будується на основі характеристик цих об'єктів.

2. Тестова (test) колекція. На ній перевіряється якість класифікації. Об'єкти з test не повинні використовуватися в процесі побудови класифікатора.

Сукупність методів і правил класифікації та її результат становлять систему класифікації.

Класифікація має численні застосування у різних сферах:

- розпізнавання рукописного тексту;
- класифікація даних;
- класифікація зображень;
- прогнозування банкрутства;
- розпізнавання мови;
- медичне діагностування;
- виявлення несправностей.

Для розв’язання задачі класифікації використовують значну кількість підходів [4] . А саме:

- байєсовий класифікатор;
- класифікація за допомогою дерева рішень;
- класифікація за допомогою нейронних мереж;
- класифікація з використанням методу опорних векторів;
- класифікація за допомогою генетичного алгоритму;
- класифікація методом найближчого сусіда;
- логістична регресія.

Кожен з цих методів має як переваги, так і недоліки. Розглянемо деякі з них детальніше в наступному підрозділі.

1.2 Методи класифікації

Приведемо теоретичні відомості відносно деяких з вище приведених методів[5].

1.2.1 Наївний байєсів класифікатор

Наївний байєсів класифікатор – простий ймовірнісний класифікатор, в основі роботи якого лежить застосування теореми Байєса з «наївною» пропозицією про незалежність ознак між собою. Він належить до найпростіших моделей мережі Байєса.

Наївна модель Байєса передбачає, що екземпляри потрапляють до одного з ряду взаємовиключних та вичерпних класів $C \in \{C_1,\ C_2, \dots, C_n\}$.

Модель також включає деяку кількість ознак $X_1,\ \dots, X_n$, значення яких зазвичай спостерігаються.

Наївне припущення Байєса полягає в тому, що функції умовно не залежать від класу екземпляра. У кожному класі екземплярів різні властивості можуть бути визначені незалежно.

Алгоритм класифікації базується на теоремі Байєса (1):

$$P(h|e) = \frac{P(e|h)P(h)}{P(e)} \tag{1}$$

Використовуючи формулу повної ймовірності, отримуємо формулу, яка є лежить в основі розрахунків для наївного байєсового класифікатора (2):

$$P(h|e) = \frac{P(e|h)P(h)}{P(e|h)P(h)+P(e|\neg h)P(\neg h)} \tag{2}$$

Ці міркування використовуються в алгоритмі побудови мережі Байєса [6].

Цей метод має багато застосувань, зокрема в автоматизованій медичній діагностиці [7].

1.2.2 Лінійний класифікатор

Лінійний класифікатор – алгоритм класифікації, заснований на побудові лінійної розділяє поверхні. У випадку двох класів поверхнею, яка розділяє, є гіперплощина, яка ділить простір ознак на два півпростору. У разі більшої кількості класів, поверхня, що розділяє, кусочно-лінійна [3].

Нехай маємо тренувальний набір даних $(y_i, x_i), i = 1, \dots, l, y_i \in \{-1, +1\}$, $x_i \in R^n$. Потрібно побудувати функцію ухвалення рішень (3)

$$d(x) = sign(w^T x - w_0), \tag{3}$$

де w – вектор вагів.

Рівняння $w^T x = w_0$ описує гіперплощину, яка розділяє класи в просторі R^n.

Лінійна класифікація є корисним інструментом у машинному навчанні та аналізі даних. На відміну від нелінійних класифікаторів, лінійні класифікатори безпосередньо працюють з даними у просторі вихідних змінних. Важливою перевагою лінійної класифікації є те, що процедури навчання та тестування набагато ефективніші. Недоліком можна вважати те, що метод непристосований до нероздільних лінійно даних, які неможливо розділити за допомогою гіперплощини [8].

1.2.3 Метод опорних векторів

Метод опорних векторів (SVM – Support Vector Machine), розроблений в серії робіт В. Вапніка та А. Червоненкіса [9-10] – це метод, пристосований як для класифікації, так і для регресії. Алгоритм опорних векторів відноситься до контрольованих алгоритмам машинного навчання [11]. Алгоритм методу опорних векторів широко використовується для вирішення задач класифікації в різних сферах діяльності, зокрема, і в медичній діагностиці [12].

Характерною особливістю SVM-класифікатор є використання ядра – спеціальної функції, яка використовується для відображення тренувального набору даних з вихідного простору характеристик в простір більш високої розмірності, в якому будується гіперплощину, що розділяє класи. По обидва боки гіперплощини, що розділяє, задаються дві паралельні гіперплощини, що визначають класи і знаходяться якнайдалі один від одного. Вважається, що чим більше відстань між цими площинами, тим менше похибка класифікатора. Вектори характеристик об'єктів, що класифікуються, найближчі до паралельним гіперплощин, називаються опорними векторами. Приклад площин, що розділяють, наведений на рис. 1.1.

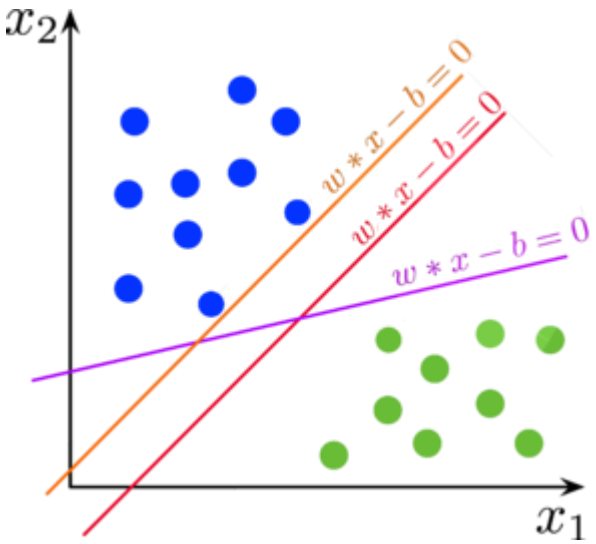

Рисунок 1.1 – Приклад площин, що розділяють, для методу опорних векторів

Якщо позначити тренувальний набір даних як $(y_i, x_i), i = 1, \dots, l,\ y_i \in \{-1, +1\}$, $x_i \in R^n$, для навчання SVM-класифікатора треба визначити опорну гіперплощину, яка задається рівнянням (4):

$$w^T x - b = 0 \qquad (4)$$

Умова $-1 < w^T x - b < 1$ визначає смугу, що розділяє класи.

Опорними векторами називаються об'єкти, які найближчі розділяючої гиперплощини. Вони розташовані точно на межах смуги і містять всю інформацію про їх поділи.

У разі лінійної роздільності класів можна вибрати гіперплощини таким чином, щоб між ними не містився жоден об'єкт з навчальної вибірки, потім максимізувати відстань між гіперплощинами (ширину смуги). Для цього потрібно вирішити задачу квадратичної оптимізації [12] (5):

$$\begin{cases} w^T w \to min \\ y_i(w^T x_i + b) \geq 1, i = 1, \dots, l \end{cases} \quad (5)$$

При вирішенні задачі класифікації у випадку лінійної нероздільності класів, задачу побудови гіперплощини, що розділяє, можна сформульовати як задачу пошуку сідлової точки функції Лагранжа, яка є задачею квадратичного програмування (6):

$$\begin{cases} -L(\lambda) = -\sum_{i=1}^{l} \lambda_i + \frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l} \lambda_i \lambda_j y_i y_j \, \mathrm{k}(x_i, x_j) \to \min\limits_{\lambda} \\ \sum_{i=1}^{l} \lambda_i y_i = 0 \\ 0 \leq \lambda_i \leq C, \ i = 1, \dots, l \end{cases} \quad (6)$$

де $\mathrm{k}(x_i, x_j)$ – функція ядра, C – параметр регуляризації ($C > 0$).

Для навчання SVM-класифікатору необхідно визначити тип функції $\mathrm{k}(x_i, x_j)$ і значення параметра регуляризації C, які дозволяють знайти компроміс між максимізацією ширини смуги, що розділяє класи, і мінімізацією сумарної похибки. При цьому в якості функції ядра, як правило, використовується одна з функцій:

– лінійна $\mathrm{k}(x_i, x_j) = {x_i}^T x_j$;

– поліноміальна $\mathrm{k}(x_i, x_j) = ({x_i}^T x_j + 1)^d$;

– радіально-базисна $\mathrm{k}(x_i, x_j) = exp\left[-\left(x_i - x_j\right)^T (x_i - x_j) \Big/ (2\sigma^2)\right]$;

– сігмоїдна $\mathrm{k}(x_i, x_j) = th(k_2 + k_1(x_i{}^T x_j))$.

Після навчання отримуємо функцію, що класифікує (7):

$$d(x) = sign(\sum_{i=1}^{l} \lambda_i y_i \mathrm{k}(x_i, x_j) + b). \quad (7)$$

Перевагою методу опорних векторів є нечутливість до перенавчання та невимогливість до обчислювальних ресурсів. Недоліками є необхідність налаштування – вибору функції ядра та параметру регуляризації.

1.2.4 Моделі логістичної регресії

Логістичні моделі регресії широко використовуються як засоби класифікаційного аналізу даних, які дозволяють спрогнозувати ймовірність приналежності об’єкта до певного класу [12].

У медицині моделі логістичної регресії часто використовується для оцінки ймовірності захворювання тією чи іншою хворобою, оцінки ймовірності виявлення злоякісних утворень, класифікації пацієнтів з певним групам ризику та ін. [13, 14].

Задачу бінарної класифікації (класифікація вибірки за двома категоріями), причому мітки цільового класу позначимо "+1" (позитивні приклади) і "-1" (негативні приклади), реалізує наступна функція:

$$d(x) = sign\,(w^T x), \quad (8)$$

де x – вектор ознак досліджуваних об’єктів, w – ваги у лінійній моделі.

Логістична регресія є окремим випадком лінійного класифікатора, яка здатна прогнозувати ймовірність p відношення окремого об'єкту x_i до певного класу (9):

$$p = P(y_i = 1|x_i) \quad (9)$$

Проблему прогнозування значення бінарної змінної можна переформулювати як пошук безперервної змінної, значення якої лежать в інтервалі від 0 до 1 при будь-яких значеннях незалежних змінних. Це можна досягти за допомогою наступного регресійного рівняння (10):

$$p = \frac{1}{1+e^{-y}}, \quad (10)$$

де p – ймовірність того, що відбудеться відповідна подія; $y = w^T x + w_0$.

Для пошуку коефіцієнтів логістичної регресії найчастіше використовується метод максимальної правдоподібності. Він призначений для отримання оцінок параметрів генеральної сукупності за даними вибірки. Метод оснований на застосуванні функції правдоподібності, яка виражає щільність ймовірності (ймовірність) спільної появи результатів вибірки $Y_1, Y_2, \dots, Y_k$ (11):

$$L(Y_1, Y_2, \dots, Y_k; \theta) = p(Y_1; \theta) \cdot \dots \cdot p(Y_k; \theta) \quad (11)$$

Згідно з методом максимальної правдоподібності в якості оцінки невідомого параметра приймається таке значення $\Theta = \Theta(Y_1, \dots, Y_k)$, яке максимізує функцію L.

