

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

Навчально-науковий інститут прикладного системного аналізу

Кафедра штучного інтелекту

До захисту допущено:

В. о. завідувачки кафедри

_____ Ірина ДЖИГИРЕЙ

«__» _____ 2026 р.

Дипломна робота

на здобуття ступеня бакалавра

за освітньо-професійною програмою «Системи і методи штучного інтелекту»

спеціальності 122 «Комп'ютерні науки»

**на тему: «Нейромережеві моделі для формування адаптивних комплексів
фізичної реабілітації за медичними показниками»**

Виконав:

студент IV курсу, групи КІ-22

Мамонов Іван Ігорович _____

Керівник:

доцент кафедри штучного інтелекту, к.ф.-м.н., доцент,

Пишнограєв Іван Олександрович _____

Консультант з нормоконтролю:

фахівець першої категорії кафедри штучного інтелекту,

к.т.н., доцент, Комариста Богдана Миколаївна _____

Рецензент:

доцент кафедри економічної кібернетики, к.ф.-м.н., доцент,

Лазаренко Ірина Сергіївна _____

Засвідчую, що у цій дипломній роботі немає
запозичень з праць інших авторів без
відповідних посилань.

Студент _____

Київ – 2026 року

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Навчально-науковий інститут прикладного системного аналізу
Кафедра штучного інтелекту

Рівень вищої освіти – перший (бакалаврський)

Спеціальність – 122 «Комп'ютерні науки»

Освітньо-професійна програма «Системи і методи штучного інтелекту»

ЗАТВЕРДЖУЮ

В. о. завідувачки кафедри

_____ Ірина ДЖИГИРЕЙ

«15» січня 2026 р.

ЗАВДАННЯ
на дипломну роботу студенту
Мамонову Івану Ігоровичу

1. Тема роботи «Нейромережеві моделі для формування адаптивних комплексів фізичної реабілітації за медичними показниками», керівник роботи Пишнограєв Іван Олександрович, доцент кафедри штучного інтелекту, к.ф-м.н., доцент, затверджені наказом по НН ІПСА від «27» травня 2026 р. № 1944-с.
2. Термін подання студентом роботи «05» червня 2026 року.
3. Вихідні дані до роботи: клінічні, демографічні та функціональні показники пацієнтів, а також призначені їм реабілітаційні комплекси, отримані з відкритих медичних баз даних (MIMIC-IV (Medical Information Mart for Intensive Care IV), CARDS (Cardiovascular And Renal Dataset), BRFSS (Behavioral Risk Factor Surveillance System)) та наукових репозиторіїв (Kaggle, Data in Brief) шляхом програмної інтеграції у єдиний мультиномінологічний датасет об'ємом 93 937 записів.
4. Зміст роботи: Аналіз предметної області фізичної реабілітації та відкритих клінічних даних, об'єднаних у мультиномінологічний датасет, дослідження методів динамічного відбору ознак та вибір оптимальних алгоритмів машинного навчання, розробка дворівневої ансамблевої системи для формування індивідуальних реабілітаційних комплексів, оцінка математичної ефективності архітектури, інтерпретація важливості предикторів та підтвердження клінічної валідності отриманих результатів.

5. Перелік ілюстративного матеріалу: електронна презентація.

6. Дата видачі завдання «02» лютого 2026 року.

Календарний план

№ з/п	Назва етапів виконання дипломної роботи	Термін виконання етапів роботи	Примітка
1	Огляд літератури за темою	20.04.2026	Виконано
2	Підготовка першого розділу	27.04.2026	Виконано
3	Підготовка другого розділу	04.05.2026	Виконано
4	Розробка програмного продукту	11.05.2026	Виконано
5	Аналіз результатів	18.05.2026	Виконано
6	Підготовка третього розділу	25.05.2026	Виконано
7	Оформлення дипломної роботи	01.06.2026	Виконано
8	Підготовка презентації доповіді	05.06.2026	Виконано

Студент

Іван МАМОНОВ

Керівник

Іван ПИШНОГРАЄВ

РЕФЕРАТ

Дипломна робота: 107 с., 23 рис., 3 табл., 21 посилань, 2 додатки.

ФІЗИЧНА РЕАБІЛІТАЦІЯ, МАШИННЕ НАВЧАННЯ, НЕЙРОННІ МЕРЕЖІ, ДВОРІВНЕВА АНСАМБЛЕВА АРХІТЕКТУРА, PYTHON, SCIKIT-LEARN, GRADIENT BOOSTING, ВИПАДКОВИЙ ЛІС.

Об'єктом дослідження є процес автоматизованого формування індивідуальних програм фізичної реабілітації на основі комплексних медичних показників пацієнтів. Предметом дослідження є нейромережеві моделі та алгоритми машинного навчання для прогнозування оптимальних клінічних рішень. Метою роботи є розробка та верифікація інтелектуальної системи підтримки прийняття рішень для підбору реабілітаційних рекомендацій, що підвищить якість і доступність медичної допомоги.

Актуальність дослідження зумовлена розривом між потребою в індивідуалізованих реабілітаційних програмах та обмеженими можливостями їхнього формування через дефіцит часу кваліфікованих кадрів і складність урахування коморбідних станів пацієнтів.

У роботі проаналізовано предметну область фізичної реабілітації, розроблено дворівневу ансамблеву архітектуру: перший рівень – 44 незалежні моделі-експерти з динамічним відбором ознак, другий – нейронна мережа-мета-синтезатор. Проведено порівняльний аналіз алгоритмів (логістична регресія, випадковий ліс, градієнтний бустинг, MLP, SVM) та показано переваги ансамблевого підходу.

Отримані результати підтверджують практичну цінність системи. Виявлено, що для кожного медичного рішення визначальними є специфічні патогенетичні показники, які повністю відповідають чинним клінічним протоколам, що підтверджує медичну обґрунтованість системи.

ABSTRACT

Thesis: 107 p., 23 figures, 3 tables, 21 references, 2 appendices.

PHYSICAL REHABILITATION, MACHINE LEARNING, NEURAL NETWORKS, TWO-LEVEL ENSEMBLE ARCHITECTURE, PYTHON, SCIKIT-LEARN, GRADIENT BOOSTING, RANDOM FOREST.

The object of the study is the process of automated formation of individual physical rehabilitation programs based on comprehensive medical indicators of patients. The subject of the study is neural network models and machine learning algorithms for predicting optimal clinical decisions. The purpose of the work is to develop and verify an intelligent decision support system for selecting rehabilitation recommendations, contributing to the improvement of the quality and accessibility of medical care.

The relevance of the study is driven by the gap between the need for individualized rehabilitation programs and the limited capacity for their formation due to a shortage of qualified specialists and the complexity of accounting for patients' comorbid conditions.

The work analyzes the subject area of physical rehabilitation and presents a two-level ensemble architecture: the first level consists of 44 independent expert models with dynamic feature selection, and the second level is a neural network meta-synthesizer. A comparative analysis of algorithms was conducted (logistic regression, random forest, gradient boosting, MLP, SVM), and the advantages of the ensemble approach were demonstrated.

The obtained results confirm the practical value of the developed system. It was found that for each medical decision, the determining factors are specific pathogenetic indicators that fully comply with current clinical protocols, confirming the medical validity of the system.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ	8
ВСТУП	10
РОЗДІЛ 1 ОГЛЯД ПІДХОДІВ ДО КОМП'ЮТЕРИЗОВАНОГО ФОРМУВАННЯ РЕАБІЛІТАЦІЙНИХ ПРОГРАМ	12
1.1 Фізична реабілітація як об'єкт автоматизованого аналізу	12
1.1.1 Поняття, цілі та принципи фізичної реабілітації	12
1.1.2 Класифікація реабілітаційних заходів і методів	14
1.2 Медичні показники як основа для формування реабілітаційних програм.....	16
1.2.1 Категорії медичних показників	16
1.2.2 Методи збору та стандартизації медичних даних	17
1.3 Сучасні підходи до комп'ютеризованого формування реабілітаційних програм	18
1.3.1 Традиційні протокольні підходи та їх обмеження	18
1.3.2 Комп'ютеризовані системи підтримки прийняття рішень у реабілітації	20
1.3.3 Застосування методів машинного навчання в медичній реабілітації	21
1.4 Постановка задачі дослідження	22
Висновки до розділу 1	24
РОЗДІЛ 2 МЕТОДИ МАШИННОГО НАВЧАННЯ ДЛЯ ЗАДАЧІ ПІДБОРУ РЕАБІЛІТАЦІЙНИХ ПРОГРАМ	25
2.1 Особливості застосування машинного навчання у медичній реабілітації	25
2.2 Огляд та порівняння моделей машинного навчання для задачі дослідження	27
2.3 Методи підготовки медичних даних для навчання моделей	32

2.4 Методи навчання та оцінювання моделей	34
2.5 Обґрунтування програмних засобів реалізації.....	36
Висновки до розділу 2.....	37
РОЗДІЛ 3 ПРОГРАМНА РЕАЛІЗАЦІЯ ТА АНАЛІЗ РЕЗУЛЬТАТІВ	39
3.1 Опис та характеристики датасету	39
3.1.1 Джерела даних та методологія формування датасету	39
3.1.2 Структура та склад даних.....	41
3.1.3 Аналіз розподілу цільових змінних та дисбаланс класів	46
3.2 Передобробка даних та підготовка навчальних вибірок	49
3.2.1 Нормалізація та кодування ознак.....	49
3.2.2 Динамічний відбір ознак для кожної цільової змінної	51
3.2.3 Формування навчальної та тестової вибірок	54
3.3 Навчання моделей-експертів	56
3.3.1 Архітектура ансамблю моделей-експертів.....	56
3.3.2 Критерій відбору найкращого експерта та порівняння моделей.....	58
3.4 Дворівнева ансамблева архітектура	61
3.4.1 Побудова матриці мета-ознак	61
3.4.2 Архітектура мета-мозку та оцінювання фінальної ансамблевої моделі	63
3.5 Базові моделі для порівняння	64
3.6 Аналіз та інтерпретація результатів	67
3.6.1 Важливість ознак та клінічна інтерпретація	67
3.6.2 Аналіз помилок моделі.....	69
3.6.3 Обговорення обмежень та напрями розвитку системи.....	70
Висновки до розділу 3.....	71
ВИСНОВКИ	73
ПЕРЕЛІК ПОСИЛАНЬ	74
ДОДАТОК А ЛІСТИНГ ПРОГРАМИ	77
ДОДАТОК Б ЕЛЕКТРОННА ПРЕЗЕНТАЦІЯ.....	102

ПЕРЕЛІК СКОРОЧЕНЬ

- AUC – Area Under the Curve (площа під кривою)
- EHR – Electronic Health Records (електронні медичні записи)
- FIM – Functional Independence Measure (шкала функціональної незалежності)
- GRU – Gated Recurrent Unit (керована рекурентна одиниця)
- LIME – Local Interpretable Model-agnostic Explanations (локальні модельно-незалежні пояснення)
- LSTM – Long Short-Term Memory (довга короткочасна пам'ять)
- MAE – Mean Absolute Error (середня абсолютна похибка)
- MCAR – Missing Completely At Random (пропуск повністю випадковий)
- MICE – Multiple Imputation by Chained Equations (множинна імпутація ланцюговими рівняннями)
- MLP – Multilayer Perceptron (багатошаровий перцептрон)
- MSE – Mean Squared Error (середньоквадратична похибка)
- PCA – Principal Component Analysis (аналіз головних компонент)
- ReLU – Rectified Linear Unit (функція активації)
- RMSE – Root Mean Squared Error (корінь середньоквадратичної похибки)
- RNN – Recurrent Neural Network (рекурентна нейронна мережа)
- ROC – Receiver Operating Characteristic (крива операційних характеристик)
- SHAP – SHapley Additive exPlanations (значення Шеплі для пояснення моделей)
- SMOTE – Synthetic Minority Over-sampling Technique (метод синтетичного перевибірки)
- SVM – Support Vector Machine (машина опорних векторів)
- XAI – Explainable Artificial Intelligence (пояснюваний штучний інтелект)
- XGBoost – eXtreme Gradient Boosting (екстремальний градієнтний бустинг)
- ІМТ – індекс маси тіла
- ВАШ – візуальна аналогова шкала болю

ЖЄЛ – життєва ємність легень

ЛФК – лікувальна фізична культура

МКФ – Міжнародна класифікація функціонування

МКХ – Міжнародна класифікація хвороб

МН – машинне навчання

СППР – система підтримки прийняття рішень

ЧСС – частота серцевих скорочень

ВСТУП

Фізична¹ реабілітація є одним із ключових напрямів сучасної медицини, що спрямована на відновлення функціональних можливостей пацієнтів після травм, операцій, гострих захворювань або за умов хронічної патології. Ефективність реабілітаційного процесу значною мірою залежить від своєчасного формування індивідуалізованих програм фізичних вправ, які враховують клінічний стан пацієнта та особливості перебігу захворювання. Традиційні протокольні підходи, що спираються на стандартизовані схеми лікування, не завжди здатні забезпечити необхідний рівень адаптивності до індивідуальних потреб пацієнта, що суттєво знижує ефективність реабілітації.

Актуальність дослідження зумовлена стрімким розвитком цифрових технологій у сфері охорони здоров'я та накопиченням великих масивів медичних даних у системах електронних медичних записів. Це відкриває принципово нові можливості для застосування методів штучного інтелекту, зокрема нейромережевих моделей, для автоматизованого аналізу медичних показників пацієнта і підбору персоналізованих реабілітаційних комплексів. Нейромережеві підходи демонструють здатність виявляти складні нелінійні залежності між клінічними параметрами та оптимальними реабілітаційними втручаннями, що недосяжне для традиційних статистичних методів.

Незважаючи на зростаючий інтерес до застосування машинного навчання в медицині, питання порівняльного аналізу та вибору найбільш ефективної нейромережевої моделі для задачі підбору комплексів фізичної реабілітації на основі початкових медичних показників пацієнта залишається недостатньо

¹ Тут і нижче використано такий інструмент штучного інтелекту як чат-бот з генеративним штучним інтелектом ChatGPT виключно для корегування та редагування тексту, створеного автором цієї дипломної роботи, на основі автоматизованої перевірки граматики, структури та стилю, що відповідає Політиці використання штучного інтелекту для академічної діяльності в КПІ ім. Ігоря Сікорського (протокол №11 Вченої ради КПІ ім. Ігоря Сікорського від 11 грудня 2023 р.).

дослідженням. Більшість існуючих систем підтримки прийняття клінічних рішень або зосереджені на діагностиці, або використовують спрощені правилі механізми рекомендацій. Це зумовлює практичну значущість дослідження, спрямованого на порівняння різних моделей машинного навчання для вирішення задачі персоналізованого підбору реабілітаційної програми.

На міжнародному рівні питання цифровізації охорони здоров'я набувають дедалі більшого значення. Стратегія ВООЗ щодо цифрового здоров'я на 2020–2025 роки прямо визначає штучний інтелект та машинне навчання як пріоритетні інструменти підвищення якості та доступності медичної допомоги. У країнах Європейського Союзу активно реалізуються ініціативи зі створення єдиного простору медичних даних, що відкриває принципово нові можливості для навчання моделей на великих міжнаціональних наборах даних. В Україні в контексті відновлення після збройного конфлікту та масового зростання потреб у реабілітаційній допомозі вкрай важлива необхідність масштабованих, автоматизованих систем підбору реабілітаційних програм є особливо актуальною. Розробка ефективних моделей машинного навчання для вирішення задач медичної реабілітації є не лише науковим завданням, а й нагальною соціальною потребою.

РОЗДІЛ 1 ОГЛЯД ПІДХОДІВ ДО КОМП'ЮТЕРИЗОВАНОГО ФОРМУВАННЯ РЕАБІЛІТАЦІЙНИХ ПРОГРАМ

1.1 Фізична реабілітація як об'єкт автоматизованого аналізу

1.1.1 Поняття, цілі та принципи фізичної реабілітації

Фізична реабілітація є самостійною медичною дисципліною, що ґрунтується на застосуванні засобів фізичного впливу з метою відновлення, збереження та зміцнення здоров'я людини, порушеного внаслідок захворювань, травм або функціональних розладів. У сучасній медицині вона розглядається не як допоміжний інструмент лікування, а як повноцінний, науково обґрунтований напрям, що інтегрується з клінічною медициною, фізіологією, біомеханікою та психологією.

Відповідно до визначення Всесвітньої організації охорони здоров'я, реабілітація – це комплекс заходів, спрямованих на те, щоб люди з порушеннями здоров'я могли досягти та підтримувати оптимальний рівень фізичного, інтелектуального, психологічного та соціального функціонування. Фізична реабілітація як складова цього комплексу зосереджена переважно на відновленні рухових функцій, підвищенні фізичної працездатності та адаптації організму до навантажень після патологічних станів [1].

Цілі фізичної реабілітації класифікують за часовим горизонтом. Стратегічні цілі охоплюють максимально можливе відновлення функціональних можливостей пацієнта, повернення до трудової та соціальної активності, а також профілактику ускладнень. Оперативні цілі конкретного реабілітаційного курсу включають усунення больового синдрому, відновлення амплітуди рухів у суглобах, зміцнення м'язового апарату, нормалізацію серцево-судинних та дихальних показників.

Досягнення зазначених цілей забезпечується дотриманням системи принципів фізичної реабілітації. Принцип індивідуалізації передбачає розроблення реабілітаційної програми з урахуванням нозологічного діагнозу, стадії захворювання, віку, статі, антропометричних характеристик і рівня фізичної

підготовленості пацієнта. Принцип поступовості та систематичності вимагає поетапного нарощування фізичних навантажень – від мінімальних до субмаксимальних. Принцип адекватності навантажень означає відповідність інтенсивності та характеру вправ поточному функціональному стану пацієнта, що оцінюється на основі об'єктивних медичних показників. Принцип комплексності відображає необхідність поєднання різних засобів і форм фізичної реабілітації в єдиній програмі.

Перелічені принципи утворюють методологічну основу сучасної фізичної реабілітації і безпосередньо визначають вимоги до систем, що автоматизують формування реабілітаційних комплексів. Зокрема, принцип індивідуалізації є ключовим аргументом на користь застосування моделей машинного навчання, здатних враховувати багатовимірний профіль початкового стану пацієнта та підбирати на його основі оптимальну програму реабілітації.

Важливою концептуальною основою сучасної реабілітаційної практики є Міжнародна класифікація функціонування, обмежень життєдіяльності та здоров'я (МКФ), запроваджена ВООЗ у 2001 році. МКФ описує функціонування людини через три взаємопов'язані виміри: структури та функції організму, активність, а також участь у суспільному житті. На відміну від традиційного нозологічного підходу, що зосереджується виключно на патологічному процесі, МКФ акцентує увагу на функціональних наслідках захворювання та якості життя пацієнта. З точки зору автоматизованих систем підбору реабілітаційних програм концепція МКФ є принципово важливою: вона розширює простір вхідних ознак моделі від суто клінічних показників до функціональних та соціальних параметрів. Реабілітаційна програма, сформована на основі МКФ-профілю пацієнта, є більш цілісною і спрямованою не лише на відновлення ураженої функції, а й на максимальну реінтеграцію пацієнта у повсякденне та соціальне життя. Саме ця багатовимірність робить задачу підбору реабілітаційної програми особливо складною для традиційних аналітичних методів і водночас перспективною для застосування нейромережових моделей.

1.1.2 Класифікація реабілітаційних заходів і методів

Різноманітність патологічних станів, що є показаннями до фізичної реабілітації, зумовлює існування широкого спектру реабілітаційних заходів і методів. Їх систематизація є необхідною умовою як для клінічної практики, так і для побудови автоматизованих систем формування реабілітаційних програм.

За характером впливу всі реабілітаційні засоби поділяються на активні та пасивні. Активні методи передбачають безпосередню м'язову роботу пацієнта і включають лікувальну фізкультуру (ЛФК), кінезіотерапію та ерготерапію. Пасивні методи реалізуються без вольових зусиль – масаж, мануальна терапія, механотерапія та більшість фізіотерапевтичних процедур.

За провідним засобом впливу виокремлюють такі основні групи методів. Лікувальна фізична культура (ЛФК) є центральним методом, що охоплює гімнастичні вправи, дозовану ходьбу, плавання та лікувальний біг. Фізіотерапія об'єднує методи апаратного фізичного впливу: електротерапію, магнітотерапію, ультразвукову терапію, лазеротерапію та гідротерапію. Масаж та мануальна терапія спрямовані на нормалізацію м'язового тону та покращення мікроциркуляції. Ерготерапія зосереджена на відновленні навичок самообслуговування та побутової активності [2].

Класифікація за інтенсивністю фізичного навантаження є одним із ключових параметрів при автоматизованому формуванні реабілітаційних програм. Інтенсивність навантаження оцінюється на основі частоти серцевих скорочень як відсотка від максимально допустимого значення. Цільова зона тренувального навантаження для конкретного пацієнта задається за формулою Карвонена:

$$\text{ЧСС}_{\text{цільова}} = \text{ЧСС}_{\text{спокій}} + k \cdot (\text{ЧСС}_{\text{max}} - \text{ЧСС}_{\text{спокій}}),$$

де k – коефіцієнт інтенсивності від 0,40 до 0,85 залежно від реабілітаційного етапу та функціонального класу пацієнта [3].

Наведена класифікація є вихідною основою для формалізації простору реабілітаційних втручань при побудові моделей машинного навчання. Кожен реабілітаційний захід може бути описаний вектором ознак, що включає тип методу, інтенсивність, тривалість та частоту застосування. Це створює передумови для подання задачі підбору оптимального комплексу як задачі класифікації або регресії засобами машинного навчання.

Невід'ємною складовою системи фізичної реабілітації є стандартизовані інструменти оцінювання функціонального стану пацієнта. Вони виконують подвійну роль: по-перше, формують об'єктивну вхідну інформацію для підбору реабілітаційної програми; по-друге, дозволяють відслідковувати динаміку відновлення в процесі курсу. Серед найбільш поширених та валідованих шкал у контексті автоматизованого формування реабілітаційних програм слід виокремити наступні. Індекс Бартел (Barthel Index) є одним із найширше використовуваних інструментів оцінювання здатності до самообслуговування та щоденної активності; містить 10 категорій функціональних можливостей із максимальною сумою балів 100. Функціональна шкала незалежності (Functional Independence Measure, FIM) розширює концепцію Бартелу, охоплюючи 18 параметрів у сферах моторики та когніції, і широко застосовується у гострій реабілітації. Шкала Борга (Borg Rating of Perceived Exertion, RPE) дозволяє суб'єктивно оцінити інтенсивність фізичного навантаження за 20-бальною шкалою та є незамінним інструментом для моніторингу інтенсивності вправ безпосередньо під час занять. Числові значення цих шкал утворюють кількісні ознаки, що безпосередньо використовуються як вхідні дані моделей машинного навчання і мають доведену предикторну цінність щодо реабілітаційних результатів.

1.2 Медичні показники як основа для формування реабілітаційних програм

1.2.1 Категорії медичних показників

Формування реабілітаційної програми принципово залежить від якості та повноти вхідних даних про початковий стан пацієнта. У реабілітаційній медицині медичні показники прийнято групувати за чотирма основними категоріями: нозологічні, функціональні, антропометричні та лабораторні показники.

Нозологічні показники описують характер і структуру захворювання або травми, що стали підставою для реабілітації: основний діагноз, стадія та ступінь тяжкості, характер перебігу, наявність супутніх захворювань. Для задач машинного навчання нозологічні показники, як правило, представляються у вигляді категоріальних змінних, що вимагає попередньої кодифікації методом one-hot encoding або ієрархічного кодування.

Функціональні показники відображають актуальний стан фізіологічних систем організму пацієнта на момент початку реабілітації. Вони охоплюють показники серцево-судинної та дихальної систем (ЧСС, артеріальний тиск, SpO₂ (Saturation of peripheral Oxygen)), показники опорно-рухового апарату (амплітуда рухів, м'язова сила, рівень болю за шкалою ВАШ), показники нейром'язової функції (координація, рівновага, швидкість ходьби) та показники повсякденної активності (індекс Бартел, шкала FIM).

Антропометричні показники характеризують фізичні параметри тіла пацієнта і є необхідними для розрахунку індивідуальних норм навантаження. Індекс маси тіла розраховується як відношення маси тіла у кілограмах до квадрату зросту у метрах. При значеннях ІМТ понад 30 кг/м² ударне навантаження (біг, стрибки) як правило замінюється на вправи у воді або велоергометрі.

Лабораторні показники надають об'єктивну інформацію про біохімічний і гематологічний статус пацієнта: рівень гемоглобіну, С-реактивного білка, глюкози, ліпідного профілю. Чотири категорії медичних показників утворюють

гетерогенний вхідний простір, що включає змінні різної природи: категоріальні (нозологічні), числові неперервні (функціональні, антропометричні, лабораторні) та порядкові (клінічні шкали). Гетерогенність показників є одним із ключових викликів при проектуванні нейромережевої архітектури [4].

1.2.2 Методи збору та стандартизації медичних даних

Збір медичних даних у реабілітаційному контексті здійснюється через кілька каналів. Клінічне обстеження та ручне введення даних залишається основним методом у більшості закладів. Електронні медичні записи (EHR) є основним джерелом структурованих клінічних даних для систем машинного навчання, хоча реальні бази містять значну частку напівструктурованих даних. Стандартизовані клінічні шкали та тести є найбільш надійним джерелом функціональних ознак завдяки встановленим нормам і психометричним властивостям.

Числова нормалізація є обов'язковим етапом підготовки даних перед подачею на вхід моделі машинного навчання. Для показників із відомим фізіологічним діапазоном застосовується min-max нормалізація:

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}},$$

де початкове значення показника, x_{min} та x_{max} мінімальне та максимальне значення у вибірці. Для показників із нормальним розподілом застосовується Z-score стандартизація:

$$x' = \frac{x - \mu}{\sigma}.$$

Для показників із суттєво скошеним розподілом (С-реактивний білок, креатинкіназа) доцільним є логарифмічне перетворення.

Реальні клінічні дані принципово відрізняються від ідеалізованих навчальних наборів. Їх характерними рисами є неоднорідність структури, значна частка пропущених значень, систематичні зміщення та шум вимірювань. Частка пропущених значень у клінічних базах даних може сягати 30–80% залежно від показника та типу закладу.

Неоднорідність медичних даних виявляється на кількох рівнях. Структурна неоднорідність виникає при об'єднанні даних з різних інформаційних систем. Популяційна неоднорідність відображає відмінності між підгрупами пацієнтів. Вимірювальна неоднорідність зумовлена залежністю результату від дослідника.

Розрізняють три механізми пропуску даних: MCAR (Missing Completely At Random) – пропуск є повністю випадковим; MAR (Missing At Random) – ймовірність пропуску залежить від інших спостережуваних змінних; MNAR (Missing Not At Random) – ймовірність пропуску залежить від самого пропущеного значення, що є найбільш проблематичним, оскільки призводить до систематичного зміщення вибірки. Для медичних даних реабілітаційного профілю найбільш обґрунтованим є поєднання множинної імпутації для показників із механізмом MAR та індикаторного кодування для показників із підозрою на MNAR [5].

1.3 Сучасні підходи до комп'ютеризованого формування реабілітаційних програм

1.3.1 Традиційні протокольні підходи та їх обмеження

Традиційна реабілітаційна практика ґрунтується на клінічних протоколах, розроблених на підставі систематичних оглядів і рандомізованих контрольованих досліджень. Їхня нормативна основа забезпечує стандартизацію практики та мінімізує міжлікарську варіабельність. Водночас протоколи мають ряд системних обмежень.

По-перше, протоколи стратифікують пацієнтів за дискретними категоріями функціонального класу (як правило, три-чотири класи), тоді як реальний клінічний профіль є безперервним і багатовимірним. Два пацієнти з ідентичним діагнозом і формально однаковим функціональним класом, але суттєво відмінними конкретними показниками, отримують одну й ту саму програму. По-друге, протоколи мають низьку чутливість до коморбідної патології. По-третє, частота оновлення клінічних протоколів не відповідає темпу накопичення нових клінічних знань – більшість протоколів переглядаються раз на п'ять-десять років. Нарешті, традиційні протоколи не є масштабованими в умовах зростання навантаження на систему охорони здоров'я та дефіциту реабілітаційного персоналу. Зазначені обмеження визначають запит на технологічні рішення, здатні забезпечити вищий рівень персоналізації на основі індивідуального профілю пацієнта.

Показовим прикладом обмежень протокольного підходу є реабілітація після тотального ендопротезування кульшового суглоба. Стандартний клінічний протокол визначає чотири послідовних фази (гостра, рання відновна, пізня відновна, підтримувальна) з відповідними наборами вправ та критеріями переходу між ними. Проте для пацієнта 65 років з ІМТ 34 та помірною серцевою недостатністю той самий протокол виявляється або надмірно інтенсивним – з ризиком кардіоваскулярних ускладнень – або занадто консервативним з точки зору ортопедичного відновлення. Подібна дихотомія є нерозв'язною у рамках жорсткого протокольного підходу і потребує або індивідуальної корекції лікарем, або автоматизованого механізму персоналізації – якому і присвячене дане дослідження. Масштабована система, яка може враховувати повний вектор клінічних ознак пацієнта без обмежень щодо їх кількості та типу, принципово знімає це протиріччя.

1.3.2 Комп'ютеризовані системи підтримки прийняття рішень у реабілітації

Комп'ютеризовані системи підтримки прийняття рішень (СППР) є технологічними відповідями на обмеження традиційних протокольних підходів. Під СППР у медичному контексті розуміють програмні системи, які аналізують клінічні дані пацієнта та надають лікарю або реабілітологу структуровані рекомендації щодо діагностики, лікування або реабілітації.

За архітектурним принципом виділяють три основні класи СППР. Системи на базі правил є найбільш поширеним типом у клінічній практиці, їх ядро складає база знань у формі продукційних правил «якщо – то». Перевагою є прозорість логіки прийняття рішень; суттєвими недоліками є висока трудомісткість розробки та відсутність механізму автоматичного навчання. Системи на основі онтологічних та семантичних баз знань забезпечують формальні моделі знань про предметну область. Системи на основі статистичних і аналітичних моделей генерують рекомендації на підставі закономірностей, виявлених у клінічних базах даних, і здатні навчатися на нових даних.

Серед відомих прикладів реабілітаційних СППР – система RehabMAGIC, орієнтована на реабілітацію після інсульту, яка автоматично формує рекомендації на основі шкали FIM; система PhysiAssist для ортопедичної реабілітації, що використовує байєсівські мережі для прогнозування оптимального часового графіку навантаження після ендопротезування. Незважаючи на доведену клінічну ефективність окремих СППР, більшість з них зосереджені на вузькопрофесійних завданнях і не забезпечують інтегрованого формування повного реабілітаційного комплексу. Ці обмеження формують запит на методи машинного навчання.

1.3.3 Застосування методів машинного навчання в медичній реабілітації

Методи машинного навчання (МН) є логічним продовженням розвитку комп'ютеризованих СППР. Якщо традиційні СППР ґрунтуються на експліцитно формалізованих правилах, то методи МН можуть автоматично виявляти складні нелінійні залежності з клінічних даних без необхідності їх явного програмування. За даними систематичних оглядів, кількість публікацій, присвячених застосуванню МН у реабілітаційній медицині, зросла більш ніж удвічі у період між 2015 та 2022 роками.

Застосування МН у медичній реабілітації охоплює кілька класів задач. Задачі прогнозування результатів реабілітації призначені для передбачення функціонального стану пацієнта після завершення курсу. Задачі класифікації та рекомендації реабілітаційних втручань є найбільш пов'язаними з темою даної роботи і формально являють собою задачі багатокласової класифікації або рекомендаційного прогнозування. Задачі прогнозування ризику ускладнень спрямовані на раннє виявлення негативних наслідків.

Систематичний огляд Hsu et al. показав, що моделі МН (Random Forest та нейронні мережі) прогнозують функціональні результати з точністю, що суттєво перевищує традиційні лінійні регресійні моделі, особливо при врахуванні нелінійних взаємодій між початковими клінічними показниками. Застосування МН у медичній реабілітації пов'язане з рядом методологічних викликів: проблемою малих і незбалансованих навчальних вибірок, гетерогенністю та пропусками в клінічних даних, а також вимогою інтерпретованості моделей як умовою клінічного прийому системи [6].