Для спрощення обчислень замість функції L зручно використовувати Знаходження оцінки спрощується, якщо максимізувати не саму функцію L, а

$ln\,(L)$, оскільки максимум обох функцій досягається при одному і тому ж значенні θ (12):

$$ln(L(Y;\theta)) \rightarrow max \qquad (12)$$

Моделі логістичної регресії – потужний класифікаційний інструмент, який надає, між іншим, опис взаємозв'язку між вхідними та вихідною змінною, він є простим в інтерпретації. До недоліків можна також віднести необхідність вибору з множини вхідних параметрів найвпливовіших для включення в модель.

На підставі аналізу методів класифікації, наведених вище, для вирішення задачі діагностики стану пацієнтів було обрано метод логістичної регресії, завдяки таким перевагам: даний метод демонструє короткий час навчання, є досить потужним для бінарної класифікації, використовує замість прямої лінії криву, що спрощує розподіл даних на групи.

Важливим моментом є те, що метод логістичної регресії активно вводиться в світову медицину для аналізу результатів досліджень [15]. Це пояснюється наявністю ряду особливостей даного методу. До них відносяться [16]:

1. Визначення для конкретної групуючої ознаки Y, набору ознак-предикторів Xi, що пояснюють наборами своїх значень ймовірності віднесення певного спостереження до групи порівняння.

2. Впорядкування відібраних ознак-предикторів X_i за рівнем впливу на залежну ознаку Y.

3. Оцінка надійності приналежності пацієнтів до конкретного класу залежної ознаки Y, за допомогою певної комбінації відібраних ознак-предикторів X_i.

4. Можливість оцінки не одного, а багатьох рівнянь логістичної регресії.

5. Вибір різних наборів потенційних предикторів, виходячи з яких алгоритми, що використовуються оцінюють різні рівняння логістичної регресії.

1.2.5 Нейромережеві моделі

Нейронними мережами називається один з напрямків наукових досліджень штучного інтелекту (ШІ). В його основі лежить імітація нервової системи людини. Головними властивостями, що беруться до уваги є самонавчання та здатність виправляти помилки. Саме ці властивості дозволяють змоделювати роботу людського мозку. Як висновок, можна сказати Штучні нейронні мережі (ШНМ) функціонують за принципами, аналогічними біологічним нейронам нервової системи [9-11] .

Штучні нейронні мережі – клас моделей, побудованих із використанням алгоритмів машинного навчання з урахуванням принципу коннекціонізму – припущення про те, що розумові явища можуть бути описані мережами з взаємопов'язаних простих елементів, за аналогією з організацією біологічних нейронних мереж. ІНС імітує поведінку системи, виходячи з наданих експериментальних або відомих з інших джерел даних, дозволяючи пропустити етап створення алгоритмічної/механічної моделі, необхідний опису системи та рішення пов'язаних з нею завдань при традиційному підході та представляє значні труднощі для складних та нелінійних систем, часто що зустрічаються у завданнях з галузі біології ШНМ успішно застосовуються в різних областях – там, де потрібно рішення задач прогнозування, класифікації та управління. Розглянемо декілька прикладів.

1. **Нейронні мережі у бізнесі.** Найбільш поширене застосування – системи розпізнавання.

2. **Нейронні мережі та фінанси**. За допомогою нейронних мереж можна передбачити, наприклад, яким буде курс валюти через деякий проміжок часу, виходячи зі статистичної інформації.

3. **Нейронні мережі у економіці**. Нейронні мережі можуть вирішувати задачі економічно-статистичного моделювання, наближуючи методи, що вивчаються до економічної реальності.

4. **Нейроуправління.** Нейронні мережі здійснюють управління найрізноманітнішими рухомими об'єктами, наприклад, електропечі, зварювальні апарати, безпілотні автомобілі, літаки і т.д.

5. **Нейронні мережі у хімії і біології.** За допомогою нейронних мереж можна полегшити створення медичних препаратів, шляхом прогнозування хімічних властивостей та біологічних активностей хімічних з'єднань.

Незамінним засобом при вирішенні складних багатовимірних задач є саме нейронні мережі, адже вони мають можливість нелінійного моделювання в поєднанні з простою реалізацією.

Нейронні мережі являють собою нелінійні системи, за допомогою яких можна краще класифікувати даних, ніж за допомогою лінійних методів. Нейронні мережі здатні прийняти рішення, базуючись на виявленні певних прихованих закономірностей в даних. Вони не програмуються, не використовують ніяких правил виведення, для постановки діагнозу, а навчаються робити це на чітко визначених прикладах.

Розглянемо переваги та недоліки методу [17].

Переваги ШНМ:

- Зберігання інформації по всій мережі: інформація, така як у традиційному програмуванні, зберігається у всій мережі, а не в базі даних. Зникнення кількох відомостей в одному місці не заважає мережі функціонувати.
- Здатність працювати з неповними знаннями: після навчання з ШНМ дані можуть давати результат навіть із неповною інформацією. Втрата продуктивності тут залежить від важливості відсутньої інформації.

- Наявність стійкості до вад: пошкодження однієї або декількох комірок ШНМ не заважає їй генерувати вихід. Ця функція робить мережі стійкими до вад.
- Наявність розподіленої пам’яті: для того, щоб ШНМ могла навчатись, необхідно визначити приклади та навчити мережу відповідно до бажаного результату, показуючи ці приклади мережі. Успіх мережі прямо пропорційний вибраним екземплярам, і якщо подія не може бути показана мережі у всіх її аспектах, мережа може видавати помилкові результати.
- Можливість паралельної обробки: штучні нейронні мережі мають чисельну потужність, яка може виконувати більше одного завдання одночасно.

Недоліки штучних нейронних мереж:

- Незрозуміла поведінка мережі: це найважливіша проблема ШНМ. Коли мережа виробляє прогнозуюче рішення, вона не дає поняття, чому і як. Це зменшує довіру до мережі.
- Визначення належної структури мережі: не існує конкретного правила для визначення структури штучних нейронних мереж. Відповідна структура мережі досягається завдяки досвіду та методом спроб та помилок.
- Складність подання проблеми в мережі: ШНМ можуть працювати з числовою інформацією. Проблеми повинні бути переведені в числові значення перед введенням в ШНМ. Механізм відображення, який буде визначено, безпосередньо впливатиме на продуктивність мережі.
- Тривалість часу навчання мережі невідома: система навчається до досягнення певного значення помилки на вибірці.

1.3 Архітектури штучних нейронних, що використовуються для розв'язування задач класифікації

Оскільки всі штучні нейроні мережі базуються на концепції нейронів, з'єднань і передатних функцій, існує подібність між різними структурами або архітектурами нейроних мереж. Більшість відмінностей залежить від різних правил навчання. Розглянемо деякі відомі моделі штучних нейромереж.

1.3.1 Багатошаровий перцептрон

Елементарний перцептрон складається з елементів 3-х типів: S-елементів, A-елементів і одного R-елемента (Рисунок)[18].

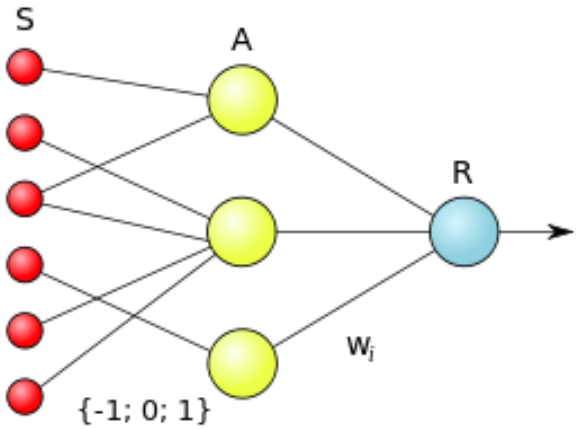

Рисунок 1.3.1.1 – Елементарний перцептрон

S-елементи - це шар сенсорів, або рецепторів. У фізичному втіленні вони відповідають, наприклад, світлочутливим клітинам сітківки ока або фоторезистора матриці камери. Кожен рецептор може знаходитися в одному з двох станів - спокою або збудження, і тільки в останньому випадку він передає одиничний сигнал в наступний шар, асоціативним елементам.

A-елементи називаються асоціативними, тому що кожному такому елементу, як правило, відповідає цілий набір (асоціація) S-елементів. A-елемент активізується, як тільки кількість сигналів від S-елементів на його вході перевищило деяку величину θ. Таким чином, якщо набір відповідних S-елементів розташовується на сенсорному полі у формі букви "Д", A-елемент

активізується, якщо достатня кількість рецепторів повідомило про появу "білої плями світла" в їх околиці, тобто А-елемент буде як би асоційований з наявністю / відсутністю літери "Д" в деякій області.

Сигнали від А-елементів, у свою чергу, передаються в суматор R, причому сигнал від і-го асоціативного елемента передається з коефіцієнтом w_i . Цей коефіцієнт називається *вагою* AR зв'язку.

Так само як і А-елементи, R-елемент підраховує суму значень вхідних сигналів, помножених на ваги. R-елемент, а разом з ним і елементарний перцептрон, видає "1", якщо лінійна форма перевищує поріг θ, інакше на виході буде "-1". Математично, функцію, що реалізовується R-елементом, можна записати так:

$$f(x) = sign(\sum_{i=0}^{n} w_i x_i - \theta) \qquad (13)$$

Навчання елементарного перцептрона полягає у зміні вагових коефіцієнтів w_i зв'язків AR. Ваги зв'язків SA і значення порогів А-елементів вибираються випадковим чином на самому початку і потім не змінюються.

Після навчання перцептрон готовий працювати в режимі розпізнавання або узагальнення . У цьому режимі перцептрону пред'являються раніше невідомі йому об'єкти, і перцептрон повинен встановити, до якого класу вони належать. Робота перцептрона полягає в наступному: при пред'явленні об'єкта, А-елементи передають сигнал R-елементу, який дорівнює сумі відповідних коефіцієнтів w_i. Якщо ця сума позитивна, то приймається рішення, що даний об'єкт належить до першого класу, а якщо вона негативна - то до другого.

1.3.2 Ймовірнісні нейронні мережі

Ймовірнісні нейронні мережі (PNN – Probabilistic Neural Network) – нейронні мережі, які на тлі інших інтелектуальних засобів, що можуть бути використані для ідентифікації систем, мають істотні переваги.

Апарат ймовірнісних нейронних мереж може бути використаний, наприклад, для дослідження наступних процесів:

- прогнозування електричних навантажень;
- прогнозування станів та якостей поверхневих вод;
- розподіл навантажень мережі між інформаційними потоками і т.д.

1.3.3 Мережі Кохонена

Нейронні мережі Кохонена – клас нейронних мереж, що використовують навчання без учителя.

Основним принципом роботи мереж Кохонена є введення у правило навчання нейрона інформації про його розташування.

Мережа Кохонена відноситься до мереж, що самоорганізуються, які під час надходження вхідних сигналів не отримують інформацію про бажаний вихідний сигнал. Усі подані вхідні сигнали із заданої навчальної множини мережа розділяє на класи, будуючи так звані топологічні мапи.

Мережу Кохонена використовують для відображення нелінійних взаємозв’язків даних на достатньо легко інтерпретовані сітки (частіше за все двомірні), які являють собою метричні та топологічні залежності вхідних векторів, що об’єднуються у кластери [19].

Зобразимо структуру нейронної мережі Кохонена.