Серед конкретних прикладів застосування МН у реабілітаційній медицині особливої уваги заслуговують дослідження з кількох нозологічних груп. У кардіологічній реабілітації Ghassemi et al. застосували рекурентні нейронні мережі для прогнозування оптимального навантажувального режиму після інфаркту міокарда на основі даних добового моніторингу ЕКГ та функціональних проб:

точність класифікації пацієнтів за рекомендованим режимом навантаження склала 84,3%, що на 17 відсоткових пунктів перевищило результати клінічних настанов. У неврологічній реабілітації після інсульту Jung et al. застосували алгоритми градієнтного бустингу для передбачення функціонального результату за шкалою Бартел з чутливістю 0,81 та специфічністю 0,79 на незалежній тестовій вибірці. В ортопедичній реабілітації моделі Random Forest використовувались для прогнозування тривалості курсу фізіотерапії після ушкоджень плечового суглоба, демонструючи середню абсолютну похибку прогнозу у 2,3 тижні порівняно з 4,7 тижнями для стандартних протоколів. Спільним висновком наведених досліджень є підтвердження доцільності застосування МН для персоналізації реабілітаційних втручань, а також вказівка на те, що жоден алгоритм не є універсально кращим – оптимальний вибір залежить від конкретної задачі, нозологічного профілю та характеристик навчальних даних. Це прямо обґрунтовує необхідність систематичного порівняльного дослідження, яке є метою даної роботи [7] [8].

1.4 Постановка задачі дослідження

Проведений аналіз предметної області демонструє, що традиційні підходи до формування реабілітаційних комплексів обмежені в забезпеченні необхідного рівня персоналізації. Накопичення масивів медичних даних та розвиток методів машинного навчання відкривають можливості для автоматизованого підбору реабілітаційних програм на основі індивідуального профілю пацієнта.

У межах даної роботи ставиться задача порівняльного аналізу моделей машинного навчання для автоматизованого підбору комплексів фізичної реабілітації на основі початкових медичних показників пацієнта. Основною метою є визначення моделі, що забезпечує найвищу точність відповідності підібраної програми реабілітаційним потребам пацієнта.

Формально кожен пацієнт представляється вектором ознак $X = (x_1, x_2, \dots, x_n)$, де x_1 – вік; x_2 – стать; x_3 – нозологічний діагноз; x_4, x_5, x_6 – антропометричні показники (маса, зріст, ІМТ); x_7, x_8 – кардіоваскулярні параметри (ЧСС, артеріальний тиск); x_9 – рівень фізичної активності; x_{10} – показники функціональних проб; x_{11} – наявність больового синдрому; x_{12} – результати лабораторних досліджень.

На основі цього вектора модель формує набір вихідних параметрів $Y = (y_1, y_2, \dots, y_m)$, де y_1 – рекомендований тип вправ; y_2 – інтенсивність навантаження; y_3 – тривалість заняття; y_4 – частота проведення занять; y_5 – необхідність фізіотерапії; y_6 – доцільність масажу чи механотерапії. Залежно від формату вихідних даних задача розглядається як задача багатокласової класифікації (вибір із готових реабілітаційних комплексів) або як задача регресії (прогнозування числових параметрів навантаження).

Вхідні дані системи формуються на основі сукупності медичних показників чотирьох категорій, описаних у підрозділі 1.2.1. Перед подачею до моделі дані потребують попередньої обробки: очищення від помилкових значень, заповнення пропусків, кодування категоріальних ознак та нормалізація числових параметрів.

Очікуваним результатом є автоматизований підбір персоналізованого комплексу фізичної реабілітації, що містить рекомендований тип фізичних вправ, рівень інтенсивності навантаження, тривалість заняття, частоту проведення процедур, рекомендовані фізіотерапевтичні методи та допустимий діапазон ЧСС під час навантаження.

Для задач класифікації застосовуються метрики: точність (Accuracy), повнота (Recall), точність позитивних прогнозів (Precision) та F1-міра. Для задач регресії: середня абсолютна похибка (MAE), середньоквадратична похибка (MSE) та коефіцієнт детермінації R^2 . Окрім математичних метрик, у медичних задачах критичними є клінічні критерії ефективності: безпечність рекомендацій, відповідність клінічним протоколам та інтерпретованість результатів для медичного персоналу.

Висновки до розділу 1

У першому розділі проведено комплексний аналіз предметної області фізичної реабілітації та сформульовано постановку задачі дослідження.

Встановлено, що фізична реабілітація є науково обґрунтованою дисципліною з чіткою системою цілей і принципів, ключовим серед яких є принцип індивідуалізації – саме він визначає вимоги до автоматизованих систем підтримки прийняття рішень.

Систематизовано чотири категорії медичних показників (нозологічні, функціональні, антропометричні та лабораторні), що формують гетерогенний вхідний простір моделі машинного навчання. Розглянуто методи збору та стандартизації медичних даних, а також проблеми неоднорідності та пропущених значень.

Проведено аналіз існуючих підходів до комп'ютеризованого формування реабілітаційних програм: традиційних протокольних підходів, комп'ютеризованих СППР на базі правил, онтологічних та статистичних моделей. Показано, що кожен із підходів має системні обмеження у частині персоналізації та обробки складних нелінійних залежностей, що обґрунтовує доцільність застосування методів машинного навчання.

Сформульовано постановку задачі: порівняльний аналіз моделей машинного навчання для автоматизованого підбору комплексів фізичної реабілітації на основі початкових медичних показників пацієнта. Визначено структуру вхідного та вихідного просторів, а також критерії оцінювання ефективності. Отримані результати закладають методологічну основу для розгляду та порівняння конкретних алгоритмів у наступному розділі.

РОЗДІЛ 2 МЕТОДИ МАШИННОГО НАВЧАННЯ ДЛЯ ЗАДАЧІ ПІДБОРУ РЕАБІЛІТАЦІЙНИХ ПРОГРАМ

2.1 Особливості застосування машинного навчання у медичній реабілітації

Сфера фізичної реабілітації є однією з найбільш складних галузей для застосування методів машинного навчання через значну кількість факторів, що одночасно впливають на підбір оптимальної програми відновлення. На відміну від багатьох класичних задач прогнозування, де система працює з відносно стабільними та добре структурованими даними, у реабілітаційній медицині модель повинна враховувати високу варіативність клінічних показників та індивідуальні особливості організму кожного пацієнта.

Основною особливістю задач машинного навчання у фізичній реабілітації є багатofакторність. Стан пацієнта визначається великою кількістю взаємопов'язаних параметрів: віком, статтю, антропометричними характеристиками, основним діагнозом, супутніми захворюваннями, функціональними показниками серцево-судинної та дихальної систем, рівнем фізичної активності та психоемоційним станом. Усі ці параметри можуть одночасно впливати на вибір реабілітаційних заходів та реакцію організму на фізичне навантаження.

Особливу складність становить нелінійний характер взаємозв'язків між медичними показниками та оптимальною програмою реабілітації. Підвищення інтенсивності фізичного навантаження може бути доцільним для одного пацієнта, але небезпечним для іншого через відмінності у функціональному резерві серцево-судинної системи або наявність супутніх патологій. Традиційні лінійні моделі часто не здатні адекватно описати подібні залежності.

Важливою характеристикою медичних даних є їх неоднорідність. У межах однієї системи можуть одночасно використовуватись числові, категоріальні та

текстові дані. Таке різноманіття форматів ускладнює процес побудови моделей та потребує спеціальних методів попередньої обробки інформації.

Обмежений обсяг доступних медичних даних є ще однією суттєвою особливістю. На відміну від сфер, де доступні мільйони прикладів для навчання, у медичній реабілітації кількість якісно розмічених клінічних записів часто є обмеженою через конфіденційність медичної інформації та відсутність стандартизованих наборів даних. Обмежений розмір вибірки підвищує ризик перенавчання, що потребує застосування методів регуляризації та крос-валідації.

Суттєвою особливістю задач фізичної реабілітації є вимога клінічної безпеки. Некоректні рекомендації здатні безпосередньо впливати на здоров'я пацієнта. Тому модель повинна враховувати клінічні обмеження, а лікар або реабілітолог повинен мати можливість зрозуміти підстави сформованої рекомендації, що є принциповою умовою для практичного впровадження.

Вимога інтерпретованості прогнозів є однією з найгостріших проблем на перетині машинного навчання та клінічної медицини. У загальній класифікації методів розрізняють “прозорі” (white-box) моделі, логіка яких безпосередньо зрозуміла без додаткових інструментів (лінійні моделі, дерева рішень), та “чорні скриньки” (black-box) – нейронні мережі та ансамблеві методи, що забезпечують вищу точність, проте потребують методів пояснюваного штучного інтелекту (eXplainable AI, XAI) для інтерпретації. У медичному контексті нерозуміння логіки рекомендації є не лише академічною проблемою: лікар несе юридичну відповідальність за призначене лікування і не може просто делегувати рішення алгоритму без можливості критично оцінити його обґрунтованість. Сучасні методи XAI – зокрема SHAP (SHapley Additive exPlanations) та LIME (Local Interpretable Model-agnostic Explanations) – дозволяють локально пояснювати рішення будь-якої “чорної скриньки”, зазначаючи, які медичні показники і якою мірою вплинули на конкретну рекомендацію для конкретного пацієнта. Врахування цих інструментів при розробці та оцінюванні моделей є принциповою вимогою клінічно прийнятного рішення [9].

Методи машинного навчання здатні автоматично виявляти приховані закономірності між великою кількістю взаємопов'язаних параметрів без необхідності жорсткого задання правил або ручного виділення ознак. Це особливо важливо у медичній реабілітації, де клінічні процеси мають складний нелінійний характер і залежать від багатьох факторів одночасно.

Однією з головних переваг є здатність моделей машинного навчання працювати зі складними нелінійними залежностями. Вплив одного показника на рекомендовану програму реабілітації може кардинально змінюватись залежно від значень інших показників – класичні алгоритми часто не здатні описати подібні залежності.

Здатність до узагальнення дозволяє після навчання на певному наборі клінічних даних формувати рекомендації для нових пацієнтів, яких не було у навчальній вибірці, що є необхідною умовою практичного застосування у реальних медичних умовах. Важливою перевагою є також можливість автоматичного виділення ознак: моделі самостійно визначають найбільш значущі комбінації параметрів у процесі навчання, що особливо корисно для медичних даних, де взаємозв'язки між показниками часто є прихованими.

2.2 Огляд та порівняння моделей машинного навчання для задачі дослідження

Логістична регресія є базовим алгоритмом для задач класифікації, що встановлює лінійну межу між класами у просторі ознак. Для задачі підбору реабілітаційної програми вона може розглядатись як базова (baseline) модель для порівняння. Ймовірність належності до певного класу обчислюється як:

$$P(y = k | X) = \text{softmax}(W \cdot X + b),$$

де W – матриця вагових коефіцієнтів, b – зміщення, X – вектор вхідних ознак. Основною перевагою логістичної регресії є висока інтерпретованість: вагові коефіцієнти безпосередньо відображають внесок кожного медичного показника у формування рекомендації, що є суттєвою перевагою у клінічному контексті. Суттєвим обмеженням є припущення про лінійну роздільність класів, яке у задачах медичної реабілітації, як правило, не виконується через складний нелінійний характер залежностей між клінічними показниками та оптимальною реабілітаційною програмою [10].

Дерева рішень будують модель у вигляді ієрархічної структури умовних правил, розбиваючи простір ознак на регіони. На кожному вузлі дерево вибирає ознаку та поріг розбиття, що мінімізує критерій нечистоти (ентропія або критерій Джині). Завдяки такій структурі дерева рішень є природно інтерпретованими: реабілітолог може відстежити логіку прийняття рішення. Проте одиночне дерево схильне до перенавчання та є нестабільним при зміні вибірки.

Random Forest є ансамблевим методом, що об'єднує велику кількість дерев рішень, навчених на випадкових підвибірках даних і підмножинах ознак. Кінцева рекомендація формується голосуванням більшості дерев:

$$\hat{y} = \operatorname{argmax}_k \sum 1[T_{i(X)} = k], i = 1..M,$$

де M – кількість дерев, $T_{i(X)}$ – прогноз i -го дерева. Random Forest є більш стійким до шуму та перенавчання порівняно з одиночним деревом, добре працює з гетерогенними медичними даними та різними типами ознак, а також забезпечує оцінку важливості ознак. Зазначені властивості роблять Random Forest одним із найбільш придатних алгоритмів для задач класифікації реабілітаційних програм [11].

Алгоритми градієнтного бустингу є потужними ансамблевими методами, що будують моделі послідовно: кожна наступна модель зосереджується на помилках попередньої. На кожній ітерації навчається нове дерево, яке апроксимує негативний градієнт функції втрат. Фінальний прогноз формується як:

$$F_{M(x)} = \sum \eta \cdot h_{m(x)}, m = 1..M,$$

де η – швидкість навчання, $h_{m(x)}$ – m -те дерево. XGBoost та LightGBM є сучасними ефективними реалізаціями градієнтного бустингу з підтримкою регуляризації, обробки пропущених значень і паралельних обчислень. У задачах класифікації реабілітаційних програм ці алгоритми часто демонструють найвищу точність серед класичних методів завдяки здатності вловлювати складні нелінійні залежності між медичними показниками. Gradient Boosting є одним із ключових претендентів для порівняльного аналізу в даній роботі [12].

Метод опорних векторів будує оптимальну гіперплощину розділення між класами, максимізуючи відступ (margin) від найближчих точок навчальної вибірки. Для нелінійно роздільних даних SVM використовує kernel trick – неявне відображення у простір вищої розмірності за допомогою ядрової функції. Найбільш поширеним є RBF-ядро:

$$K(x, x') = \exp(-\gamma ||x - x'||^2),$$

де γ – параметр, що контролює ширину ядра. SVM демонструє хорошу ефективність на задачах класифікації з невеликим обсягом даних та добре розробленими ознаками. Для медичних даних реабілітаційного профілю SVM може бути ефективним при правильному виборі ядра та налаштуванні гіперпараметрів. Проте SVM погано масштабується для великих наборів даних і потребує ретельного підбору гіперпараметрів [13].

Багатошаровий перцептрон є базовою архітектурою штучних нейронних мереж, що складається з вхідного шару, одного або декількох прихованих шарів і вихідного шару. Кожен нейрон виконує зважене сумування вхідних сигналів та передає результат через нелінійну функцію активації:

$$a_j^l = f(\sum w_{ij}^l \cdot a_i^{l-1} + b_j^l),$$

де w_{ij} – ваговий коефіцієнт, b_j – зміщення, f – функція активації. У задачах фізичної реабілітації вхідними параметрами для MLP є вік, стать, ІМТ, ЧСС, артеріальний тиск, результати функціональних проб, лабораторні показники та рівень больового синдрому. MLP здатний моделювати складні нелінійні залежності між цими показниками та оптимальною реабілітаційною програмою завдяки багатошаровій структурі. Для вихідного шару у задачах класифікації застосовується функція Softmax, що перетворює результати мережі у ймовірності належності до кожного класу реабілітаційних програм. MLP навчається методом зворотного поширення помилки в поєднанні з оптимізатором Adam [14].

Вибір функції активації у прихованих шарах MLP суттєво впливає на ефективність навчання. Функція ReLU (Rectified Linear Unit), визначена як $f(x) = \max(0, x)$, є найбільш поширеною у сучасних архітектурах завдяки обчислювальній простоті та відсутності проблеми затухаючих градієнтів у порівнянні з сигмоїдою та гіперболічним тангенсом. Для прихованих шарів у задачах реабілітаційної класифікації рекомендованою є саме ReLU або її варіант Leaky ReLU, що дозволяє уникнути “мертвих нейронів” при малих від’ємних значеннях входу. Для вихідного шару у задачах багатокласової класифікації застосовується виключно функція Softmax. Архітектура мережі (кількість прихованих шарів та нейронів у кожному шарі) підбирається емпірично або методом grid search у просторі гіперпараметрів. Для задачі підбору реабілітаційних програм на основі медичних показників вхідного виміру від 12 до 20 ознак типовою є архітектура з двома-трьома прихованими шарами розміром 64–256 нейронів. Надмірне ускладнення архітектури підвищує ризик перенавчання, що в умовах обмеженого обсягу медичних даних є особливо критичним. Для стабілізації навчання та прискорення збіжності рекомендується застосування Batch Normalization між шарами.