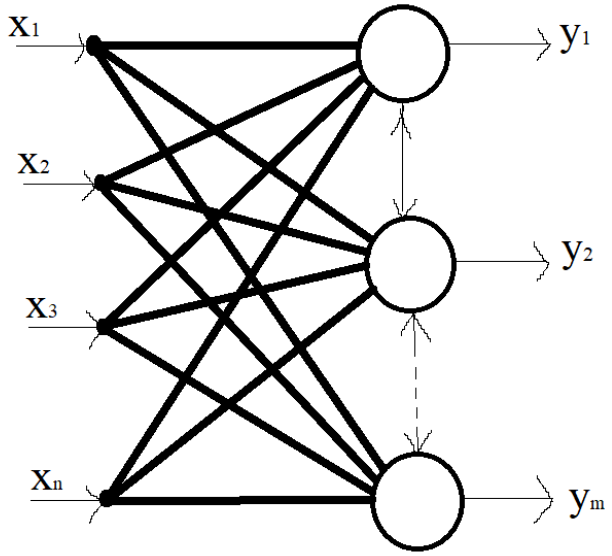

Рисунок 1.1 – Структура мережі Кохонена

Опишемо мережу за допомогою формул:

$$w_m = (w_{1m}, w_{2m}, w_{3m\ldots} w_{nm}), \quad x(i) = \big(x_1(i), x_2(i), \ldots, x_n(i)\big), \quad (14)$$

де i = 1,2,…n – вектор вхідних сигналів

Вихідний вектор:

$$y = (y_1, y_2, \ldots, y_m), \quad (15)$$

де m – кількість кластерів.

Наведемо матрицю множини вагових коефіцієнтів:

$$W = \begin{pmatrix} W^{11} & W^{11} & \ldots & W^{11} \\ W^{11} & W^{11} & \ldots & W^{11} \\ . & . & \ldots & . \\ W^{11} & W^{11} & \ldots & W^{11} \end{pmatrix} \quad (16)$$

де $W^{ij} = (w_1^{ij}, w_2^{ij}, \ldots, w_n^{ij})$ – вектори вагових коефіцієнтів.

Кожен нейрон характеризується своїм розміщенням у шарі й ваговим коефіцієнтом. Розміщення нейронів – є деякою метрикою й визначається топологією шару, при якій сусідні нейрони під час навчання впливають один на одного сильніше, ніж розташовані далі. Кожен нейрон утворює зважену

суму вхідних сигналів з $w_{ij} > 0$, якщо синапси прискорювальні, і $w_{ij} < 0$ – гальмуючі.

Наявність зв’язків між нейронами призводить до того, що при збудженні одного з них можна обчислити збудження інших нейронів у шарі.

1.3.4 Згорткові нейронні мережі

Згорткові нейронні мережі (ЗНМ, CNN, ConvNet) – це клас глибинних штучних нейронних мереж прямого поширення, який застосовується до аналізу зображень. ЗНМ використовують різновид багатошарових перцептронів, розроблений так, щоб вимагати використання мінімальної обробки. Виходячи з їхньої архітектури спільних ваг та характеристик інваріантності відносно паралельного перенесення [20].

ЗНМ – тип багатошарової нейронної мережі, яка свою назву «згорткова мережа» отримала за назвою операції – згортка, вона часто використовується для обробки зображень і може бути описана наступною формулою:

$$(f \times g)[m,n] = \sum_{k_1,1} f[m-k, n-l] \cdot g[k,l], \qquad (1.16)$$

де f – вихідна матриця, g – ядро згортки [100].

Ідея згорткових нейронних мереж полягає в чергуванні згорткових шарів і субдискретизуючих шарів.

ЗНМ складається з шарів входу та виходу, а також із декількох прихованих шарів. Приховані шари ЗНМ зазвичай складаються зі згорткових шарів, агрегувальних шарів, повноз'єднаних шарів та шарів нормалізації. Згорткові шари застосовують до входу операцію згортки, передаючи результат до наступного шару. Згортка імітує реакцію окремого нейрону на зоровий стимул.

Модель згорткової мережі складається з трьох типів шарів:

- згорткові шари,

- субдискретизуючі верстви;
- прошарки перцептрона.

Архітектура згорткових нейронних мереж реалізує три ідеї, які забезпечують інваріантність мережі до невеликих зрушень, змін масштабу і спотворень:

- кожен нейрон отримує вхідний сигнал від локального рецептивного поля (local receptive fields) у попередньому шарі, що забезпечує локальну двовимірну зв'язність нейронів;
- кожен прихований шар мережі складається з безлічі карт ознак, на яких всі нейрони мають загальні ваги (shared weights), що забезпечує інваріантність до зміщення і скорочення загальної кількості вагових коефіцієнтів мережі;
- за кожним шаром згортки слідує обчислювальний шар, який здійснює локальне усереднення та підвибірку, що забезпечує зменшення розширення для карт ознак.

Робота згорткової нейронної мережі забезпечується двома основними елементами.

1) Фільтри (filters) (визначники ознак).

2) Карти ознак (feature maps).

Фільтр – це невелика матриця, що представляє ознаку, яку необхідно знайти на вихідному зображенні. За допомогою верхнього фільтра визначаються частини вихідного зображення з вертикальними лініями, нижній фільтр служить для визначення частин зображення з горизонтальними лініями. Безпосередньо процес визначення заснований на операції згортки фільтром оригінального зображення.

Результати згортки, які визначають місце розташування ознак вихідного зображення, називаються **картами ознак**.

Мета процесу згортки – зменшити розмірність карти ознак до такої міри, щоб з повним набором ознак могла працювати мережа прямого поширення.

Згортковий шар реалізує ідею локальних рецептивних полів, тобто кожен вихідний нейрон з’єднаний тільки з певною (невеликою) областю вхідної матриці і таким чином моделює деякі особливості людського зору. Недоліками згорткових нейронних мереж (ЗНМ) є:

- висока складність архітектури;
- повнозв’язаність;
- фіксована площа вікна шару згортки.

З метою підвищення ефективності роботи ЗНМ необхідно знайти оптимальні значення наступних параметрів:

- кількість карт ознак;
- щільність зв'язків між картами ознак;
- розмір вікна;
- площа перекриття;
- початкова ініціалізація ваг.

Висновки до першого розділу

В даному розділі було приведено основні теоретичні матеріали відносно поняття «класифікація». Розглянуто методи, за допомогою яких можна реалізувати задачу класифікації. Наведемо стисло з’ясовані дані.

Класифікація — один з розділів машинного навчання, присвячений вирішення наступного завдання: є безліч об'єктів, розділених деяким чином на класи.

Класифікація має численні застосування у різних сферах: розпізнавання рукописного тексту, класифікація даних, класифікація зображень, прогнозування банкрутства, розпізнавання мови, медичне діагностування, виявлення несправностей.

Задачу класифікації можна реалізувати за допомогою наступних методів: байєсовий класифікатор, класифікація за допомогою дерева рішень, класифікація за допомогою нейронних мереж, класифікація з використанням методу опорних векторів, класифікація за допомогою генетичного алгоритму, класифікація методом найближчого сусіда, логістична регресія.

Про кожен з методів було приведено короткі теоретичні відомості. Головну увагу було виділено методу нейронних мереж.

Нейронними мережами називається один з напрямків наукових досліджень штучного інтелекту. В його основі лежить імітація нервової системи людини. Головними властивостями, що беруться до уваги є самонавчання та здатність виправляти помилки.

Для розв’язання задачі класифікації було розглянуто 4 архітектури:

- багатошаровий перцептрон;
- мережі Кохонена;
- ймовірнісні нейронні мережі;
- згорткові нейронні мережі.

Для реалізації поставленої задачі було обрано архітектуру ймовірнісної нейронної мережі.

РОЗДІЛ 2.
ВИРІШЕННЯ ЗАДАЧІ КЛАСИФІКАЦІЙ ЗА ДОПОМОГОЮ ЙМОВІРНІСНИХ НЕЙРОННИХ МЕРЕЖ

2.1 Основні теоретичні відомості

Ймовірнісна нейронна мережа (PNN, ЙНМ) — це різновид нейронної мережі з прямим зв'язком, яка використовується для вирішення проблем класифікації та розпізнавання образів.

Цей тип ШНМ був створений з використанням байєсівської мережі та статистичного підходу, відомого як дискримінантний аналіз ядра Фішера.

PNN пропонують масштабовану альтернативу звичайним нейронним мережам із зворотним поширенням у задачах класифікації без необхідності проведення масивних прямих і зворотних обчислень, які пов'язані зі звичайними нейронними мережами. Крім того, вони можуть працювати з меншими наборами навчальних даних. Однак ця перевага може призвести до потреби великого обсягу пам'яті, оскільки навчальні дані збільшуються.

Ймовірнісні нейронні мережі широко застосовуються для розв'язання задачі класифікації та розпізнавання об'єктів. До задач, що розв'язуються за допомогою ЙНМ можна віднести наступні:

- діагностика різних захворювань: аритмія, гепатит, онкологія і т.д.;
- визначення особистості, на основі розпізнання голосу;
- класифікація зображень різного типу, а також їх обробка та розпізнавання;
- класифікаційні задачі у хімії, біології і т.д [22].

PNN показали багато перспектив у вирішенні складних наукових та інженерних завдань. Вони успішно вирішують всі види інженерних задач у різних областях з тих пір, як були запропоновані. У імовірнісній нейронній мережі розповсюдження має великий вплив на її продуктивність, і ймовірнісна

нейронна мережа буде генерувати погані результати прогнозу, якщо його вибрано неправильно. Вручну підібрати оптимальний варіант досить складно.

Нижче наведено основні типи труднощів, які можна вирішити за допомогою цього типу нейронних мереж:

- Класифікація шаблонів маркованих стаціонарних даних.
- Класифікація шаблонів даних, у якій дані мають змінну в часі імовірнісну функцію щільності.
- Програми для обробки сигналів, які працюють із сигналами як шаблонами даних.
- Неконтрольовані алгоритми для немаркованих наборів даних тощо.

2.2 Ймовірнісні штучні нейронні мережі в мові R

За базовою структурою Шпехта, ймовірнісна нейронна мережа складається з чотирьох шарів:

- вхідного;
- прихованого;
- підсумовуючого;
- вихідного.

Нижче наведено структурне зображення, а також детально описано кожен шар ЙНМ (рис.2.1) [23].

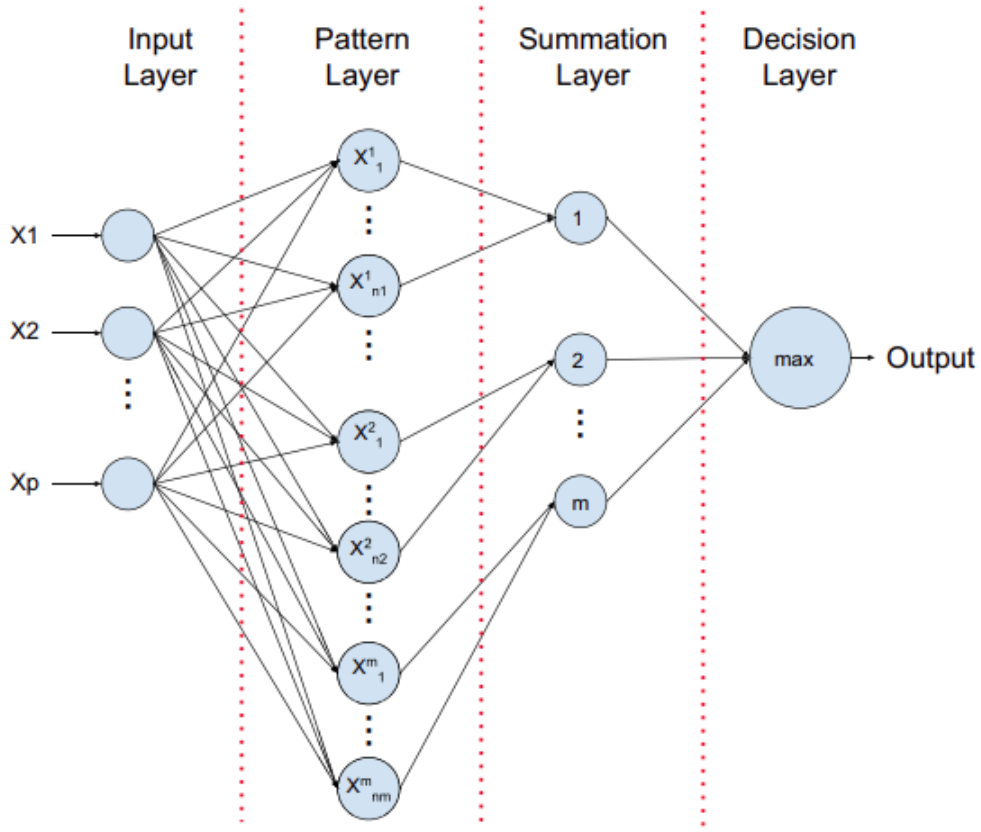

Рис. 2.1 – Структура ймовірнісної нейронної мережі

1. Вхідний шар

Кожна змінна предиктора представлена нейроном у вхідному шарі. Коли в категорійній змінній є N категорій, використовуються N-1 нейрон. Віднімаючи медіану та поділяючи на інтерквартильний діапазон, діапазон даних стандартизується. Потім вхідні нейрони передають значення кожному з нейронів прихованого шару.

2. Прихований шар

Кожен випадок у наборі навчальних даних має один нейрон у цьому шарі. Він зберігає значення змінних-провісників випадку, а також цільове значення. Прихований нейрон обчислює евклідову відстань між тестовим прикладом і центральною точкою нейрона, а потім використовує значення сигма для застосування функції ядра радіального базису.

3. Підсумовуючий шар

Кожна категорія цільової змінної має один нейрон шаблону в PNN. Кожен прихований нейрон зберігає фактичну цільову категорію кожної навчальної події; зважене значення, виведене прихованим нейроном, передається лише нейрону шаблону, який відповідає категорії прихованого нейрона. Значення для класу, який представляють нейрони шаблону, додаються.

4. Вихідний шар

Вихідний шар порівнює зважені голоси, накопичені в підсумовуючому шарі для кожної цільової категорії, і використовує найбільший голос для прогнозування цільової категорії.

2.3 Алгоритм навчання ЙНМ

На початку відомо вектори-зразки ознак навчальної вибірки. Також відомо класи, до яких належать кожен з них. Тоді, PNN налаштовується наступним чином.

1. Введення файлу, що містить зразки векторів і номери класів.
2. Сортуйте векторів на k наборів, кожен з яких містить один клас векторів.
3. Створення функції Гауса з центром кожного вектору-зразку в наборі k. Визначення кумулятивної вихідної функції Гауса для кожного k.
4. Після того, як PNN буде визначена, до неї подаються вектори та класифікуються за описаними нижче пунктами.

- Зчитування вхідного вектору і призначення функції Гауса відповідно до їхньої продуктивності в кожній категорії.
- Обчислення усіх функціональних значень Гауса у прихованих вузлах.
- Передача значень Гауса з кластера прихованих вузлів до єдиного вихідного вузла кластера.

- Для кожного вихідного вузла категорії додаються всі вхідні дані та множаться на константу.
- Визначення найцінніших з усіх значень, складених разом у вихідних вузлах.

Висновки до другого розділу

В даному розділі розглянуто основні теоретичні відомості відносно ймовірнісних нейронних мереж:

- Наведено характеристику ЙНМ;
- Приведено структурну схему;
- Описано алгоритм;
- Приведено основні переваги та недоліку цього типу штучних нейронних мереж.

Ймовірнісна нейронна мережа — це різновид нейронної мережі з прямим зв’язком, яка використовується для вирішення проблем класифікації та розпізнавання образів.

ЙНМ складається з чотирьох шарів: вхідного, прихованого, підсумовуючого, вихідного.

Ймовірнісні нейронні мережі можна використовувати для задач класифікації, розпізнавання. Вони гарно себе показують у вирішенні медичних, біологічних, хімічних та інженерних задач.

Коротко алгоритм навчання цього типу нейронних мереж можна записати наступним чином: коли представлено вхід, перший рівень обчислює відстані від вхідного вектора до навчальних вхідних векторів і створює вектор, елементи якого вказують, наскільки близький вхід до навчального входу. Другий рівень підсумовує ці внески для кожного класу вхідних даних, щоб отримати як його чистий вихід вектор ймовірностей. Вкінці, функція передачі конкуренції на виході другого рівня вибирає максимальну з цих ймовірностей і створює 1 для цього класу та 0 для інших класів.

Даний вид нейронних мереж має як свої переваги, так і недоліки.

До переваг відносять:

- квадрат PNN вимірюється набагато швидше, ніж багатошаровими персептронними мережами;
- PNN можуть бути набагато правильнішими, точнішими, ніж багатошарові персептронні мережі;
- квадратна міра мереж PNN порівняно нечутлива до викидів.
- мережі PNN генерують правильно-передбачувану ймовірність цілі;
- PNN наближаються до байєсоптимуму.

До недоліків відносяться:

- PNN працює повільніше в порівнянні з багатошаровим персептроном.
- PNN потребує багато областей пам'яті для зберігання.

РОЗДІЛ 3.
ПРАКТИЧНА РЕАЛІЗАЦІЯ

3.1 Опис початкових даних

На початку даного розділу приведемо статистичні дані, які будемо класифікувати. Спочатку, розглянемо взагалі поняття, пов’язані зі статистичними данними.

Статистичні дані - інформація, отримана на підставі проведених статистичних спостережень, яка була опрацьована і подана у формалізованому вигляді відповідно до загальноприйнятих принципів та методології. Можна сказати, статистичні дані — це сукупність об’єктів і ознак, що їх характеризують, де **об’єкти** — спостереження та випадки, а **ознаки** — змінні.

Змінними називаються величини, які в результаті вимірювання можуть приймати різні значення. Змінні можуть бути залежними або незалежними. **Незалежні змінні** - це змінні, значення яких в процесі експерименту можна змінювати. **Залежними змінними** називають такі змінні, значення яких можна тільки вимірювати.

Для реалізації поставленої мети, розв’язання задачі класифікації за допомогою нейромережевих методів, було обрано вибірку даних, яка містить у собі показники стану пацієнтів з гепатитом С на основі лабораторних досліджень.

Вибірка складається зі 155 записів, тобто в ній приведені дані 155 пацієнтів. Вона містить 20 змінних, з яких одна змінна є залежною, а 19 незалежних. Залежна змінна (Class), в даній вибірці — змінна, що показує стан пацієнта, а саме — «Пацієнт стабільний», «Пацієнт нестабільний» (0 чи 1). До незалежних речових змінних відносяться:

- вік (AGE) — відповідає віку пацієнта.

- стать (SEX) — приймає значення 1, якщо пацієнт є чоловіком, а 2 — якщо жінка.
- чи приймав пацієнт стероїди (STEROID) - приймає значення 1, якщо пацієнт не вживав стероїди, а 2 — якщо вживав.;
- чи приймав пацієнт антивірусні засоби (ANTIVIRALS) - приймає значення 1, якщо пацієнт не вживав антивірусні засоби, а 2 — якщо вживав;
- чи є ознаки нездужання (FATIGUE) - приймає значення 1, якщо пацієнт не мав ознак нездужання, а 2 — якщо мав;
- втома (MALAISE) - приймає значення 1, якщо пацієнт не мав ознак втоми, а 2 — якщо мав;
- анорексія (ANOREXIA) - приймає значення 1, якщо пацієнт не мав ознак анорексії, а 2 — якщо мав;
- збільшення печінки (LIVER BIG) - приймає значення 1, якщо пацієнт не мав ознак збільшення печінки, а 2 — якщо мав.
- ущільнення печінки (LIVER FIRM) - , приймає значення 1, якщо пацієнт не мав ознак ущільнення печінки , а 2 — якщо мав;
- чи має пацієнт судинні сітки (SPIDERS) - приймає значення 1, якщо у пацієнта немає судинної сітки, а 2 — якщо має;
- чи прощупується селезінка (SPLEEN PALPABLE) - приймає значення 1, якщо у пацієнта не прощупується селезінка, а 2 — якщо прощупується;
- чи є в пацієнта асцит (ASCITES) - приймає значення 1, якщо у пацієнта немає асциту, а 2 — якщо є;
- опис показників крові (білірубін) (BILIRUBIN) — показує рівень білірубіну;
- опис показників крові (альбумін) (ALBUMIN) — показує рівень альбуміну;

- опис показників крові (фермент АСТ) (SGOT) — показує рівень ферменту АСТ;
- опис показників крові (протромбін) (PROTIME) — незалежна змінна, показує рівень протромбіну;
- чи має пацієнт варикоз (VARICES) – приймає значення 1, якщо у пацієнта немає варикозу, а 2 — якщо є.
- чи робилась гістологія пацієнту (HISTOLOGY) - приймає значення 1, якщо у пацієнта не робили гістологію, а 2 — якщо робили.

3.2 Програмна реалізація комп'ютерної моделі

Для реалізації поставленої мети та задач було використано мову R та середовище розробки RStudio. Нижче приведено теоретичну інформацію відносно засобів реалізації моделі ЙНМ.

R — це безкоштовне програмне середовище для статистичних обчислень і графіки. Він компілюється та працює на різних платформах UNIX, Windows і MacOS.

R — це система аналізу, мова програмування й середовище для статистичних обчислень і графічного аналізу, яка була створена в 1996 році Россом Іхакою та Робертом Гентлеманом. Мова R сьогодні активно використовується по всьому світу як для навчальних цілей (в університетах, коледжах), так і для вирішення серйозних різноманітних задач аналізу даних. Вона є популярною та широко-використованою завдяки своїм особливостям, що є, безумовно, перевагами: R є безкоштовним програмним забезпеченням, в цій мові реалізовані всі засоби для аналізу даних, вона одночасно виступає як мовою програмування, так і програмним забезпеченням, це проста та ефективна мова, що дозволяє імпорт даних з різних джерел. Безумовною перевагою є те, що R надає доступ до великої колекції інструментальних засобів, що дозволяють проводити статистичний аналіз.

RStudio - це інтегроване середовище розробки (SDI) для мови програмування R, присвяченої статистичним обчисленням та графіці. Він включає консоль, редактор синтаксису, що підтримує виконання коду, а також інструменти для побудови графіків, налагодження та управління робочою областю.