Вибір оптимальної моделі є одним із ключових питань даного дослідження. Порівняльний аналіз п'яти обраних алгоритмів проводиться за критеріями, що є найбільш релевантними для задачі підбору реабілітаційних програм: точністю

класифікації, здатністю до роботи з нелінійними залежностями, стійкістю до малих вибірок і гетерогенних даних, а також інтерпретованістю, що описано у таблиці 2.1.

Таблиця 2.1 – Порівняння моделей машинного навчання для задачі підбору реабілітаційних програм

<i>Модель</i>	<i>Здатність до нелінійності</i>	<i>Стійкість до малих вибірок</i>	<i>Інтерпретованість</i>	<i>Доцільність використання</i>
Логістична регресія	Низька (лінійна межа)	Висока	Висока (коефіцієнти)	Базова модель для порівняння
Дерево рішень / Random Forest	Висока (ансамбль дерев)	Середня–Висока	Середня (важливість ознак)	Нелінійні залежності, гетерогенні дані
Gradient Boosting (XGBoost)	Висока (послідовний ансамбль)	Середня	Середня	Найвища точність на структурованих даних
SVM з RBF-ядром	Висока (ядровий метод)	Висока	Низька	Невеликі вибірки з добре підібраними ознаками
Багатошаровий перцептрон	Висока (глибока мережа)	Низька–Середня	Низька (потребує ХАІ)	Складні взаємозв'язки, великі вибірки

Логістична регресія включається як базова модель порівняння, що дає змогу оцінити, наскільки виправдане застосування складніших алгоритмів. Random Forest

і Gradient Boosting є ключовими претендентами через поєднання високої точності та відносної інтерпретованості. SVM розглядається у контексті невеликих вибірок медичних даних. MLP тестується для оцінки потенціалу нейромережевого підходу для даної задачі.

2.3 Методи підготовки медичних даних для навчання моделей

Якість роботи моделей машинного навчання значною мірою залежить від правильності підготовки вхідних даних. У задачах фізичної реабілітації медичні показники характеризуються неоднорідністю та різними одиницями вимірювання. Нормалізація є обов'язковим етапом підготовки, оскільки різні показники можуть мати суттєво різні діапазони значень: вік пацієнта від 18 до 90 років, ЧСС від 50 до 180 уд/хв, ІМТ від 16 до 40.

Найпоширенішим методом є min-max нормалізація, яка приводить значення параметра до діапазону від 0 до 1. Як альтернатива застосовується Z-score стандартизація, при якій ознаки приводяться до нульового середнього та одиничного стандартного відхилення, що менш чутливо до викидів. Метод доцільний для антропометричних та більшості лабораторних параметрів. Для показників із суттєво скошеним розподілом застосовується логарифмічне перетворення.

Для категоріальних ознак (стать, тип захворювання, рівень активності) застосовуються методи кодування. Label encoding присвоює кожному класу числовий індекс, проте створює штучний порядковий зв'язок між категоріями. Метод one-hot encoding є більш надійним: кожна категорія перетворюється на окремий бінарний вектор, що усуває штучний порядок і забезпечує коректну роботу моделі. Для систем із великою кількістю категорій (повний перелік кодів МКХ) можуть використовуватись embedding-представлення.

Реальні клінічні дані майже завжди містять пропущені або неповні значення. У реабілітаційних системах пропуски виникають через неповне обстеження пацієнта, відсутність певних лабораторних аналізів або технічні помилки при введенні інформації.

Найпростішим методом є видалення записів із пропусками, проте він призводить до зменшення обсягу вибірки і втрати важливої інформації. Тому частіше застосовуються методи заповнення пропущених значень. Заміна середнім або медіаною є простим методом, однак не враховує індивідуальних особливостей пацієнтів. Складніші підходи – прогнозування пропущених значень на основі інших параметрів за допомогою k-NN або множинної імпутації (MICE) – дають більш точні оцінки для показників із механізмом пропуску MAR.

Важливим, але нерідко недооціненим кроком є валідація якості даних після їх збору та попередньої обробки. У медичних наборах даних систематично зустрічаються клінічно неможливі або підозрілі значення: наприклад, ЧСС у спокої 200 уд/хв або зріст 2,5 м. Такі значення, потрапляючи у навчальну вибірку, суттєво деформують модель. Для їх виявлення використовуються методи статистичного контролю: значення, що виходять за межу $\pm 3\sigma$ від середнього або за клінічно встановлені фізіологічні межі, підлягають перевірці та, за потреби, виключенню або заміщенню. Окрему загрозу становить дублювання записів – ситуація, коли один і той самий пацієнт представлений кількома ідентичними або майже ідентичними записами. У задачах медичної реабілітації дублювання може виникати при перенесенні даних між відділеннями або при повторних госпіталізаціях. Для виявлення дублікатів застосовуються методи нечіткого порівняння рядків і дедублікації за ключовими клінічними ідентифікаторами. Лише після завершення всіх кроків валідації та очищення набір даних вважається готовим до формування навчальної вибірки.

Для балансування вибірки при наявності дисбалансу класів застосовується метод SMOTE (Synthetic Minority Over-sampling Technique), що створює нові синтетичні приклади шляхом інтерполяції між сусідніми об'єктами менш

представленого класу. Метод зберігає інформацію вибірки та збільшує різноманітність прикладів рідкісних класів [15].

Навчальна вибірка є множиною медичних записів пацієнтів, де кожен запис включає вектор вхідних медичних показників та відповідну реабілітаційну програму як цільову змінну. Для забезпечення ефективного навчання вибірка повинна охоплювати різні категорії пацієнтів, ступені тяжкості, вікові групи та типи патологій.

Модель, навчена на незбалансованих даних, схильна до систематичної помилки на користь домінуючого класу, що є неприйнятним у клінічному контексті. Для оцінки дисбалансу використовується відношення кількості об'єктів найбільшого та найменшого класів. При суттєвому дисбалансі застосовується SMOTE або зважування класів у функції втрат.

Для правильного оцінювання якості моделі вибірка ділиться на навчальну, валідаційну та тестову частини у співвідношенні 70% – 15% – 15%. При цьому дані одного пацієнта не можуть одночасно потрапляти до навчальної та тестової вибірки. Для більш надійного оцінювання застосовується k-fold крос-валідація, де середній результат обчислюється по всіх k фолдах.

2.4 Методи навчання та оцінювання моделей

Вибір функції втрат визначає, наскільки ефективно модель зможе навчатися на медичних даних. Для задач класифікації реабілітаційних програм застосовується Categorical Cross-Entropy:

$$L = - \sum y_i \cdot \log(\hat{y}_i),$$

де y_i – справжнє значення (one-hot), $\hat{y}_i$ – передбачена ймовірність для класу i . Для задач регресії – прогнозування числових параметрів навантаження – застосовується середньоквадратична похибка (MSE) або середня абсолютна похибка (MAE), яка є менш чутливою до викидів.

Для нейромережевої моделі (MLP) як алгоритм оптимізації обрано Adam (Adaptive Moment Estimation), що поєднує momentum та адаптивну швидкість навчання. Для дерево-подібних алгоритмів (Random Forest, Gradient Boosting) навчання виконується за власними критеріями розбиття (Gini, ентропія, MSE). Для SVM мінімізується функція втрат hinge з регуляризацією.

Для порівняння моделей у даному дослідженні застосовуються єдині метрики оцінювання. Точність класифікації (Accuracy) визначає частку правильних прогнозів; проте вона може бути оманливою при дисбалансі класів. Метрика Precision показує, яка частка позитивних прогнозів є правильними, Recall – яку частку реальних позитивних випадків система змогла виявити. Recall є особливо важливим у медичних системах, де пропуск потрібного класу програми може мати серйозні наслідки. F1-score є гармонійним середнім Precision та Recall, що дає збалансовану оцінку якості при нерівномірному розподілі класів. Площа під ROC-кривою (AUC-ROC) показує здатність моделі розділяти класи незалежно від порогу класифікації.

Окрім математичних метрик, у медичних задачах критичними є клінічні критерії ефективності: безпечність рекомендацій та відповідність клінічним протоколам. Результати порівняння всіх моделей за переліченими метриками на тестовій вибірці є основою для вибору оптимального рішення.

Перенавчання є суттєвою проблемою для всіх розглянутих моделей в умовах обмеженого обсягу медичних даних. Основна ознака перенавчання – зменшення похибки на навчальній вибірці при одночасному зростанні похибки на валідаційній.

Для нейромережевої моделі (MLP) застосовуються L2-регуляризація (штраф за великі вагові коефіцієнти), Dropout (випадкове вимкнення нейронів під час навчання) та Early Stopping (зупинка навчання при припиненні покращення на

валідаційній вибірці). Для Random Forest перенавчання обмежується природно завдяки механізму усереднення по деревах. Для Gradient Boosting застосовується регуляризація через параметри `max_depth`, `min_samples_leaf` та `learning rate`. Для SVM рівень регуляризації контролюється параметром `C`.

Крос-валідація є універсальним інструментом об'єктивного оцінювання узагальнюючої здатності для всіх моделей та зменшує залежність результатів від конкретного поділу даних. У медичній реабілітації з обмеженим обсягом даних також доцільна аугментація навчальних даних через SMOTE та варіації числових показників у межах фізіологічних діапазонів.

2.5 Обґрунтування програмних засобів реалізації

Для реалізації та порівняння моделей машинного навчання обрано сукупність програмних засобів, що забезпечують ефективну обробку медичних даних, навчання моделей та аналіз результатів.

Основною мовою програмування обрано Python, що є де-факто стандартом у сфері машинного навчання та медичних інформаційних систем. Python забезпечує простий синтаксис, велику кількість бібліотек для машинного навчання та хорошу інтеграцію з базами даних.

Для реалізації Random Forest, Gradient Boosting, SVM та логістичної регресії обрано бібліотеку Scikit-learn, яка надає уніфікований інтерфейс для навчання та оцінювання різних моделей, засоби нормалізації, кодування категоріальних ознак, балансування вибірки (SMOTE через `imbalanced-learn`), крос-валідації та розрахунку метрик якості. Для реалізації моделі Gradient Boosting з розширеними можливостями регуляризації використовуються бібліотеки XGBoost та LightGBM. Для реалізації нейромережевої моделі MLP обрано TensorFlow у поєднанні з Keras, що спрощує побудову архітектури та забезпечує підтримку GPU (Graphics Processing Unit) [16].

Для обробки медичних даних використовуються бібліотеки NumPy та Pandas. NumPy забезпечує ефективну роботу з багатовимірними масивами та швидкі математичні обчислення, Pandas – роботу з табличними медичними даними, електронними картками та результатами лабораторних досліджень. Для візуалізації результатів навчання та порівняння моделей використовуються бібліотеки Matplotlib та Seaborn. Для забезпечення інтерпретованості результатів доцільно використовувати бібліотеку SHAP, що дозволяє оцінити внесок кожного медичного показника у формування рекомендації та підвищити довіру медичного персоналу до системи.

Висновки до розділу 2

У другому розділі проведено комплексний аналіз методів машинного навчання для задачі підбору реабілітаційних програм та обґрунтовано методологічний підхід до порівняльного дослідження.

Встановлено, що задачі машинного навчання у сфері фізичної реабілітації характеризуються багатофакторністю, нелінійним характером медичних залежностей, неоднорідністю даних та обмеженим обсягом вибірок. Показано, що лінійні методи мають системні обмеження в описі нелінійних клінічних залежностей, що обґрунтовує доцільність порівняння складніших моделей.

Для порівняльного дослідження обрано п'ять моделей: логістична регресія (базова модель), Random Forest, Gradient Boosting (XGBoost/LightGBM), SVM з RBF-ядром та багатошаровий перцептрон (MLP). Вибір моделей забезпечує охоплення різних класів алгоритмів – від лінійних до ансамблевих та нейромережових, що дозволяє провести об'єктивний порівняльний аналіз.

Описано методи підготовки медичних даних: нормалізацію (min-max, Z-score), кодування категоріальних ознак (one-hot encoding), обробку пропущених значень (медіана, MICE) та балансування класів методом SMOTE. Проаналізовано

методи оцінювання моделей, включаючи метрики Accuracy, Precision, Recall, F1-score та AUC-ROC, а також методи запобігання перенавчанню для кожного з алгоритмів.

Обґрунтовано програмні засоби реалізації: Python, Scikit-learn, XGBoost, LightGBM, TensorFlow/Keras, NumPy, Pandas та SHAP. Отримані результати закладають методологічну основу для практичної реалізації та порівняльного оцінювання моделей, що розглядається у наступному розділі.

РОЗДІЛ 3 ПРОГРАМНА РЕАЛІЗАЦІЯ ТА АНАЛІЗ РЕЗУЛЬТАТІВ

3.1 Опис та характеристики датасету

3.1.1 Джерела даних та методологія формування датасету

Однією з принципових умов побудови ефективної моделі машинного навчання для підбору реабілітаційних програм є наявність репрезентативного та достатньо великого набору даних, що охоплює широкий спектр нозологій, вікових груп і клінічних профілів пацієнтів. Водночас публічно доступні реабілітаційні бази даних мають суттєву обмеженість: вони, як правило, зосереджені на одній нозологічній групі, містять вузький набір клінічних показників або не мають розмітки реабілітаційних втручань як цільових змінних. З огляду на це в рамках даного дослідження було сформовано інтегрований датасет, що поєднує та уніфікує дані з кількох гетерогенних відкритих джерел.

Первинною базою для формування неврологічного та кардіологічного сегментів датасету слугувала MIMIC-IV (Medical Information Mart for Intensive Care, версія 4) – деідентифікована база електронних медичних записів пацієнтів клінічного центру Beth Israel Deaconess Medical Center, що містить демографічні дані, результати лабораторних досліджень, показники гемодинаміки та інформацію про діагнози у структурованому форматі. З MIMIC-IV були відібрані записи пацієнтів, які проходили реабілітаційне лікування або відповідну фізіотерапію після неврологічних та серцево-судинних епізодів, зокрема інсульту, інфаркту міокарда та стану після кардіохірургічних втручань [17].

Для розширення нейрореабілітаційного компонента датасету використовувалась відкрита база даних CARDS (Centralized Open-Access Rehabilitation Database for Stroke), описана у роботі Barnard et al., опублікованій у *Frontiers in Neurology*. Ця база акумулює дані кількох незалежних клінічних досліджень із реабілітації після інсульту та включає стандартизовані функціональні оцінки за шкалою FIM, індексом Бартел, показники рухової активності та профілі відновлення пацієнтів із різним ступенем неврологічного дефіциту. Призначення

та рекомендації реабілітологів, зафіксовані у цій базі, слугували основою для формування частини цільових міток нейрореабілітаційного підмасиву [18].

Ортопедичний та опорно-руховий сегмент датасету сформовано на основі даних дослідження Tack et al., опублікованого у Data in Brief (PMC7110331), яке надає клінічні записи пацієнтів із захворюваннями опорно-рухового апарату різної етіології: дегенеративними змінами хребта, артрозами суглобів, станами після ортопедичних операцій, травматичними ураженнями. Датасет охоплює функціональні показники, дані болю за шкалою ВАШ та інформацію про застосовані реабілітаційні методики [19].

Для забезпечення охоплення загальнотерапевтичного профілю та метаболічних патологій залучено два публічні набори даних: датасет Physical Therapy Exercises Dataset, що містить клінічні та функціональні показники пацієнтів фізіотерапевтичного профілю з рекомендованими типами вправ, а також Heart Disease Health Indicators Dataset на базі обстеження BRFSS, що забезпечив серцево-судинний контекст для формування обмежувальних рекомендацій при кардіологічних ризиках [20] [21].