RStudio доступний для Windows, Mac і Linux або для браузерів, підключених до RStudio Server або RStudio Server Pro (Debian / Ubuntu, RedHat/CentOS та SUSE Linux). RStudio має на меті забезпечити статистичне обчислювальне середовище R. Це дозволяє проводити аналіз та розробку для будь-якого аналізу даних за допомогою R [24].

Першим кроком, який необхідно виконати – завантажити вибірку даних. Завантаження текстового файлу з даними виконаємо за допомогою команди read.csv(“PATH”) (рис. 3.1).

```
hep1 <- read_csv("C:/hepatitis1.data.txt",
                 +     col_types = cols(AGE = col_integer(),
                                        +         SEX = col_character(), STEROID = col_integer(),
                                        +         FATIGUE = col_integer(), MALAISE = col_integer(),
                                        +         ANOREXIA = col_integer(), LIVER_BIG = col_integer(),
                                        +         LIVER_FIRM = col_integer(), SPLEEN_PALPABLE = col_integer
                                        +         SPIDERS = col_integer(), ASCITES = col_integer(),
                                        +         VARICES = col_integer(), BILIRUBIN = col_double(),
                                        +         ALK_PHOSPHATE = col_integer(), SGOT = col_integer(),
                                        +         ALBUMIN = col_double(), PROTIME = col_integer(),
                                        +         HISTOLOGY = col_integer()))
```

Рисунок 3.1 – Завантаження початкових даних до середовища RStudio

Після завантаження даних необхідно проаналізувати та зробити висновок, від яких саме факторів найбільше залежить результат. Цей процес було реалізовано за допомогою моделі «випадкового лісу».

Випадковий ліс (RF, random forest) – це безліч вирішальних дерев.

Випадкові ліси базуються на наступному принципі: сукупність результатів кількох предикторів дає кращий прогноз, ніж найкращий

індивідуальний предиктор. Група предикторів називається ансамблем. Таким чином, ця методика називається ансамблевим навчанням [25].

У задачі класифікації приймається рішення голосуванням у більшості.

Для процесу побудови моделей випадкового лісу необхідно використати пакет **randomForest**.

Нижче приведено отримані результати після побудови однієї з усіх можливих моделей (рис. 3.2).

```
>  print(rf)

Call:
 randomForest(formula = Class ~ BILIRUBIN + ASCITES + AGE + PROTIME +      SGOT + ALK_PHOSPHATE + VARIC
ES + MALAISE + ANOREXIA + FATIGUE +      STEROID + ANTIVIRALS + SEX + FATIGUE + LIVER_BIG + LIVER_FIRM
 +      SPIDERS + SPLEEN_PALPABLE + ALBUMIN + HISTOLOGY, data = hepatitis1.data,      mrtry = 9, ntree
 = 1000, na.action = "na.omit")
               Type of random forest: classification
                     Number of trees: 1000
No. of variables tried at each split: 4

        OOB estimate of  error rate: 10%
Confusion matrix:
  1  2 class.error
1 6  7  0.53846154
2 1 66  0.01492537
```

Рисунок 3.2 – Приклад одної з побудованих моделей

Модель «випадкового лісу» була побудована з наступними параметрами:

- mrty = 9 – кількість змінних-кандидатів, відібраних випадковим циклом побудови дерева.

- ntree = 1000 - кількість дерев для вирощування.

- na.action = “na.omit” - функція для визначення дії, яку потрібно виконати, якщо знайдено NA (відсутнє значення).

За допомогою моделі випадкового лісу було досліджено вплив різних змінних на результат. Графічно його можна уявити наступним чином (рис. 3.3):

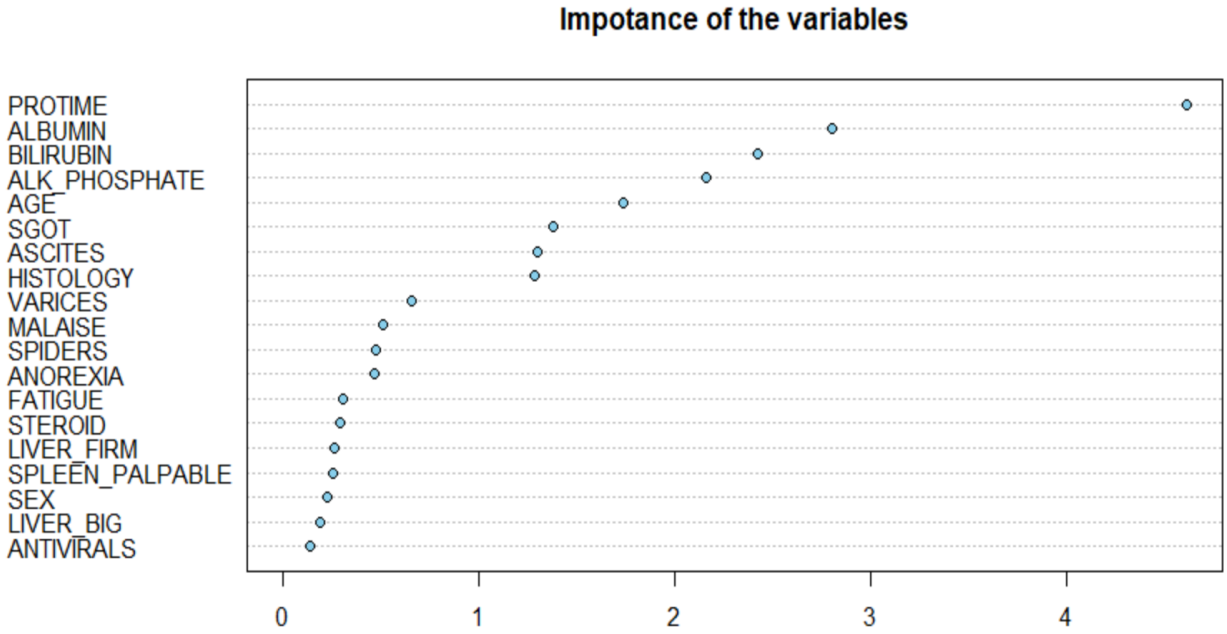

Рисунок 3.3 – Важливість змінних

За допомогою графіків зробимо аналіз кореляції найбільш впливаючих на результат змінних (рис. 3.4 - рис. 3.6).

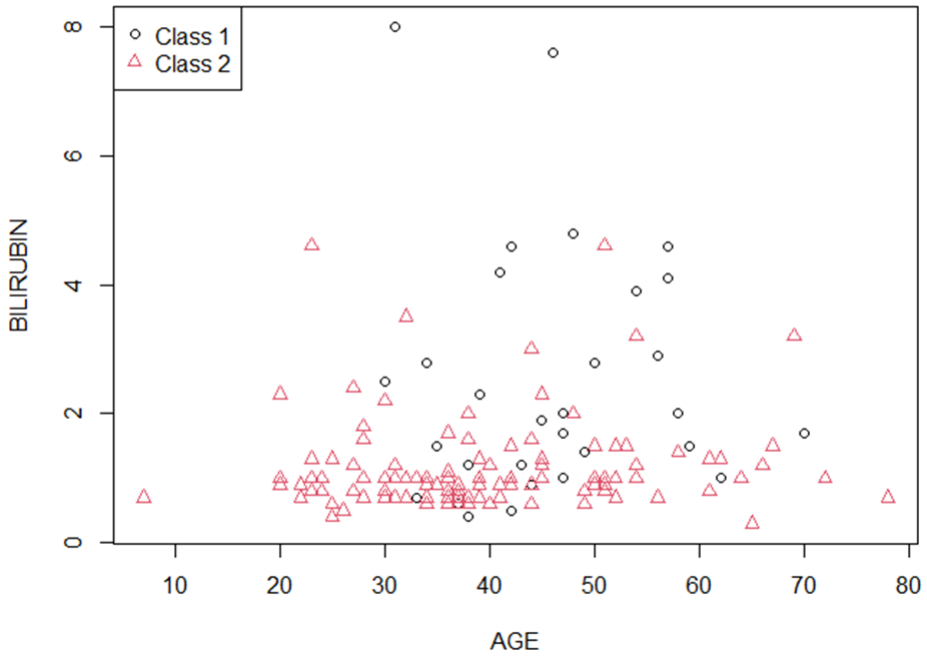

Рисунок 3.4 - Зв’язок BILIRUBIN-AGE

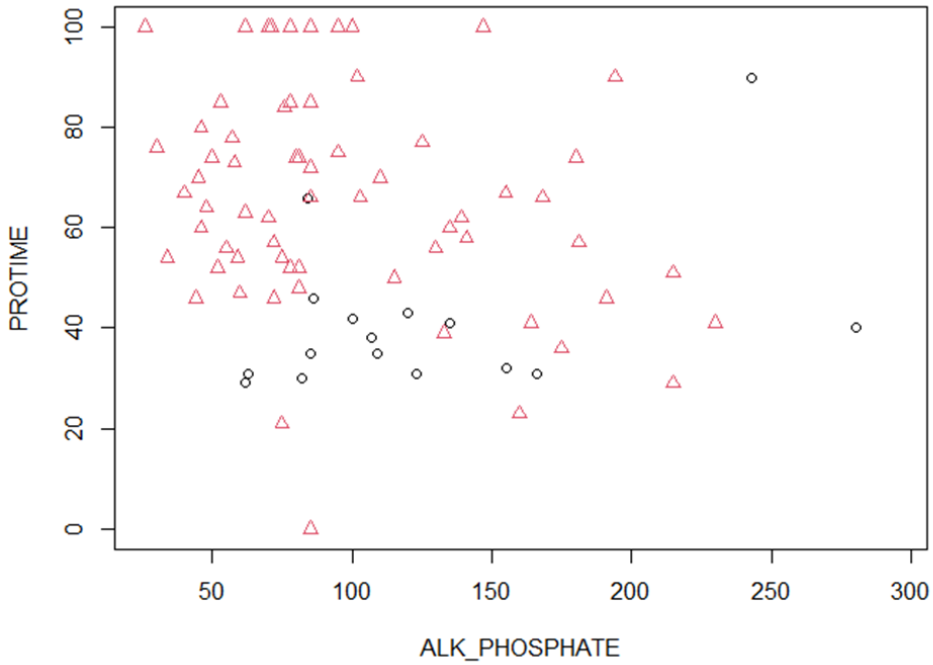

Рисунок 3.5 - Зв’язок PROTIME-ALK_PHOSPHATE

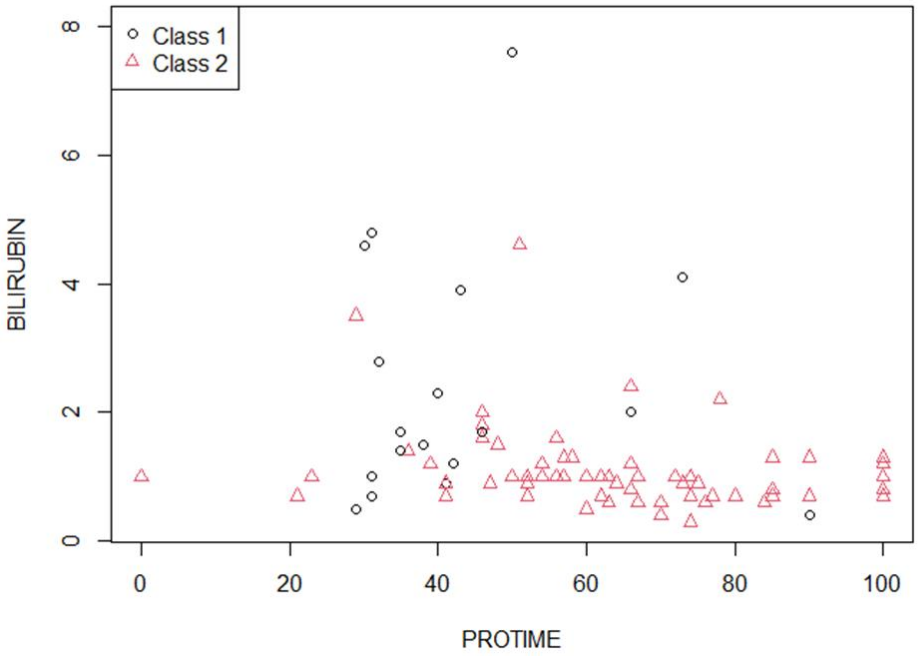

Рисунок 3.6 - Зв’язок BILIRUBIN – PROTIME

Для реалізації ймовірнісних штучних мереж було використано вбудований пакет **pnn**.

Пакет PNN реалізує алгоритм, запропонований Specht (1990) [26]. Він написаний статистичною мовою R. Він вирішує поширену проблему автоматичного навчання. Знання набору спостережень описується вектором кількісних змінних, класифікуємо їх у задану кількість груп. Тоді, алгоритм навчається з цими наборами даних і повинен потім вгадати групу будь-якого нового спостереження. Ця нейронна мережа має головну перевагу, щоб почати узагальнення миттєво з невеликим набором відомих спостережень.

Пакет PNN експортує чотири функції. Ці функції задокументовані прикладами та забезпечені модульними тестами.

Після завантаження даних за допомогою функції **learn** з пакету **pnn** було побудовано **ймовірнісну нейронну мережу.** На малюнку нижче приведено отримані характеристики при побудові нейронної мережі (рис.3.7).

```
> pnn<- learn(hepatitis1.data,category.column = 1)
> pnn$model
[1] "Probabilistic neural network"
> pnn$category.column
[1] 1
> pnn$categories
[1] "0" "1"
> pnn$k
[1] 19
> pnn$n
[1] 155
```

Рисунок 3.7 – Характеристики побудованої моделі

Як можна бачити, модель дійсно побудована за допомогою ймовірнісної нейронної мережі. З навчальної вибірки саме перший стовпчик є категоріальним (використовується для класифікації). Програма повинна

ділити дані на два класи – 0 та 1. Кількість незалежних змінних дорівнює 19. Загальна кількість записів дорівнює 155.

За допомогою функції **smooth** з пакету **pnn** було встановлено параметр згладжування. За допомогою функції **perf** виконано прогноз (рис.3.8).

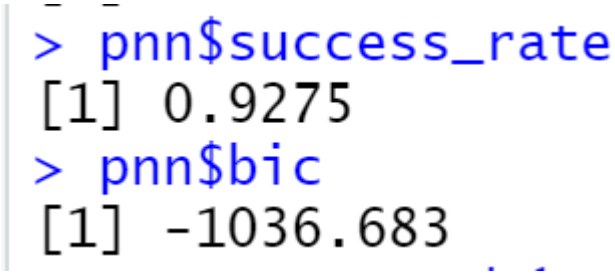

Рисунок 3.8 – Отримані розрахунки після побудови моделі нейронної мережі

success_rate - показник успішності за всіма спостереженнями в навчальному наборі.

bic - це адаптована версія байєсівського інформаційного критерію, яка допомагає порівнювати різні версії імовірнісних нейронних мереж. Критерій вибору статистичної моделі з деякого кінцевого набору. Перевага надається моделі з мінімальним значенням критерію.

Приведемо таблицю залежності показника успішності та інформаційного критерію від значення сигма (Таблиця 3.1).

Таблиця 3.1

Залежність показника успішності та інформаційного критерію від значення сигма

Значення сигма	success_rate	bic
0,6	0.925	-1023.123
0,7	0.9275	-1036.683
0,9	0.9225	-1010.007

Приведемо графіки, які найкраще демонструють отримані результати (рис.3.9).

Рис. 3.9 – Графіки залежностей показника успішності та інформаційного критерію від значення сигма

Як можна бачити з отриманих графіків, найкраще значення за показником успішності має модель, при побудові якої значення сигма

дорівнювало 0.7. Найкраще значення за інформаційним критерієм має модель, при побудові якої значення сигма також дорівнювало 0.7. Тобто, можна зробити висновок, що для побудови найбільш якісної моделі необхідно було взяти показник сигма = 0,7.

Висновки до третього розділу

В даному розділі наведено практичну реалізацію моделі класифікації станів за допомогою ймовірнісних штучних нейронних мереж. Якщо казати детальніше:

- зроблено огляд використовуваного ПЗ для вирішення задачі;
- наведено опис початкових даних;
- розглянуто пакети, необхідні для реалізації моделі;
- проаналізовано вплив різних змінних на результат, за допомогою моделі «штучного лісу»;
- побудовано моделі класифікації;
- за інформаційним показником та показником успішності за всіма критеріями обрано найкращу.

Вибірка складається зі 155 записів, тобто в ній приведені дані 155 пацієнтів. Вона містить 20 змінних, з яких одна змінна є залежною, а 19 незалежних. Залежна змінна (Class), в даній вибірці — змінна, що показує стан пацієнта, а саме — «Пацієнт стабільний», «Пацієнт нестабільний» (0 чи 1). До незалежних речових змінних відносяться: AGE, SEX, STEROID, ANTIVIRALS, FATIGUE,MALAISE, ANOREXIA, LIVER BIG, LIVER FIRM, SPIDERS, SPLEEN PALPABLE, ASCITES, BILIRUBIN, ALBUMIN, SGOT, PROTIME,VARICES, HISTOLOGY

З отриманих результатів видно, що процес класифікації пройшов відмінно, показник успішності – 0,9275, що говорить про високу якість класифікації.

ВИСНОВКИ

В даній кваліфікаційній роботі було розв’язано наступні задачі:

- формулювання задачі класифікації ознак в медико-біологічних системах,
- аналіз існуючих методів класифікації об’єктів,
- огляд обраного методу для вирішення задачі,
- складання математичної моделі вирішення задачі класифікації за допомогою ймовірнісних штучних мереж,
- розробка програмно-алгоритмічної моделі системи класифікації,
- виконання тестування моделі та аналіз отриманих результатів.

Якщо говорити більш детально, то:

1. В роботі приведено основні теоретичні матеріали відносно поняття «класифікація». Розглянуто методи, за допомогою яких можна реалізувати задачу класифікації. Наведемо стисло з’ясовані дані.

Класифікація — один з розділів машинного навчання, присвячений вирішення наступного завдання: є безліч об'єктів, розділених деяким чином на класи. Вона має численні застосування у різних сферах: розпізнавання рукописного тексту, класифікація даних, класифікація зображень, прогнозування банкрутства, розпізнавання мови, медичне діагностування, виявлення несправностей.

Задачу класифікації можна реалізувати за допомогою наступних методів: байєсовий класифікатор, класифікація за допомогою дерева рішень, класифікація за допомогою нейронних мереж, класифікація з використанням методу опорних векторів, класифікація за допомогою генетичного алгоритму, класифікація методом найближчого сусіда, логістична регресія. Кожен з методів був частково розглянутий. Основна увага приділялася нейронним мережам.

Нейронними мережами називається один з напрямків наукових досліджень штучного інтелекту. В його основі лежить імітація нервової системи людини. Головними властивостями, що беруться до уваги є самонавчання та здатність виправляти помилки.

Для розв'язання задачі класифікації було розглянуто 4 архітектури: багатошаровий перцептрон; мережі Кохонена; ймовірнісні нейронні мережі; згорткові нейронні мережі.

Для реалізації поставленої задачі було обрано архітектуру ймовірнісної нейронної мережі. Теоретичний опис архітектури містить: характеристику ЙНМ; структурну схему, алгоритм; переваги та недоліки.

Ймовірнісна нейронна мережа — це різновид нейронної мережі з прямим зв'язком, яка використовується для вирішення проблем класифікації та розпізнавання образів. Вона складається з чотирьох шарів: вхідного, прихованого, підсумовуючого, вихідного.

Коротко алгоритм навчання цього типу нейронних мереж можна записати наступним чином: коли представлено вхід, перший рівень обчислює відстані від вхідного вектора до навчальних вхідних векторів і створює вектор, елементи якого вказують, наскільки близький вхід до навчального входу. Другий рівень підсумовує ці внески для кожного класу вхідних даних, щоб отримати як його чистий вихід вектор ймовірностей. Вкінці, функція передачі конкуренції на виході другого рівня вибирає максимальну з цих ймовірностей і створює 1 для цього класу та 0 для інших класів.

Даний вид нейронних мереж має як свої переваги, так і недоліки. Вибірка складається зі 155 записів, тобто в ній приведені дані 155 пацієнтів. Вона містить 20 змінних, з яких одна змінна є залежною, а 19 незалежних.

Для процесу побудови моделей випадкового лісу необхідно використати пакет randomForest. За допомогою функції smooth з пакету pnn було встановлено параметр згладжування. За допомогою функції perf виконано прогноз.

З отриманих результатів видно, що процес класифікації пройшов відмінно, показник успішності – 0,9275, що говорить про високу якість класифікації.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Машинне навчання [Електронний ресурс] Режим доступа: https://www.it.ua/knowledge-base/technology-innovation/machine-learning. Дата звернення: 06.03.2021.

3. Основы линейной регрессии [Електронний ресурс] Режим доступа: http://statistica.ru/theory/osnovy-lineynoy-regressii/. Дата звернення: 15.04.2020.

4. Классификация [Електронний ресурс] Режим доступа: http://www.machinelearning.ru/wiki/index.php?title=%D0%9A%D0%BB%D0%B0%D1%81%D1%81%D0%B8%D1%84%D0%B8%D0%BA%D0%B0%D1%86%D0%B8%D1%8F. Дата звернення: 15.04.2020

5. Machine Learning Methods in Medicine Diagnostics Problem [Електронний ресурс] / [V. Strilets, N. Bakumenko, V. Donets та ін.] // Proceedings of the 16th International Conference on ICT in Education, Research and Industrial Applications. Integration, Harmonization and Knowledge Transfer. Volume II: Workshops, Kharkiv, Ukraine, October 06-10, 2020, pp. 89-101. – [Електронний ресурс] Режим доступа: http://ceur-ws.org/Vol-2732/20200089.pdf.

6. Korb K. Bayesian Artificial Intelligence / K. Korb, A. Nicholson. – London: CRC Press, 2011. – 452 с.

7. Rish I. An empirical study of the naive Bayes classifier [Електронний ресурс] / Irina Rish // IJCAI Workshop on Empirical Methods in AI. – 2001. – Режим доступу до ресурсу: https://www.researchgate.net/publication/228845263_An_Empirical_Study_of_the_Naive_Bayes_Classifier.