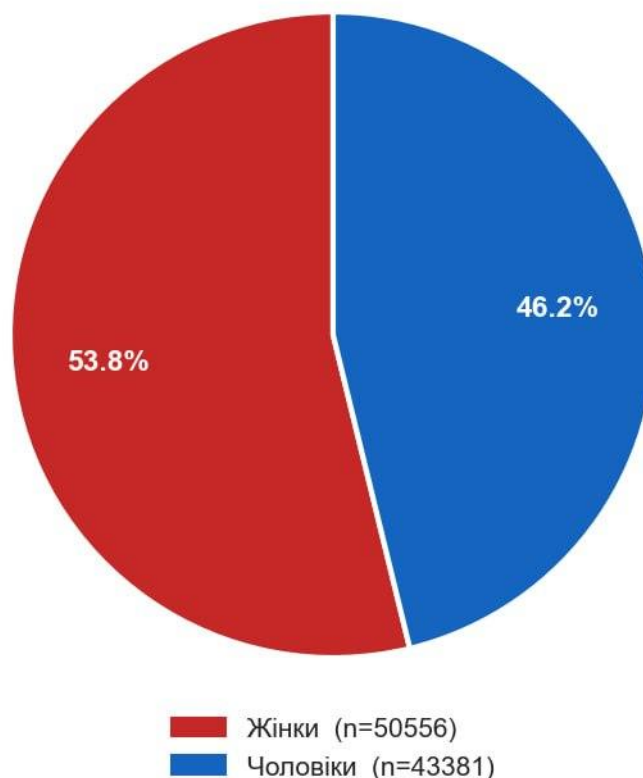
Оскільки жодне з перелічених джерел не містило в готовому вигляді мультинозологічного датасету з одночасною розміткою за десятками цільових реабілітаційних змінних, обов'язковим етапом стала глибока інтеграція та трансформація даних. На першому кроці здійснювалась нозологічна уніфікація – приведення діагностичних кодів із різних джерел до єдиної номенклатури, що відображає клінічно значущі нозологічні категорії в контексті фізичної реабілітації. На другому кроці відбувалась уніфікація числових параметрів: різнорідні формати клінічних показників приводились до єдиної схеми іменування та фізіологічно допустимих діапазонів, значення за межами яких виключались як артефактні. На третьому, ключовому, кроці виконувалось формування цільових змінних: там, де первинні записи містили явні призначення реабілітологів, вони переносились безпосередньо; для записів без розмітки цільові мітки формувались на основі стандартизованих клінічних алгоритмів відповідно до чинних реабілітаційних протоколів. На завершальному кроці інтегрований масив проходив процедуру

дедублікації та контроль якості. Результатом описаного процесу є датасет, що налічує 93 937 записів і є репрезентативним за широким спектром нозологій та клінічних профілів фізичної реабілітації.

3.1.2 Структура та склад даних

Сформований набір даних має табличну структуру і складається зі 180 стовпців, розподілених за двома функціональними групами: ознаки, що описують клінічний стан пацієнта, та цільові змінні, які відображають реабілітаційні рішення лікаря. Загальний обсяг вибірки становить 93 937 записів.

Гендерний розподіл вибірки є відносно рівномірним: жінки становлять 53,8% когорти ($n = 50\,556$), чоловіки – 46,2% ($n = 43\,381$), що відповідає загальній епідеміологічній закономірності більшої поширеності звернень жінок за реабілітаційною допомогою, зокрема через більшу частоту дегенеративно-дистрофічних захворювань хребта та остеоартрозу в жіночій популяції середнього і похилого віку. Детальніше зображено на рисунку 3.1.

Розподіл пацієнтів за статтю**Рисунок 3.1 – Гендерний розподіл вибірки**

Розподіл за віком має виражену унімодальну форму з піком у категорії 55–60 років та поступовим спаданням у напрямку молодшого і похилого віку, що продемонстровано на рисунку 3.2. Середній вік становить приблизно 55 років, що є закономірним для реабілітаційного контингенту: основні нозології, представлені в даних, – дегенеративні захворювання хребта, артрози, серцево-судинна патологія – мають пікові прояви та найбільшу частоту звернень саме у п'ятому-шостому десятилітті життя.

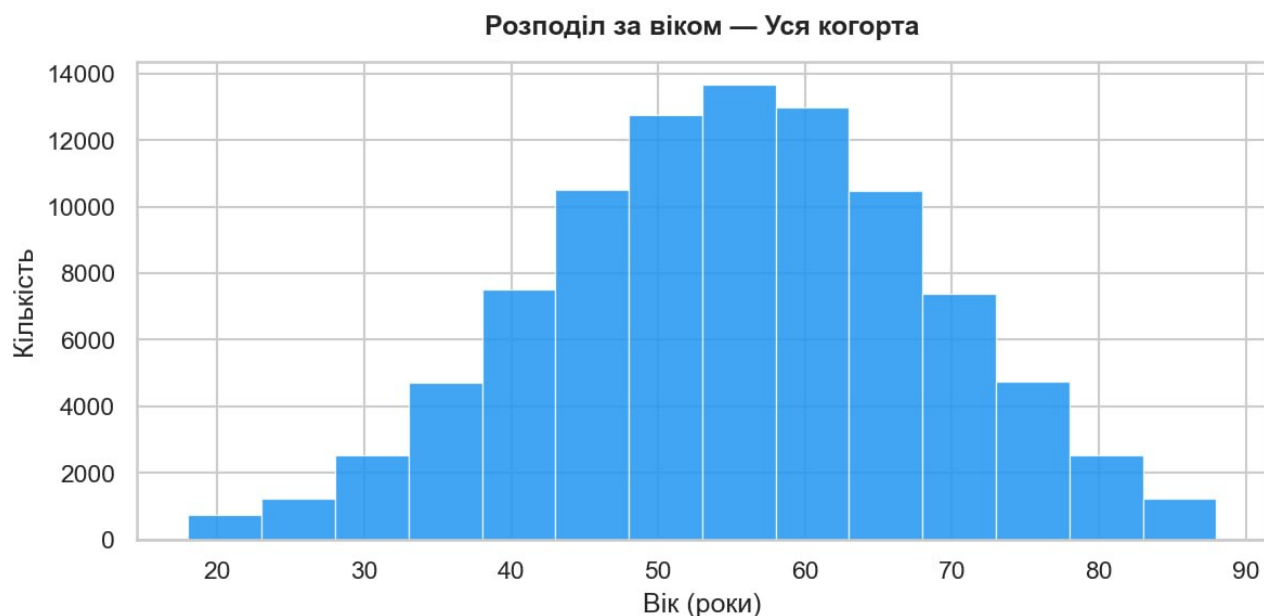


Рисунок 3.2 – Розподіл пацієнтів за віком

Розподіл індексу маси тіла концентрується в межах 20–30 кг/м², тобто на межі норми та надлишкової ваги, з помітним «хвостом» у бік ожиріння; у чоловіків спостерігається незначне зміщення у бік вищих значень порівняно з жінками, ознайомитись з розподілом можна на рисунку 3.3. Такий розподіл є характерним для реабілітаційних когорт, оскільки надлишкова маса тіла та ожиріння є незалежними факторами ризику захворювань опорно-рухового апарату та серцево-судинної системи.

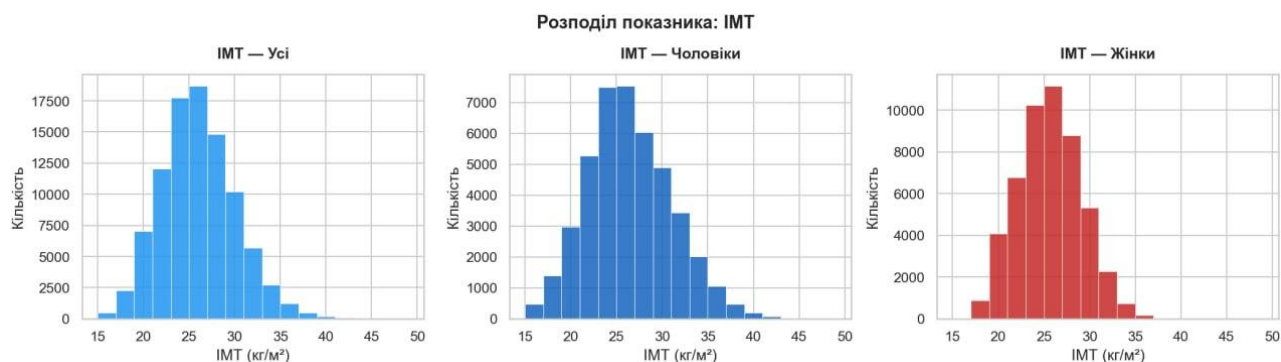


Рисунок 3.3 – Розподіл індексу маси тіла (ІМТ)

Розподіл систолічного артеріального тиску сконцентрований у межах 110–145 мм рт. ст. з піком близько 125–130 мм рт. ст., що відповідає змішаній популяції з нормотонією та артеріальною гіпертензією I–II ступеня. Діастолічний тиск розподілений більш вузько, з піком у 70–80 мм рт. ст. У чоловіків дещо більший «хвіст» у зоні підвищеного систолічного тиску – понад 150 мм рт. ст. – що узгоджується з вищою поширеністю артеріальної гіпертензії у чоловіків середнього віку.

Аналіз частоти діагнозів у топ-20 найпоширеніших нозологій демонструє очікувану клінічну структуру: перші позиції займають остеохондроз поперекового відділу хребта (64,9%) та шийного відділу (55,2%), далі йдуть гіпертонічна хвороба (40,0%), варикозне розширення вен (29,9%) та ішемічна хвороба серця (24,8%), частоти найбільш поширених діагнозів зображено на рисунку 3.4. Ця ієрархія є медично очікуваною: дегенеративні захворювання хребта та серцево-судинна патологія є найпоширенішими коморбідними станами серед пацієнтів реабілітаційного профілю середнього і похилого віку, а їхнє поєднання утворює найбільш складні та багатокомпонентні реабілітаційні програми. Окремо зафіксовано наявність імплантованих конструкцій – кардіостимулятора, ендопротезів суглобів, металоконструкцій хребта – та ампутацій різного рівня, оскільки вони безпосередньо визначають протипоказання та технічні обмеження реабілітаційної програми.

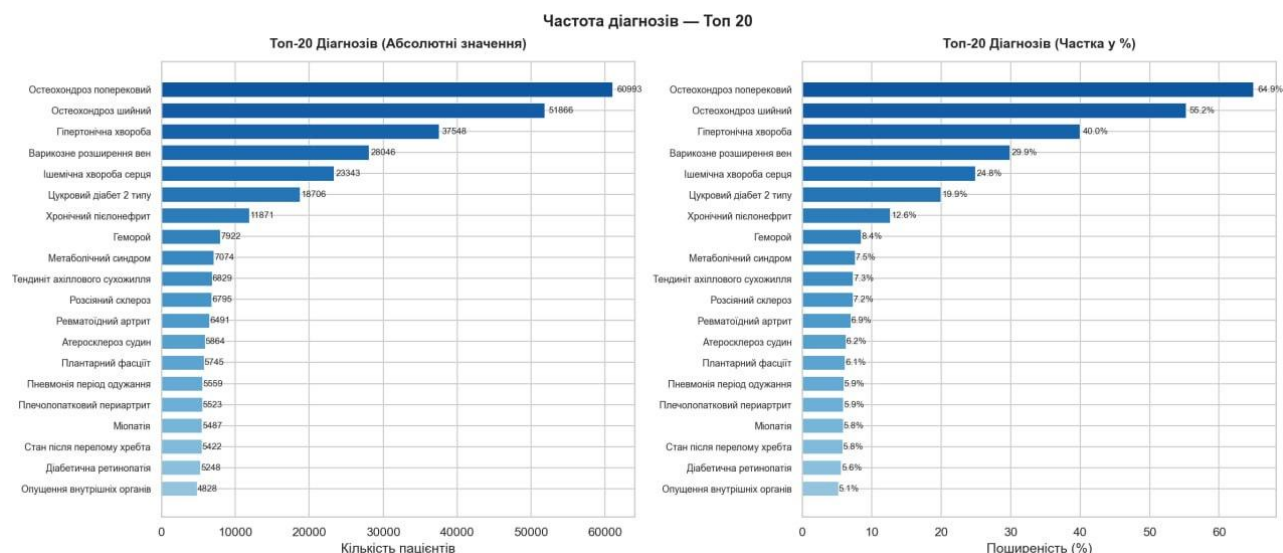


Рисунок 3.4 – Топ-20 найпоширеніших діагнозів у вибірці

Клінічні кількісні показники охоплюють 36 числових змінних: ступінь розриву зв'язок і ротаторної манжети, атрофії квадрицепса, контрактур суглобів, зносу хряща, дефіциту м'язової маси, зниження фракції викиду лівого шлуночка, дефіциту ЖЄЛ, зниження склепіння стопи та кута Кобба; показники болю за шкалою ВАШ у дев'яти анатомічних зонах; лабораторні показники – гемоглобін, лейкоцити, тромбоцити, ШОЕ, С-реактивний білок, рівень глюкози; ступінь парезу кінцівок у балах та кількість місяців після останнього хірургічного втручання.

Цільові змінні налічують 44 бінарні ознаки двох функціональних підтипів. Перший підтип – рекомендовані вправи та реабілітаційні компоненти (15 змінних): дихальна динамічна та статична гімнастика, діафрагмальне та дренажне дихання, звукова гімнастика, вправи на розслаблення, ізометрика, тракція хребта, коригувальні асиметричні вправи, вправи для дрібних і великих м'язових груп, зміцнення м'язового корсету спини, вправи для м'язів склепіння стопи, координація та рівновага, ідеомоторні вправи, теренкур. Другий підтип – клінічні обмеження та протипоказання (29 змінних): обмеження осьового навантаження, виключення ударних вправ і конкретних положень тіла, обмеження при критичному рівні глюкози, кардіологічні та офтальмологічні протипоказання, обмеження при

нестабільності хребта, виключення активної реабілітації при гострому запаленні та ряд інших спеціальних обмежень безпеки.

3.1.3 Аналіз розподілу цільових змінних та дисбаланс класів

Фундаментальною характеристикою бази даних, що безпосередньо впливає на вибір стратегії навчання моделей, є значний і нерівномірний дисбаланс класів за цільовими змінними. Оскільки кожна цільова змінна є бінарною ознакою наявності або відсутності певної рекомендації, ступінь дисбалансу визначається часткою позитивного класу у загальній вибірці.

У середньому кожен пацієнт має від 10 до 20 активних правил із 44 можливих, детальніше ознайомитись можна на рисунку 3.5. Розподіл рекомендацій на пацієнта є більш симетричним і концентрується в діапазоні 4–7 активних призначень, тоді як розподіл обмежень є правосторонньо асиметричним із піком у 7–10 активних протипоказань. Це відображає реальну клінічну картину: пацієнти зі складним коморбідним профілем накопичують значну кількість обмежень, тоді як активний реабілітаційний компонент програми є більш стандартизованим за кількістю елементів.

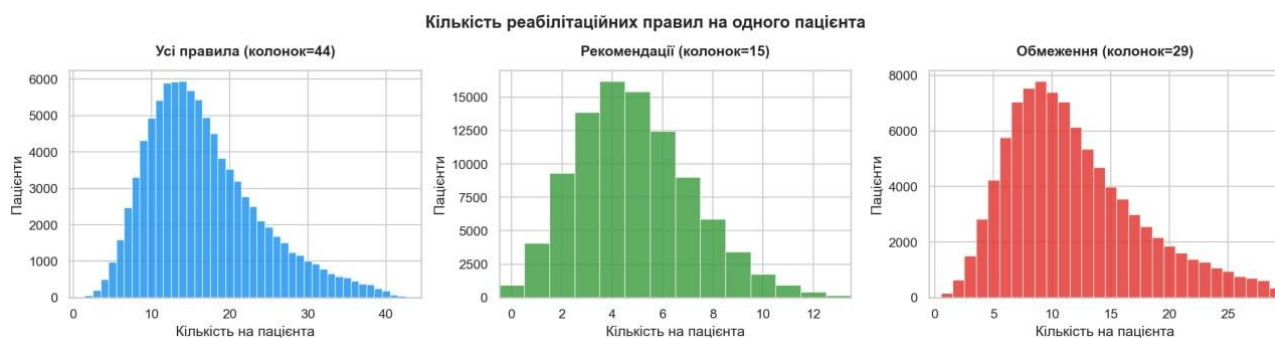


Рисунок 3.5 – Кількість активних реабілітаційних правил, рекомендацій та обмежень на одного пацієнта

Аналіз частоти позитивного класу за всіма 44 цільовими змінними виявив суттєву варіативність: від найбільш частотних – виключення активної реабілітації при гострому запаленні (близько 70%) та обмеження гострого болю без активних рухів (близько 62%) – до рідкісних спеціальних обмежень із часткою, що не перевищує 0,8–2%. Середня частота позитивного класу за всіма 44 цільовими показниками становить 20,1%, що означає: кожна конкретна реабілітаційна рекомендація або обмеження є актуальним лише для кожного п'ятого пацієнта у вибірці.