8. Линейный классификатор [Електронний ресурс] – Режим доступу до ресурсу: http://www.machinelearning.ru/wiki/index.php?title=%D0%9B%D0%B8%D0%B

D%D0%B5%D0%B9%D0%BD%D1%8B%D0%B9_%D0%BA%D0%BB%D0%B0%D1%81%D1%81%D0%B8%D1%84%D0%B8%D0%BA%D0%B0%D1%82%D0%BE%D1%80.

9. Вапник В. Теория распознавания образов (статистические проблемы обучения) / В. Вапник, А. Червоненкис. – Москва: Наука, 1974. – 416 с.

10. Vapnik V. Statistical Learning Theory / Vladimir N. Vapnik. – New York: Wiley, 1998. – 732 с.

11. Демидова Л. Классификация больших данных: использование SVM-ансамблей и SVM-классификаторов с модифицированным роевым алгоритмом / Л. Демидова, Е. Никульчев, Ю. Соколова. // Cloud of Science.. – 2016. – С. 5–42.5. Artificial neural networks для решения бизнес задач [Електронний ресурс] Режим доступа: https://evergreens.com.ua/ru/development-services/neural-network.html Дата звернення: 17.04.2021

12. Золин А. Применение нейронных сетей в медицине / А. Золин, А. Силаева. // Сборник «Актуальные проблемы науки, экономики и образования XXI века». – 2012. – С. 264–271.

13. Maad M. Mijwil. Artificial Neural Networks Advantages and Disadvantages [Електронний ресурс] / Maad M. Mijwil Maad. – 2018. – Режим доступу до ресурсу: https://www.linkedin.com/pulse/artificial-neural-networks-advantages-disadvantages-maad-m-mijwel/.

14. MTH594 Advanced data mining: theory and applications [Електронний ресурс] Режим доступу: https://github.com/diefimov/MTH594_MachineLearning. Дата звернення: 17.09.2021

15. Методологія наукових досліджень в медицині : навчальний посібник / В. Д. Бабаджан, Н. С. Бакуменко, О. І. Кадикова [та ін.] ; за ред. П. Г. Кравчуна, В. Д. Бабаджана, В. В. М'ясоєдова. – Харків : ХНМУ, 2020. – 260 с.

16. Machine Learning Methods in Medicine Diagnostics Problem [Електронний ресурс] / [V. Strilets, N. Bakumenko, V. Donets та ін.] //

Proceedings of the 16th International Conference on ICT in Education, Research and Industrial Applications. Integration, Harmonization and Knowledge Transfer. Volume II: Workshops, Kharkiv, Ukraine, October 06-10, 2020, pp. 89-101. – [Електронний ресурс] Режим доступу: http://ceur-ws.org/Vol-2732/20200089.pdf. Дата звернення: 17.09.2021

17. Нейросети для чайников [Електронний ресурс] Режим доступу: https://stevsky.ru/kompiuteri/iskusstvennie-neyronnie-seti-ins-chto-takoe-neyroseti-kak-oni-rabotaiut-preimuschestva-i-nedostatki-iskusstvennich-neyronov-gde-ispolzuiutsya-neyroseti

18. Перцептрон [Електронний ресурс] Режим доступу: https://znaimo.com.ua/%D0%9F%D0%B5%D1%80%D1%86%D0%B5%D0%BF%D1%82%D1%80%D0%BE%D0%BD

19. Мережі Кохонена [Електронний ресурс] Режим доступу: http://inmad.vntu.edu.ua/portal/static/D28C6207-9C44-4C8C-B56E-43E2CF72C372.pdf Дата звернення: 05.10.2021

20. ЗГОРТКОВІ НЕЙРОННІ МЕРЕЖІ [Електронний ресурс] Режим доступу: https://conf.ztu.edu.ua/wp-content/uploads/2019/02/45-1.pdf Дата звернення 05.10.2021

21. Глибокі Нейронні Мережі для Вирішення Завдань Розпізнавання і Класифікації Зображення [Електронний ресурс] Режим доступу: http://itcm.comp-sc.if.ua/2017/Sineglazov.pdf Дата звернення 15.10.2021

22. Использование вероятностных нейронных сетей для предсказания локализации белков в клеточных компартментах Назин П.С.*1,2, Готовцев П.М.1 1НИЦ "Курчатовский институт", Москва, Россия, Московский физико-технический институт, Москва, Россия [Електронний ресурс] Режим доступу: https://www.matbio.org/2019/Nazin_14_220.pdf#:~:text=%D0%92%D0%B5%D1%80%D0%BE%D1%8F%D1%82%D0%BD%D0%BE%D1%81%D1%82%D0%BD%D0%B0%D1%8F%20%D0%BD%D0%B5%D0%B9%D1%80%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F%20%D1%81%D0%B5%D1%82%D1%8C

%20%D1%81%D0%BE%D1%81%D1%82%D0%BE%D0%B8%D1%82%20%D0%B8%D0%B7,%D0%BF%D1%80%D0%BE%D0%B2%D0%BE%D0%B4%D0%B8%D1%82%D1%81%D1%8F%20%D1%81%D1%80%D0%B0%D0%B2%D0%BD%D0%B5%D0%BD%D0%B8%D0%B5%20%D0%B2%D0%B5%D1%80%D0%BE%D1%8F%D1%82%D0%BD%D0%BE%D1%81%D1%82%D0%B5%D0%B9%20%D0%B8%20%D0%B2%D1%8B%D0%B2%D0%BE%D0%B4%D0%B8%D1%82%D1%81%D1%8F Дата звернення 21.10.2021

23. Структура вероятностных нейронных сетей [Електронний ресурс] Режим доступу: https://www.sciencedirect.com/science/article/pii/B978012816514000014X Дата звернення 03.11.2021

24. R language [Електронний ресурс] Режим доступу: https://www.r-project.org/ Дата звернення 03.11.2020

25. R Случайный Лес Учебник [Електронний ресурс] Режим доступу: https://coderlessons.com/tutorials/mashinnoe-obuchenie/r-programmirovanie/29-r-sluchainyi-les-uchebnik Дата звернення 03.10.2021

26. Donald Specht. (1990). Probabilistic Neural Networks. [Електронний ресурс] Режим доступу: https://www.wi.hs-wismar.de/~cleve/vorl/projects/dm/ss13/PNN/Quellen/Specht_ProbabilisticNeuralNetworks.pdf Дата звернення 28.11.2020

ДОДАТКИ

Додаток А

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Харківський національний університет імені В. Н. Каразіна

Факультет комп'ютерних наук
Кафедра теоретичної та прикладної системотехніки
Рівень вищої освіти (освітньо-кваліфікаційний рівень) Магістр
Галузь знань: 15 – Автоматизація та приладобудування
Спеціальність: 151 – Автоматизація та комп'ютерно-інтегровані технології

ЗАТВЕРДЖУЮ
Завідувач кафедри теоретичної
та прикладної системотехніки
_____д.т.н., проф. Шматков С. І.

«____» _____20___року

З А В Д А Н Н Я
НА КВАЛІФІКАЦІЙНУ РОБОТУ

Максимук Анастасії Родіонівни
(прізвище, ім'я, по батькові студента)

1. Тема роботи **«Модель інформаційної системи класифікації пацієнтів за допомогою ймовірнісних штучних мереж»**
керівник роботи **Бакуменко Ніна Станіславівна, доцент**
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від "___"_____20__року
№_____

2. Строк подання студентом роботи_____

3. Перелік питань, які потрібно розробити:
1. Постановка задачі класифікації ознак в медико-біологічних системах.
2. Детальний огляд обраного методу для вирішення задачі.
3. Розробка математичної моделі вирішення задачі класифікації за допомогою ймовірнісних штучних мереж.

4. Розробка програмно-алгоритмічної моделі класифікації системи.
5. Тестування моделі та аналіз результатів роботи.

4. План роботи

№ з/п	Назви етапів роботи	Термін виконання етапів роботи
1.	Підбір наукової літератури.	15.12.2020 – 10.02.2021
2.	Збір теоретичного матеріалу щодо ймовірнісних штучних мереж, відносно вирішення задачі класифікації.	25.03.2021 – 30.04.2021
3.	Розробка комп'ютерної моделі класифікації станів медико-біологічної системи за допомогою ймовірнісних штучних мереж.	01.05.2021 – 18.06.2021
4.	Відладка та тестування розробленої комп'ютерної моделі.	01.09.2021 – 03.10.2021
5.	Корегування моделі після тестування.	10.10.2021 – 15.11.2021
6.	Представлення дипломного проекту керівнику дипломної роботи та рецензенту.	16.11.2021– 30.11.2021

5. Дата видачі завдання_____

Студент Максимук А. Р. _____
підпис

Керівник роботи Бакуменко Н.С. _____
підпис

Додаток Б

Затверджую

_____

«_____»_____2021 р.

Технічне завдання
на розробку програмного виробу
«Модель інформаційної системи класифікації пацієнтів за допомогою ймовірнісних штучних мереж»

Назва розділу	Назва і зміст підрозділу
1. Введення	1.1 Назва: Модель інформаційної системи класифікації пацієнтів за допомогою ймовірнісних штучних мереж 1.2 Область застосування: медична діагностика
2. Підстава для розробки	2.1 Навчальний план ФКН за спеціальністю 151 – Автоматизація комп'ютерно-інтегровані технології. 2.2 Завдання на кваліфікаційну роботу, затверджене наказом № 0210-05/1804 від 10.09.2021р привести в Додатку А.
3. Призначення розробки	3.1. Мета розробки програмного виробу є подальше використання цього виробу у медико-біологічній сфері для покращення та полегшення процесу медичного діагностування. 3.2. Призначення програмного вироби: для використання в медичній діагностиці. 3.3. Вихідні дані для розробки: вірно класифіковані за спільними симптомами пацієнти.