Кореляційна матриця між 94 діагнозами демонструє виражену блокову структуру: у межах функціональних підгруп – вертебродіагностичні, кардіологічні, метаболічні, неврологічні – спостерігаються помірні та сильні позитивні кореляції, тоді як між підгрупами різного профілю кореляція є слабкою або відсутньою, детальніше на рисунку 3.6. Ця структура підтверджує, що патологічна приналежність пацієнта визначає цілий кластер взаємопов'язаних діагнозів, що у свою чергу формує відповідний кластер реабілітаційних рішень.

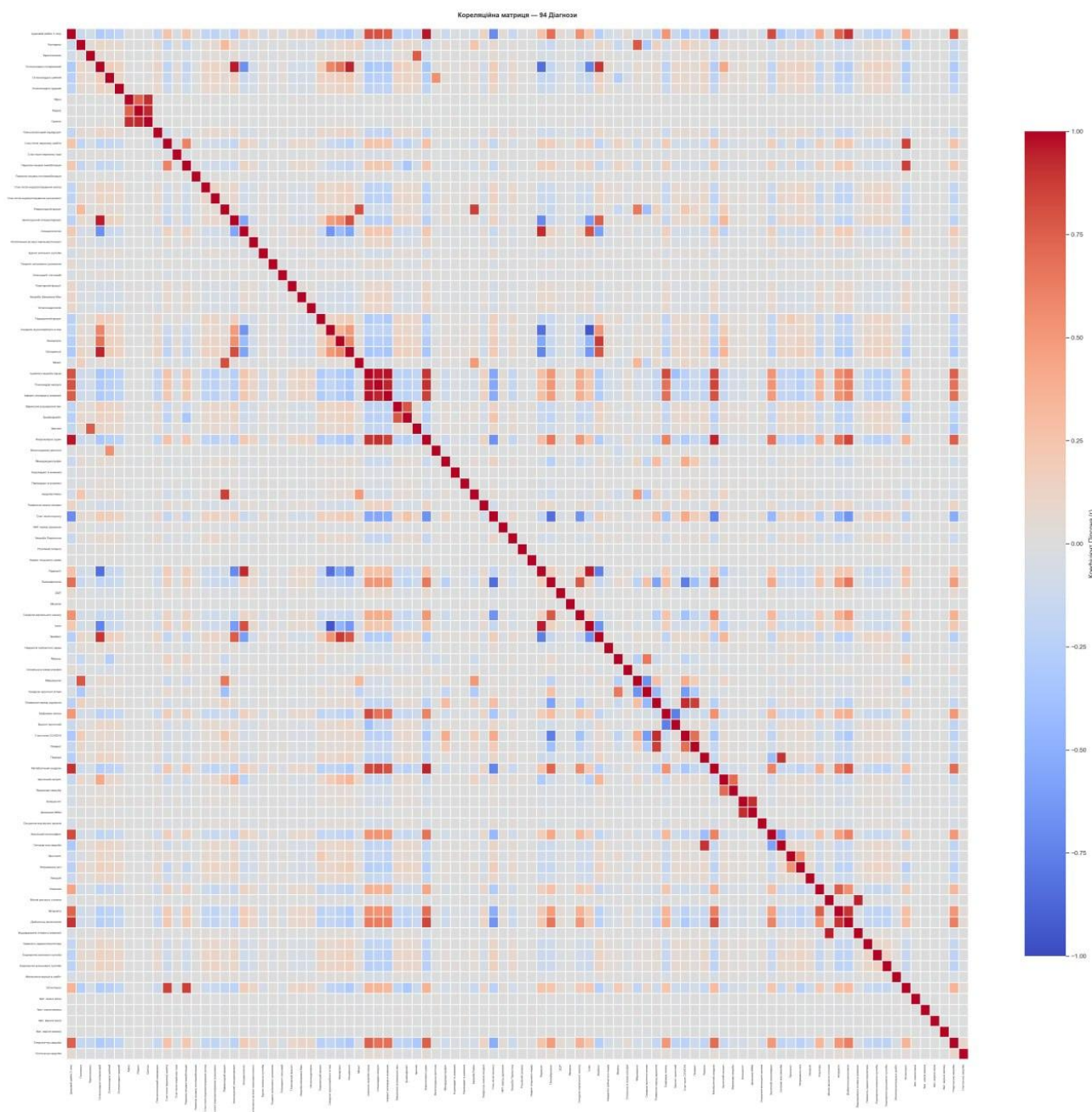


Рисунок 3.6 – Кореляційна матриця для 94 діагнозів

Сформулюємо наслідки для стратегії моделювання. Наявність дисбалансу є принциповою методологічною проблемою, оскільки модель без спеціальних механізмів корекції систематично ухиляється в бік домінуючого класу, фактично ігноруючи рідкісні, але клінічно критичні рекомендації. Для нейтралізації цього ефекту при навчанні кожної з моделей застосовувалось зважування класів — надання підвищеної штрафної ваги помилкам на рідкісному позитивному класі. Як

наслідок, метрика точності (Assurasy) сама по собі є недостатньо інформативною: модель, що завжди передбачає відсутність рекомендації, матиме формально високу точність на цільових змінних із часткою позитивного класу 2–5%. Тому основними показниками порівняння обрано ROC-AUC та F1-Score, які явно враховують дисбаланс класів.

3.2 Передобробка даних та підготовка навчальних вибірок

3.2.1 Нормалізація та кодування ознак

Перш ніж перейти до навчання будь-яких моделей, необхідно привести набір даних до стану, придатного для числової обробки. Вихідний набір містить 136 предикторів та 44 цільові змінні при загальному обсязі 93 937 записів. Ознаки є принципово різнорідними за своєю природою – серед них є текстові категорії, бінарні індикатори та неперервні числа в різних шкалах і одиницях вимірювання, – що унеможлиблює застосування єдиної уніфікованої стратегії обробки до всієї бази даних. Саме тому розроблено диференційований підхід, де тип перетворення визначається семантикою кожної групи ознак.

Усі 44 цільові колонки є бінарними й представляють медичні рішення: призначення конкретних вправ, встановлення обмежень або визначення фокусу реабілітаційної програми. На підготовчому етапі ці змінні приводяться до цілочисельного типу даних. Будь-яке масштабування чи нормалізація міток класів є методологічно неприпустимим, оскільки спотворює їхню дискретну семантику і перетворює задачу класифікації на задачу регресії.

Найчисленнішу групу предикторів – 99 зі 136 – становлять бінарні показники наявності або відсутності певних клінічних станів. До цієї групи належать понад 80 діагностичних ознак (наявність конкретного захворювання: цукровий діабет, різні форми остеохондрозу, артрити, серцево-судинної патології тощо), а також індикатори ампутації різних сегментів кінцівок та перенесених оперативних

втручань. Усі ці ознаки вже природно перебувають у бінарній шкалі, тому будь-яке їхнє масштабування є не лише зайвим, але й шкідливим: після стандартизації ознака втрачає свою медичну інтерпретованість.

У датасеті присутня одна категоріальна текстова ознака – руховий режим пацієнта. Вона набуває чотири взаємовиключні значення: постільний, палатний, вільний та тренувальний. Ці режими відображають дозволений рівень фізичної активності і є одним із ключових чинників при визначенні реабілітаційного плану. Розподіл записів за режимами є нерівномірним: вільний режим домінує у вибірці (38,8%), далі йдуть палатний (28,8%), постільний (18,0%) та тренувальний (14,3%), що відображає реальну структуру реабілітаційного контингенту, детальніше на рисунку 3.7.

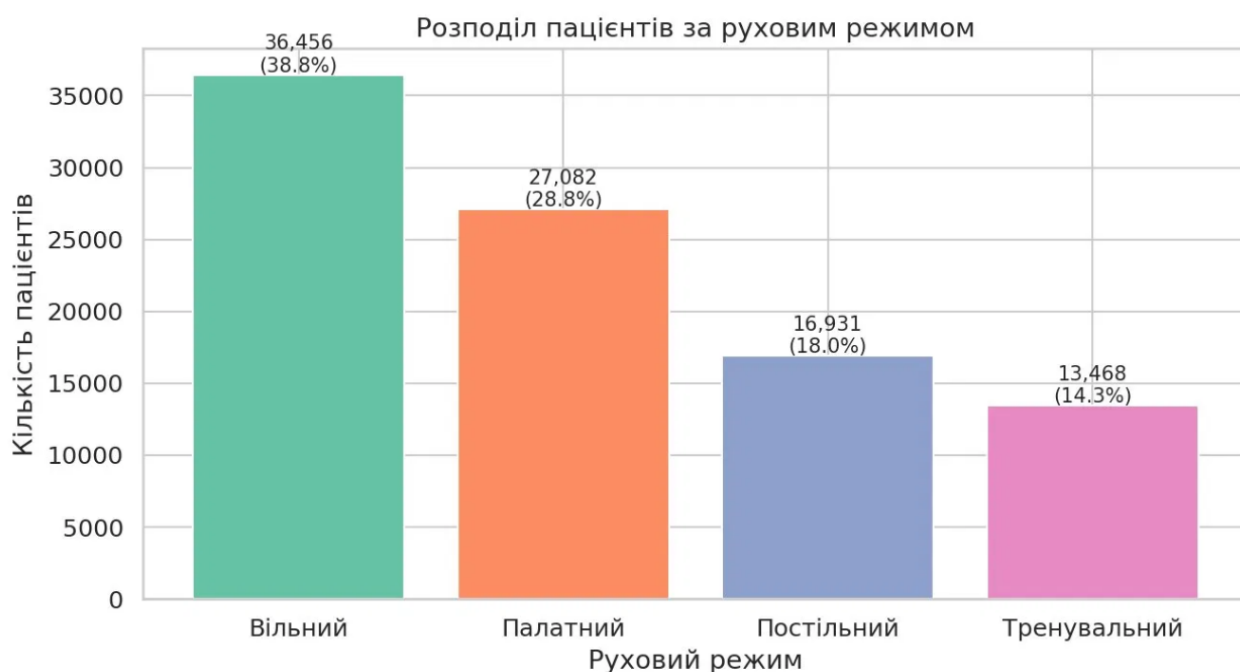


Рисунок 3.7 – Розподіл пацієнтів за руховим режимом

Для кодування цієї ознаки застосовано метод One-Hot Encoding з відкиданням першої категорії для уникнення мультиколінеарності. Такий підхід є стандартним для номінальних категоріальних ознак, між значеннями яких немає природного числового порядку. Порядкове кодування (наприклад, 0, 1, 2, 3) було б

методологічно хибним, оскільки імплікувало б метричні відстані між режимами та штучну ієрархію між ними.

До групи неперервних ознак належать 36 кількісних показників, що охоплюють кілька клінічних доменів: антропометричні показники (вік, зріст, вага, індекс маси тіла), гемодинамічні параметри (сistolічний і діастолічний артеріальний тиск), лабораторні показники (рівень глюкози крові, гемоглобін, лейкоцити, тромбоцити, швидкість осідання еритроцитів, С-реактивний білок), функціональні дефіцити (ступінь атрофії м'язів, знос хряща, кут Кобба, зниження фракції викиду лівого шлуночка, дефіцит м'язової маси, ступінь контрактури), показники больового синдрому за дев'ятьма локалізаціями за візуально-аналоговою шкалою, а також операційний анамнез і неврологічний статус.

Ці ознаки вимірюються в принципово різних одиницях і мають несумісні між собою шкали. Без вирівнювання масштабів дистанційно-чутливі алгоритми надавали б необґрунтовано велику вагу ознакам із ширшим числовим діапазоном. Для усунення цього ефекту застосовується стандартизація методом z-score: кожна ознака центрується відніманням середнього по вибірці та нормується на одиничне стандартне відхилення. Параметри стандартизації обчислюються виключно на навчальних даних і потім застосовуються до тестових, що запобігає «витіканню» інформації з тестової вибірки у навчальний процес.

Внаслідок застосування One-Hot Encoding до рухового режиму початкові 136 предикторів розширюються приблизно до 139 колонок. Підготовлений набір даних зберігається як окремий файл і слугує вхідними даними для всіх подальших етапів моделювання.

3.2.2 Динамічний відбір ознак для кожної цільової змінної

Задача, що розв'язується в цій роботі, є багатозадачною: необхідно водночас побудувати 44 незалежних класифікатори, кожен з яких прогнозує одне конкретне

реабілітаційне рішення. Найпростіший підхід передбачав би навчання кожної моделі на одному й тому самому наборі всіх 142 ознак. Проте такий підхід має суттєві вади. По-перше, він ігнорує те, що різні клінічні рішення зумовлені принципово різними наборами факторів. По-друге, включення нерелевантних ознак вносить шум у навчальні дані, підвищує ризик перенавчання і знижує загальну здатність моделей до узагальнення, особливо при вираженому дисбалансі класів. По-третє, зайва розмірність суттєво збільшує обчислювальні витрати при навчанні ансамблів і нейронних мереж.

Для подолання цих обмежень розроблено алгоритм динамічного відбору ознак, де для кожної цілі обирається власний, індивідуально підібраний набір найбільш інформативних предикторів.

На першому кроці для кожного з 44 цілей незалежно навчається ансамблева модель на основі надзвичайно рандомізованих дерев. Цей алгоритм обрано як інструмент оцінки важливості ознак, що добре зарекомендував себе в задачах із дисбалансом класів та неоднорідними типами предикторів. На другому кроці всі 139 ознак ранжуються за спаданням їхньої важливості, обчисленої як середнє зменшення домішки Джині в усіх деревах ансамблю. Далі будується кумулятивна сума нормованих важливостей, і відбирається мінімальна кількість ознак, що в сукупності покривають 95% сумарної пояснювальної здатності. Поріг у 95% є компромісом між повнотою інформації та боротьбою з надлишковою розмірністю. Нижня межа відбору встановлена на рівні 5 ознак – щоб уникнути ситуацій, коли один потужний предиктор «поглинає» майже всю важливість, залишаючи модель без достатнього контексту. Результат для кожної цілі – іменований список відібраних предикторів – зберігається у файлі конфігурації для подальшого відтворюваного використання.

Кількість відібраних ознак за 44 цілями змінюється від 5 до 109. Розподіл цих значень ілюструє рисунок 3.8.

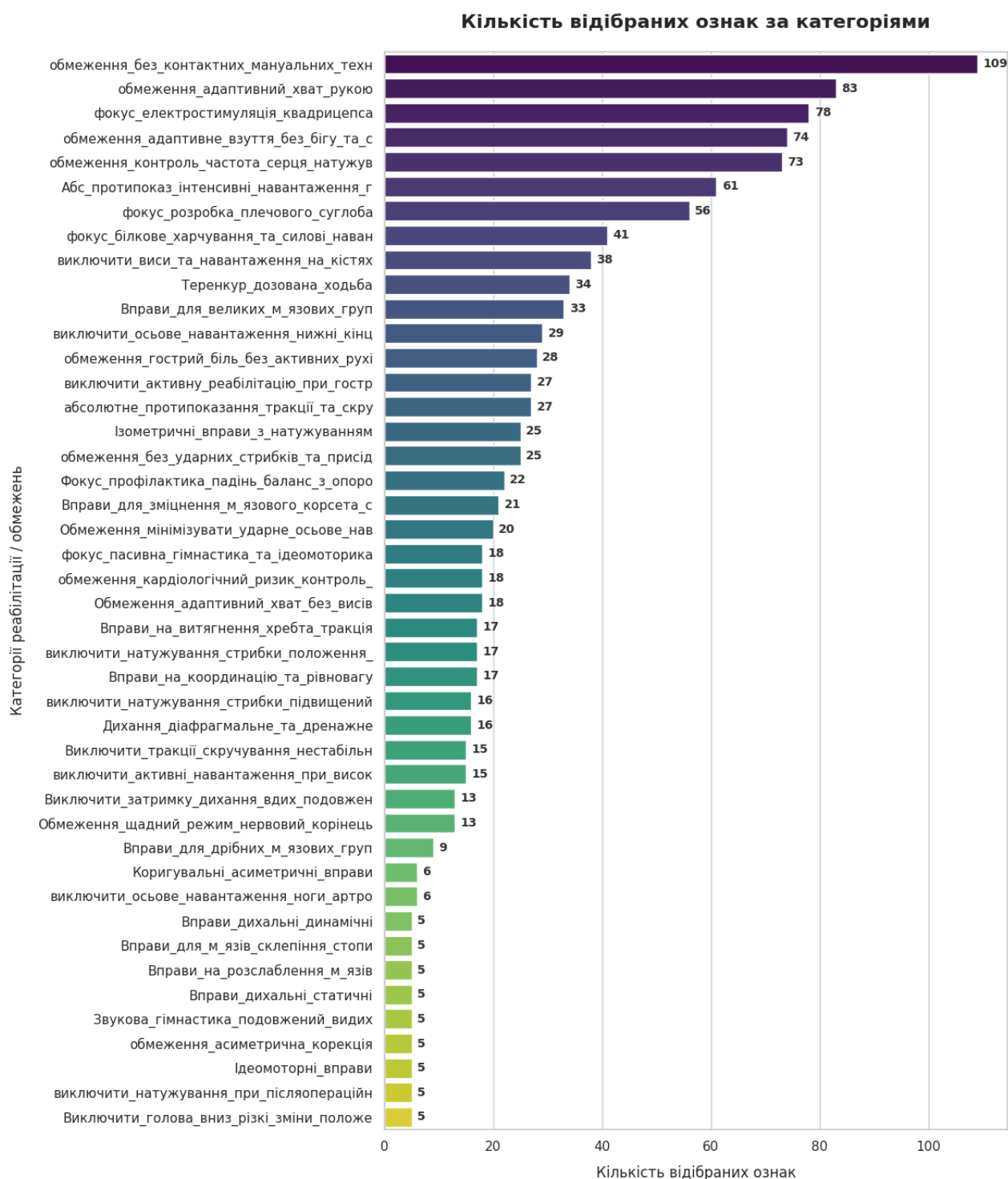


Рисунок 3.8 – Кількість відібраних ознак для кожної з 44 цільових змінних

Широкий діапазон відібраних ознак є не технічним недоліком, а безпосереднім відображенням медичної реальності та різної природи прогнозованих рішень. В одному кінці спектра знаходяться вузькоспецифічні призначення – наприклад, рішення про вправи для зміцнення склепіння стопи

практично повністю визначається наявністю плоскостопості або плантарного фасціїту, рівнем болю у відповідній локалізації та руховим режимом пацієнта. Для таких цілей достатньо п'яти-десяти ознак. На протилежному кінці спектра – системні протипоказання та складні обмеження, як-от кардіологічний ризик або заборона інтенсивних навантажень при гіпоксії. Ці рішення зумовлені поліморбідним клінічним портретом пацієнта, і для них алгоритм закономірно відбирає значно більшу кількість предикторів – до 109. Таким чином, кількість відібраних ознак є непрямым показником клінічної складності кожного реабілітаційного рішення.

3.2.3 Формування навчальної та тестової вибірок

Після того, як для кожної цілі визначено індивідуальний набір ознак, здійснюється розбиття даних на навчальну та тестову вибірки. Підхід «одна ціль – один піддатасет» передбачає, що для кожної з 44 цільових змінних формується власний піддатасет: з повної матриці вибираються лише ті предикатори, які були відібрані для цієї конкретної цілі на попередньому кроці.

Кожен піддатасет ділиться у пропорції 80% на навчальну вибірку та 20% на тестову. При загальному обсязі 93 937 записів це забезпечує приблизно 75 150 зразків для навчання та близько 18 790 зразків для незалежного тестування.

Ключовою особливістю є обов'язкове збереження пропорції класів у кожній із двох підвбірок. Необхідність цього прийому пояснюється вираженням та неоднорідним дисбалансом класів між різними цілями. Частка позитивного класу змінюється від 0,8% (адаптивний захват рукою) до 69,7% (виключення активної реабілітації при гострому запаленні), при цьому середня частота позитивного класу за всіма 44 цілями становить 20,1% , детальніше на рисунку 3.9. Без стратифікації при випадковому розбитті рідкісний позитивний клас може потрапити до тестової

вибірки у кількості, недостатній для репрезентативної оцінки здатності моделі його виявляти.

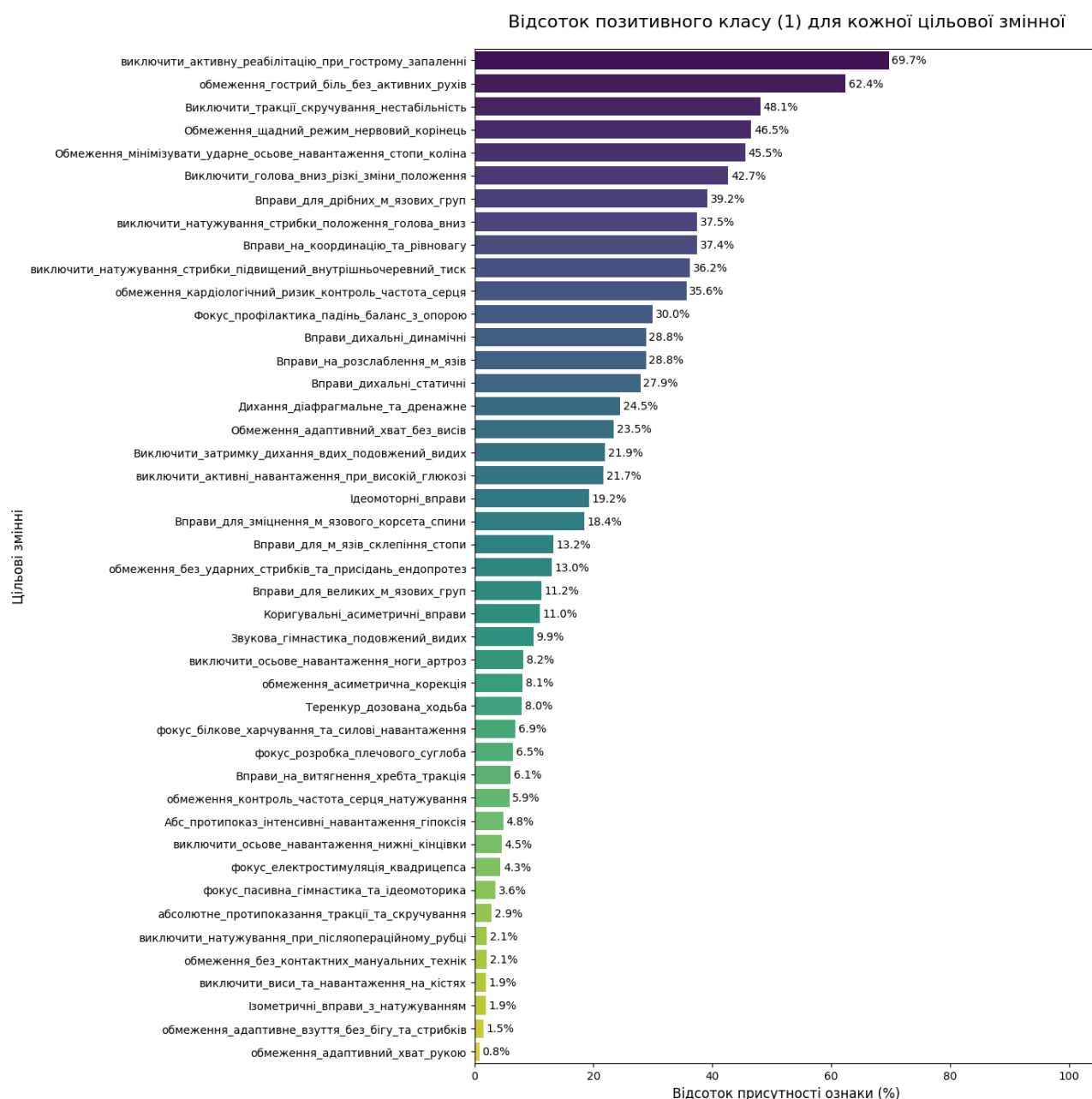


Рисунок 3.9 – Частота позитивного класу за всіма 44 цільовими змінними

Для забезпечення повної відтворюваності результатів у всіх операціях розбиття використовується фіксоване значення генератора випадкових чисел. Це дозволяє будь-якому досліднику точно відтворити ті самі навчальні та тестові

підмножини при повторному запуску пайплайну, що є важливою вимогою до відтворюваності наукових результатів.

3.3 Навчання моделей-експертів

3.3.1 Архітектура ансамблю моделей-експертів

Центральною архітектурною ідеєю системи є принцип спеціалізації: замість єдиної універсальної моделі, яка намагається одночасно вирішити всі 44 клінічні задачі, будується ансамбль із 44 незалежних моделей-експертів, де кожен експерт повністю зосереджений на одній цільовій змінній. Такий підхід відображає логіку клінічного мислення, де рішення про призначення конкретного виду вправ чи встановлення певного обмеження ухвалюється на основі специфічного для цього рішення набору клінічних даних.

Кожна модель-експерт навчається ізольовано: вона отримує власний піддасет зі своїм набором відібраних ознак, власну навчальну та тестову вибірку і повністю незалежно від решти 43 експертів оптимізує свої параметри. Взаємодія між експертами відсутня на рівні навчання – вони об'єднуються лише на стадії формування прогнозу, коли для нового пацієнта паралельно запитуються відповіді кожного з них.

Для кожної цілі змагаються п'ять алгоритмів класифікації, що відносяться до принципово різних класів моделей машинного навчання. Така різноманітність є навмисною: різні алгоритми мають різні індуктивні упередження і можуть по-різному справлятися зі специфікою окремих цілей – ступенем дисбалансу класів, нелінійністю та взаємодіями між відібраними ознаками.

Логістична регресія є лінійним класифікатором, який моделює ймовірність позитивного класу через логістичну функцію від лінійної комбінації ознак. Параметр збалансованого зважування класів автоматично коригує штрафи за помилки класифікації обернено пропорційно до частоти кожного класу. Попри

концептуальну простоту, логістична регресія демонструє чудову ефективність на структурованих медичних даних, де бінарні предикатори мають прямі та переважно незалежні зв'язки з результатом. Додатковою перевагою є повна інтерпретованість: вагові коефіцієнти безпосередньо відображають внесок кожної ознаки у прогноз.

Випадковий ліс будує ансамбль зі 100 дерев рішень, кожне з яких навчається на випадковій підвибірці даних і розглядає випадкову підмножину ознак при кожному розбитті вузла. Обмеження глибини дерев запобігає перенавчанню на малих класах. Випадковий ліс природно справляється з нелінійними залежностями, взаємодіями між ознаками та шумом у даних.

Градiєнтний бустинг реалізує ансамбль дерев, що будується послідовно: кожне наступне дерево коригує помилки попередніх. Дисбаланс класів компенсується масштабуванням ваги позитивного класу, що обчислюється індивідуально для кожної цілі як відношення кількості негативних прикладів до позитивних.

Багатошаровий перцептрон є нейронною мережею прямого поширення з двома прихованими шарами: 64 та 32 нейрони відповідно. Рання зупинка навчання активована: тренування припиняється автоматично, якщо якість на валідаційній вибірці перестає покращуватися. Нейронна мережа здатна вловлювати складні нелінійні взаємозалежності і є цінним кандидатом для цілей із неочевидними закономірностями.

Метод опорних векторів із радіально-базисним ядром будує оптимальну гіперплощину максимального зазору у перетвореному ознаковому просторі. Застосування гаусового ядра дозволяє моделювати нелінійні межі рішень. Параметр збалансованого зважування класів застосовується аналогічно до логістичної регресії.

3.3.2 Критерій відбору найкращого експерта та порівняння моделей

Після того, як усі п'ять алгоритмів-кандидатів навчені та оцінені на тестових вибірках кожного з 44 цілей, виникає потреба у формалізованому критерії відбору найкращого експерта. Як головну метрику відбору обрано площу під ROC-кривою (ROC-AUC).

Вибір цієї метрики продиктований кількома принциповими міркуваннями. По-перше, ROC-AUC є пороговонезалежною метрикою: вона оцінює здатність моделі ранжувати позитивні приклади вище за негативні незалежно від конкретного порогу класифікації. По-друге, ROC-AUC є стійкою до дисбалансу класів: на відміну від точності (Accuracy), яка на датасеті з 98% негативних прикладів досягає 98% навіть при класифікації всіх записів як негативних, ROC-AUC явно штрафує модель за нездатність виявляти позитивний клас. По-третє, ROC-AUC є порівнянним між цілями різної природи та різного ступеня складності, що дозволяє сформувавши уніфікований рейтинг моделей у рамках єдиного пайплайну.

Поряд із ROC-AUC для кожної моделі обчислюються також точність і F1-міра, що дозволяє отримати повнішу картину якості та виявити патологічні випадки. Ці метрики зберігаються в повному журналі навчання, що містить 220 записів – по 5 моделей для кожного з 44 цілей.

Результати навчання свідчать про надзвичайно високу якість ансамблю моделей-експертів. Серед обраних найкращих моделей середнє значення ROC-AUC становить 0,9991, середня точність – 0,9981, середнє значення F1-міри – 0,9888. Мінімальний ROC-AUC серед усіх 44 найкращих експертів дорівнює 0,9887 (ціль «вправи для великих м'язових груп»), що демонструє стабільно високу якість у всіх задачах без винятку. Для 15 цілей із 44 усі п'ять алгоритмів-кандидатів одночасно досягли ідеального значення $\text{ROC-AUC} = 1,0$ – це насамперед чітко структуровані локальні призначення з очевидними предикторами.

Середні значення метрик усіх п'яти алгоритмів-кандидатів узагальнено в таблиці 3.1.

Таблиця 3.1 – Середні значення метрик за всіма 44 цілями у розрізі алгоритмів

<i>Алгоритм</i>	<i>Точність</i>	<i>ROC-AUC</i>	<i>F1-міра</i>
Логістична регресія	0,9860	0,9951	0,9506
Випадковий ліс	0,9981	0,9990	0,9918
Гرادієнтний бустинг	0,9983	0,9990	0,9919
Нейронна мережа (MLP)	0,9972	0,9988	0,9847
Метод опорних векторів	0,9768	0,9949	0,9456

Аналіз того, який алгоритм став найкращим для кожної цілі, демонструє виражене домінування двох методів. Випадковий ліс здобув перемогу у 21 цілі (47,7%), логістична регресія – у 17 (38,6%), градієнтний бустинг – у 4 (9,1%), нейронна мережа – у 2 (4,5%), метод опорних векторів не здобув жодної перемоги. Таким чином, логістична регресія або випадковий ліс є найкращим вибором для 86,4% цільових показників. Розподіл перемог між алгоритмами ілюструє рисунок 3.10.