4. Технічні вимоги до програмного виробу	4.1. Вимоги до функціональних характеристик: Програма повинна: 1) Представляти з себе комп'ютерну реалізацію моделі класифікації станів медико-біологічних систем; 2) Забезпечити можливість загрузки користувачем нових даних, для процесу аналізу; 3) Забезпечувати можливість виведення результату на екран за допомогою інтуїтивно зрозумілого інтерфейсу; 4) Забезпечити можливість навчання системи за допомогою штучних нейронних мереж. 4.2. Вимоги до надійності: 1) Програма повинна видавати повідомлення про помилки. 4.3. Вимоги до умов експлуатації: 1) Умови експлуатації співпадають з умовами експлуатації персональної електронно-обчислювальної машини, на якій буде працювати, а також сумісних з нею персональними комп'ютерами. 2) Для користування програмою користувачу необхідно пройти короткий курс навчання роботі з програмою. 4.4. Вимоги до складу і параметрів технічних засобів: 1) Персональний комп'ютер у повній комплектації або ноутбук. 4.5. Вимоги до інформаційної та програмної сумісності: «Windows» будь-яка версія, залежно від версії RStudio, «Linux» (будь-який дистрибутив);

<table>
<tr><td></td><td colspan="2">6) Вимоги до маркування та упаковки (не висуваються);
7) Вимоги до транспортування і зберігання (не висуваються);
8) Спеціальні вимоги (не пред’являються).</td></tr>
<tr><td>5. Вимоги до програмної документації.</td><td colspan="2">Програмною документацією до виробу «Модель інформаційної системи класифікації пацієнтів за допомогою ймовірнісних штучних мереж» вважати:
1) Дане Технічне завдання на розробку програмного виробу (представити у вигляді Додатку Б до пояснювальної записки до кваліфікаційної роботи).
2) Програму і методику випробувань розробленого програмного виробу (представити у вигляді Додатку В до пояснювальної записки до кваліфікаційної роботи).
3) Опис програмного виробу (представити в розділі 3 пояснювальної записки до кваліфікаційної роботи).
4) Лістинг програмного коду (представити у Додатку Г).</td></tr>
<tr><td>6. Техніко-економічні показники</td><td colspan="2">6.1 Визначення економічних переваг методу у порівнянні з вітчизняними та зарубіжними аналогами: виконати в розділі 2 пояснювальної записки до кваліфікаційної роботи
6.2 Оцінка економічної ефективності – непотрібна.</td></tr>
<tr><td rowspan="2">7. Стадії і етапи розробки</td><td>Дата</td><td>Назва етапу</td></tr>
<tr><td>15.04.21- 05.05.21
16.06.21-16.07.21

17.07.21-17.09.21</td><td>Підбір наукової літератури.
Збір теоретичного матеріалу щодо ймовірнісних штучних мереж, відносно вирішення задачі класифікації.
Розробка комп’ютерної моделі класифікації станів медико-біологічної системи за допомогою ймовірнісних штучних мереж.</td></tr>
</table>

	17.09.21-17.10.21 17.10.21-30.10.21 01.11.21-15.11.21	Відладка та тестування розробленої комп’ютерної моделі. Корегування моделі після тестування. Представлення дипломного проекту керівнику дипломної роботи та рецензенту.
8. Порядок контролю і приймання	8.1 Перевірку ходу розробки програмного виробу Керівнику робіт виконувати 1 раз в 3 тижні. 8.2 Випробування програмного вироби відповідно до Програми і методики випробувань провести на базі комп'ютерного класу. 8.3 Захист розробленого програмного виробу провести на засіданні атестаційної комісії. 8.4 Пояснювальну записку представити на паперових носіях в одному примірнику, в електронному вигляді - на CD-диску в одному екземплярі.	

Виконавець
студентка групи КУ-61
Максимук А.Р.
_____

Замовник
к. т. н., доцент, доцент кафедри ТПС
Бакуменко Н.С.
_____

Додаток В

Програма і методика випробувань
програмного виробу
«Модель інформаційної системи класифікації пацієнтів за допомогою ймовірнісних штучних мереж»

1 Об'єкт випробувань

1.1 Найменування випробуваного програмного вироби: «Модель інформаційної системи класифікації пацієнтів за допомогою ймовірнісних штучних мереж»

1.2 Область його застосування: медична діагностика.

1.3 Умовне позначення розробки (при необхідності).

2. Мета випробувань

Підтвердження коректності функціонування програмного виробу.

3. Загальні положення

3.1 Підстави для проведення випробувань

Підставою для проведення випробувань є наказ про призначення атестаційної комісії.

3.2 Місце і тривалість випробувань

Приймальні (приймально-здавальні) випробування проводяться дистанційно в період роботи атестаційної комісії».

3.3 Обсяг випробувань

Приймальні випробування програмного вироби проводяться в обсязі відповідному цієї Програми і методики випробувань.

3.4 Організації, які беруть участь у випробуваннях

Приймальні випробування проводяться атестаційною комісією напередодні засідання за участю Замовника, Виконавця та інших осіб, присутніх на засіданні в дистанційному режимі.

4. Вимоги до програми або програмного виробу

Модель повинна:

1) Представляти з себе комп’ютерну реалізацію моделі класифікації станів комп’ютеризованої системи;

2) Забезпечити можливість загрузки користувачем нових даних, для процесу аналізу;

3) Забезпечувати можливість виведення результату на екран за допомогою інтуїтивно зрозумілого інтерфейсу;

4) Забезпечити можливість навчання системи за допомогою нейронних мереж.

4.2. Вимоги до надійності:

1) Програма повинна видавати повідомлення про помилки.

4.3. Вимоги до умов експлуатації:

1) Умови експлуатації співпадають з умовами експлуатації персональної електронно-обчислювальної машини, на якій буде працювати, а також сумісних з нею персональними комп'ютерами.

2) Для користування програмою користувачу необхідно пройти короткий курс навчання роботі з програмою.

4.4. Вимоги до складу і параметрів технічних засобів:

1) Персональний комп'ютер у повній комплектації або ноутбук.

4.5. Вимоги до інформаційної та програмної сумісності:

«Windows» будь-яка версія, залежно від версії RStudio, «Linux» (будь-який дистрибутив), R, RStudio.

6) Вимоги до маркування та упаковки (не висуваються);

7) Вимоги до транспортування і зберігання (не висуваються);

8) Спеціальні вимоги (не пред'являються).

5. Вимоги до програмної документації

Склад програмної документації, що подається на випробування, включає:

1) Технічне завдання на розробку програмного вироби (представлено в Додатку Б до пояснювальної записки до дипломної роботи).

2) Ця Програма і методика випробувань розробленого програмного виробу (представлена в Додатку В до пояснювальної записки до дипломної роботи).

3) Опис програмного виробу (представлено в розділі 3 пояснювальної записки до дипломної роботи).

4) Лістинг програми (представлений в Додатку Г до пояснювальної записки до дипломної роботи) .

6. Засоби і порядок випробувань

6.1 Засоби випробувань

Випробування проводяться на технічних засобах, таких як персональний комп'ютер у повній комплектації або ноутбук.

Випробування проводяться з використанням програмних засобів, таких як операційна система «Windows» будь-яка версія, залежно від версії RStudio, «Linux» (будь-який дистрибутив), R, RStudio.

6.2 Порядок проведення випробувань

1) Перевірка програмної документації

1.1. Перевірка складу програмної документації. Перевірку здійснювати за критерієм наявності, представленої в ТЗ документації.

1.2. Критерієм успішності теста вважати відповідність наявної документації згідно зі списком в ТЗ.

1.3. Перевірка якості програмної документації. Перевірку здійснювати за критерієм відповідності вимогам ЕСПД.

1.4. Критерієм успішності теста вважати відповідність якості наявної документації згідно з вимогами ЕСПД.

1.5. Модель працює відповідно до умов експлуатації ОС MS Windows будь-якої версії, залежно від версії RStudio, а також сумісних з ним.

2) Для роботи необхідні встановлені програмні засоби R, RStudio .

Тест 1. Перевірка сумісності моделі з ОС та програмним забезпеченням

Критерієм успішності вважати сумісність моделі з ОС, R, RStudio.

Порядок проведення випробувань:

- Запуск програми здійснюється завантаженням в середовище RStudio програмного коду, з підключенням початкових даних.

- Завантажуються додаткові пакети, необхідні для роботи.

- Порядково запускається програмний код, для відстежування роботи програми на кожному кроці.

Висновки: після проведення тесту встановлено, що створена модель успішно функціонує на ОС, на якій тестувалася, завантажується до програмного середовища, проходить процес відладки.

Тест 2. Відповідність створеної моделі до поставленої мети

Для проведення випробувань пропонується провести ті заходи, опис яких містяться в розділі 3. Критерієм успішності є коректні отримані результати на виході.

```
> View(hepatitis1.data)
> pnn <- learn(hepatitis1.data,category.column = 1)
> pnn$model
[1] "Probabilistic neural network"
> pnn$set[1:10,]
   Class AGE SEX STEROID ANTIVIRALS FATIGUE MALAISE ANOREXIA
1      2  30   2       1          2       2       2        2
2      2  50   1       1          2       1       2        2
3      2  78   1       2          2       1       2        2
4      2  31   1      NA          1       2       2        2
5      2  34   1       2          2       2       2        2
6      2  34   1       2          2       2       2        2
7      1  51   1       1          2       1       2        1
8      2  23   1       2          2       2       2        2
9      2  39   1       2          2       1       2        2
10     2  30   1       2          2       2       2        2
```

Рисунок В.1 — Результат виконання Теста 2

Висновок: в результаті даного тесту було встановлено, що побудована модель повністю відповідає поставленій меті. Процес класифікації проходить успішно, результати вказують на відмінний результат.

Висновки: тести 1 та 2 пройдені успішно, отримані результати є коректними, процес класифікації відмінним.

Виконавець Максимук А.Р.

Додаток Г

Лістинг програми

Перевірка важливості змінних за допомогою моделі «випадковий ліс»

```
     library(readr)
      hep1 <- read_csv("C:/hepatitis1.data.txt", +      col_types
= cols(AGE = col_integer(),
                                                 +             SEX =
col_character(), STEROID = col_integer(),
                                                 +         FATIGUE =
col_integer(), MALAISE = col_integer(),
                                                 +          ANOREXIA
= col_integer(), LIVER_BIG = col_integer(),
                                                 +        LIVER_FIRM
= col_integer(), SPLEEN_PALPABLE = col_integer(),
                                                 +         SPIDERS =
col_integer(), ASCITES = col_integer(),
                                                 +         VARICES =
col_integer(), BILIRUBIN = col_double(),
                                                 +
ALK_PHOSPHATE = col_integer(), SGOT = col_integer(),
                                                 +         ALBUMIN =
col_double(), PROTIME = col_integer(),
                                                 +         HISTOLOGY
= col_integer()))



     library("randomForest")
      library("caret")
      str(hepatitis1.data)
      hepatitis1.data$Class<-as.factor(hepatitis1.data$Class)
      set.seed(123)
      rf<-
randomForest(Class~BILIRUBIN+ASCITES+AGE+PROTIME+SGOT+ALK_PHOSPH
ATE+VARICES+MALAISE+ANOREXIA+FATIGUE+STEROID+ANTIVIRALS+SEX+FATI
GUE+LIVER_BIG+LIVER_FIRM+SPIDERS+SPLEEN_PALPABLE+ALBUMIN+HISTOLO
GY,data=hepatitis1.data,na.action="na.omit",mrtry=9,ntree=1000)
      print(rf)
      varImpPlot(rf,main="Impotance   of   the   variables",bg   =
"skyblue")

      hepatitis1.data$Class<-as.integer(hepatitis1.data$Class)


      plot(hepatitis1.data$ALK_PHOSPHATE,
hepatitis1.data$PROTIME,               col=hepatitis1.data$Class,
pch=hepatitis1.data$Class,xlab ="ALK_PHOSPHATE",ylab="PROTIME")
```

```
      legend("topleft", legend = c("Class 1","Class 2"), col =
c(1,2),pch =c(1,2))
```

Реалізація ймовірнісної нейронної мережі

```
     library("caret")
     library("pnn")
     hepatitis1.data    <-    read.csv("C:/hepatitis1.data.txt",
na.strings="?")
     View(hepatitis1.data)
     pnn<- learn(hepatitis1.data,category.column = 1)
     pnn$model
     pnn$set[1:10,]
     pnn$category.column
     pnn$categories
     pnn$k
     pnn$n
     pnn<-smooth(pnn, sigma = 0.8)
     pnn<-perf(pnn)
     pnn$success_rate
     pnn$bic
     pnn1 <- smooth(pnn1, sigma=0.6)
     pnn1 <- perf(pnn1)
     pnn2$success_rate
     pnn2$bic
     pnn2 <- smooth(pnn2, sigma=0.6)
     pnn2 <- perf(pnn2)
     pnn2$success_rate
     pnn2$bic
```