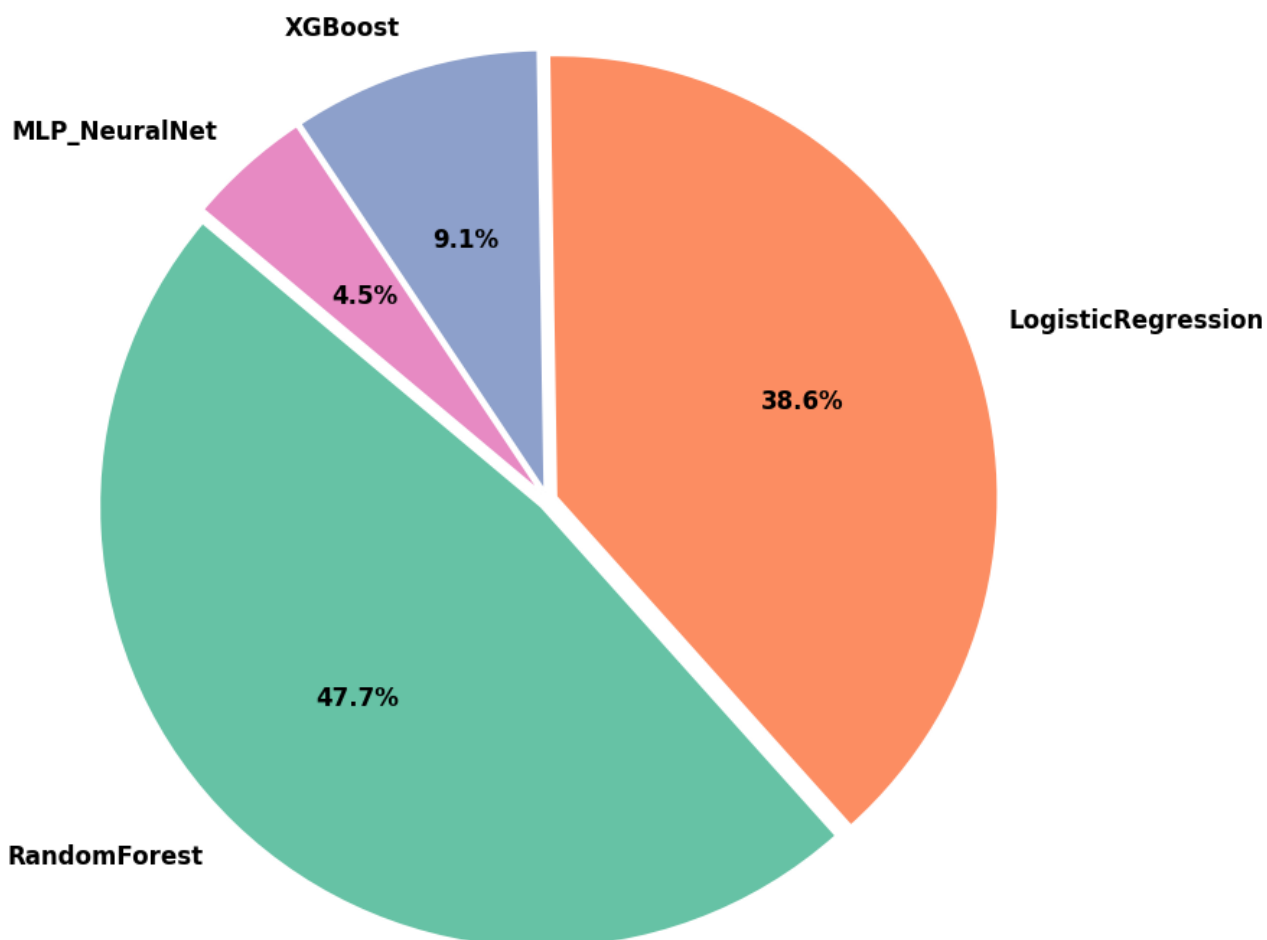
Розподіл обраних алгоритмів для 44 моделей-експертів

Рисунок 3.10 – Розподіл перемог між алгоритмами-кандидатами серед 44 моделей-експертів

Домінування логістичної регресії та випадкового лісу над нейронними мережами і методом опорних векторів є закономірним наслідком природи медичних реабілітаційних даних. Структура предикторів є переважно бінарною та категоріальною: 99 зі 136 вихідних ознак є індикаторами наявності або відсутності конкретних клінічних станів. Зв'язки між такими ознаками та цільовими змінними є здебільшого прямими і логічно інтерпретованими, не потребуючи складних

нелінійних перетворень. Медичні правила призначення реабілітаційних процедур за своєю природою є ієрархічними та кон'юнктивними: «якщо присутній діагноз А і руховий режим В, то призначити процедуру С». Така логіка природно відображається деревами рішень, що складають основу випадкового лісу.

Якість найгірших цілей – «вправи для великих м'язових груп» ($\text{ROC-AUC} = 0,9887, F1 = 0,882$) та «теренкур / дозована ходьба» ($\text{ROC-AUC} = 0,9926, F1 = 0,898$) – є дещо нижчою за середню, що медично пояснюється широким і розмитим клінічним профілем пацієнтів, яким призначаються ці загальнозміцнювальні вправи: вони показані при великій кількості різномірних станів без чіткого і вузького набору предикторів.

3.4 Дворівнева ансамблева архітектура

3.4.1 Побудова матриці мета-ознак

Після того, як усі 44 моделі-експерти навчені та збережені, вони слугують першим рівнем дворівневої ансамблевої архітектури – стекінгу (stacking). Ідея стекінгу полягає в тому, що замість прямого використання прогнозів окремих моделей вводиться додатковий рівень навчання: мета-модель, яка вчиться на виходах моделей першого рівня як на нових ознаках.

Перший рівень архітектури формує матрицю мета-ознак. Для кожного запису з повної бази даних (93 937 пацієнтів) кожен із 44 навчених експертів генерує ймовірність позитивного класу – оцінку того, наскільки ймовірним є призначення конкретної процедури або встановлення певного обмеження для цього пацієнта. Результатом є матриця розміром $93\,937 \times 44$, де кожен стовпець відповідає одній цілі, а кожен рядок – одному пацієнту. Значення в клітині цієї матриці є не бінарним прогнозом, а відкаліброваною ймовірністю від 0 до 1, що несе значно більше інформації і дозволяє мета-моделі врахувати ступінь «впевненості» кожного з 44 експертів у своєму рішенні.

До отриманої матриці ймовірностей додається п'ять контекстних ознак пацієнта – вік, стать, зріст, вага та індекс маси тіла. Ці ознаки відображають базовий демографічний та антропометричний профіль, що є загальним контекстом для прийняття будь-якого реабілітаційного рішення. Включення цих ознак дозволяє мета-моделі враховувати, що набір рекомендацій для молодого пацієнта з нормальним індексом маси тіла систематично відрізняється від рекомендацій для пацієнта похилого віку з ожирінням – навіть при однаковому наборі прогнозів від експертів першого рівня. Фінальна матриця мета-ознак має розмір $93\,937 \times 49$ і є вхідними даними для другого рівня архітектури. Структуру дворівневої системи ілюструє рисунок 3.11.

Структура дворівневої системи стекінгу



Рисунок 3.11 – Структура дворівневої системи стекінгу

3.4.2 Архітектура мета-мозку та оцінювання фінальної ансамблевої моделі

Другий рівень – мета-модель, або «мета-мозок» – навчається на матриці мета-ознак і вирішує задачу синтезу: на основі відкаліброваних сигналів від усіх 44 спеціалізованих експертів та демографічного контексту пацієнта побудувати узгоджений прогноз усіх 44 цільових змінних одночасно.

Як архітектуру мета-мозку обрано багатошаровий перцептрон із двома прихованими шарами – відповідно 128 та 64 нейрони. Функція активації прихованих шарів – ReLU, оптимізатор – адаптивний метод Adam. Матриця мета-ознак розбивається на навчальну та тестову частини у пропорції 80/20.

Вибір нейронної мережі як мета-моделі є методологічно обґрунтованим і суттєво відрізняється від ситуації на першому рівні. На першому рівні моделі-експерти отримують на вхід безпосередньо клінічні показники з переважно прямими медичними зв'язками, де лінійні та деревоподібні моделі природно переважали. На другому рівні вхідними даними є вже не сирі клінічні факти, а 44 відкалібровані ймовірності – складні агреговані сигнали, що вже несуть у собі нелінійно опрацьовану інформацію про пацієнта. Завдання мета-мозку – виявити кореляції та взаємозалежності між цими 44 сигналами. Ці взаємозалежності є нелінійними, тому нейронна мережа з достатньою глибиною є найбільш адекватним інструментом для їхнього вловлювання.

Фінальна дворівнева ансамблева система досягла таких результатів на тестовій вибірці: ROC-AUC (Macro) = 0,9666, F1-Score (Macro) = 0,9603.

Зіставлення метрик мета-мозку із середніми метриками моделей-експертів першого рівня потребує коректної інтерпретації. Середній ROC-AUC найкращих моделей-експертів за індивідуальними задачами становить 0,9991, тоді як мета-мозок демонструє ROC-AUC (Macro) = 0,9666. Ці значення вирішують принципово різні задачі і не є безпосередньо порівнянними: моделі першого рівня оцінювалися кожна на власній тестовій вибірці для однієї бінарної задачі, тоді як мета-мозок вирішує значно складнішу задачу одночасної багатомічкової класифікації за всіма

44 цілями й оцінюється макроусередненим показником, що однаково штрафує за помилки як на частих, так і на найрідкісніших класах.

Значення ROC-AUC (Macro) = 0,9666 у контексті задачі одночасного прогнозування 44 різнорідних клінічних рішень є сильним результатом, що підтверджує здатність системи інтегрувати сигнали від спеціалізованих моделей-експертів у цілісний та клінічно обґрунтований реабілітаційний план. Значення F1-Score (Macro) = 0,9603 свідчить про те, що система зберігає прийнятний баланс між чутливістю та специфічністю навіть для рідкісних цілей.

3.5 Базові моделі для порівняння

Перша базова модель є прямою альтернативою запропонованій архітектурі: замість ансамблю спеціалізованих експертів із динамічним відбором ознак використовується єдина нейронна мережа, яка одночасно вирішує всі 44 задачі класифікації в режимі мультивиходу.

Архітектура мережі складається з трьох прихованих шарів – 256, 128 і 64 нейрони відповідно – із функцією активації ReLU та оптимізатором Adam. Рання зупинка навчання активована для запобігання перенавчанню. Принципова відмінність від запропонованої системи виявляється у двох аспектах. По-перше, модель отримує на вхід повний набір усіх ~138 ознак без будь-якого попереднього відбору. По-друге, усі 44 цільові змінні навчаються спільно в рамках єдиної моделі, тобто градієнти від усіх цілей одночасно впливають на ваги спільних прихованих шарів.

Перша базова модель досягла ROC-AUC (Macro) = 0,9657 та F1-Score (Macro) = 0,8048.

Значення ROC-AUC є досить високим і вказує на те, що нейронна мережа здатна ранжувати приклади за ймовірністю позитивного класу навіть без відбору ознак. Однак значення F1-Score (Macro) = 0,8048 суттєво поступається

запропонованій системі: спільне навчання на всіх ознаках і всіх цілях одночасно призводить до погіршення балансу між чутливістю та специфічністю, особливо для рідкісних цільових класів. Коли градієнти від 44 різнорідних задач конкурують за одні й ті самі ваги мережі, модель вимушена шукати компромісні представлення, неоптимальні для кожної окремої задачі.

Друга базова модель використовує випадковий ліс у режимі мультивиходу: єдиний ансамбль зі 100 дерев рішень із обмеженням глибини до 15 рівнів, навчений одночасно на всіх 44 цільових змінних та повному наборі ознак без попереднього відбору.

Мультивихідний ліс концептуально простіший від нейронної мережі: кожне дерево ансамблю одночасно передбачає всі 44 виходи, а розбиття вузлів оптимізується за агрегованим критерієм якості за всіма цілями. Такий підхід не дозволяє адаптувати набір ознак під кожну окрему цільову змінну і не враховує специфіку індивідуальних задач.

Друга базова модель досягла ROC-AUC (Macro) = 0,9325 та F1-Score (Macro) = 0,5307.

Результати цієї моделі є найнижчими серед усіх порівнюваних систем. Особливо показовим є значення F1-Score (Macro) = 0,5307, що наближається до рівня, типового для випадкового класифікатора на задачах із дисбалансом класів. При одночасному навчанні на 44 різнорідних цілях кожен вузол дерева розбивається за критерієм, усередненим за всіма 44 виходами. Для рідкісних цілей із часткою позитивного класу 0,8–2% такий підхід є особливо деструктивним: їхній внесок у критерій розбиття розчиняється серед більш «гучних» цілей із вищою частотою позитивного класу.

Зведені результати всіх трьох підходів наведено в таблиці 3.2.

Таблиця 3.2 – Порівняння фінальної ансамблевої системи та базових моделей

<i>Система</i>	<i>ROC-AUC (Macro)</i>	<i>F1-Score (Macro)</i>
Базова модель 1 – MLP (мультивихід, без відбору ознак)	0,9657	0,8048
Базова модель 2 – Випадковий ліс (мультивихід, без відбору ознак)	0,9325	0,5307
Фінальна ансамблева система (стекинг)	0,9666	0,9603

Порівняльна діаграма метрик трьох систем наведена на рисунку 3.12.

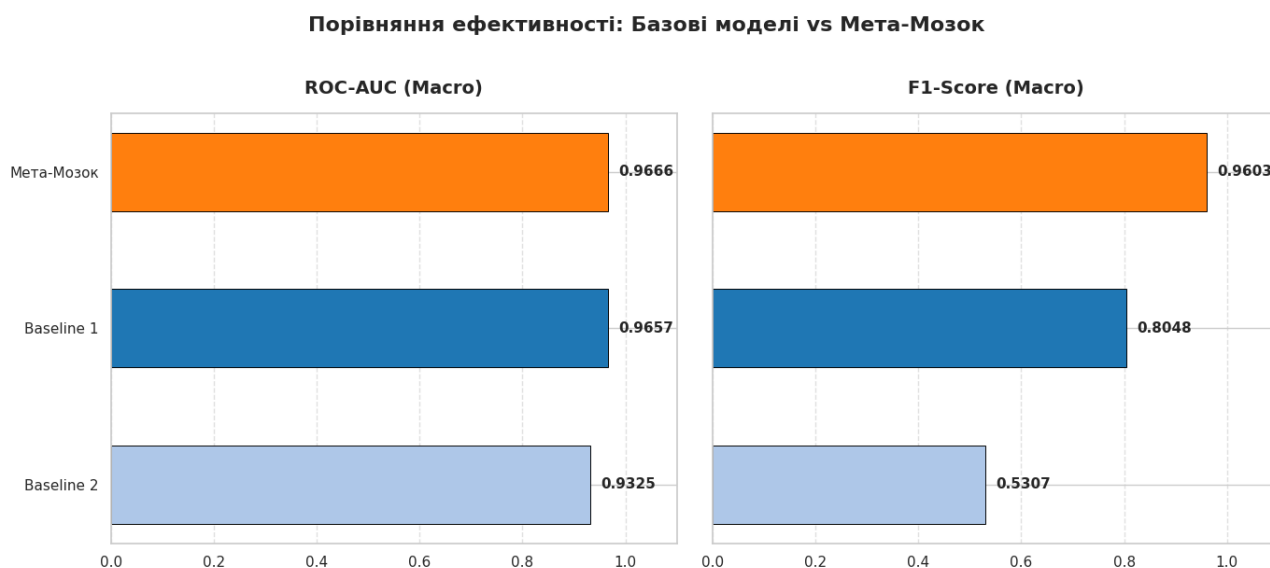


Рисунок 3.12 – Порівняння ефективності фінальної ансамблевої системи та базових моделей

Проаналізувавши результати, робимо висновок, що запропонована дворівнева ансамблева архітектура зі стекингом та динамічним відбором ознак перевершує обидві базові моделі за обома метриками, причому перевага є особливо вираженою за показником F1-Score (Macro).

Порівняно з першою базовою моделлю фінальна система демонструє практично ідентичне значення ROC-AUC (+0,0009) при суттєво вищому F1-Score: 0,9603 проти 0,8048, тобто абсолютний приріст становить +0,1555, або відносне покращення на 19,3%. Це свідчить про те, що обидві системи мають схожу здатність ранжувати приклади, проте ансамблева архітектура значно краще конвертує цю здатність у збалансовані прогнози – суттєво рідше пропускаючи позитивний клас у задачах із дисбалансом.

Порівняно з другою базовою моделлю перевага ансамблю є ще більш вираженою: ROC-AUC вищий на +0,0341, а F1-Score вищий на +0,4296, що у відносному виразі становить покращення на 80,9%.

Ключові структурні рішення, що забезпечують перевагу фінальної системи, можна сформулювати наступним чином. Динамічний відбір ознак усуває шум для кожної цілі індивідуально, покращуючи якість навчання на рідкісних класах. Спеціалізація навчання усуває конкуренцію між градієнтами різних цілей – кожна модель-експерт оптимізується під свою задачу без будь-якого компромісу. Стекінг на мета-рівні дозволяє мета-мозку навчитися узгоджувати взаємозалежні рішення та коригувати систематичні помилки окремих експертів. Нарешті, індивідуальне зважування дисбалансу для кожної цілі на рівні кожного експерта є значно точнішим механізмом, ніж єдина стратегія для всіх 44 задач у мультивихідних моделях.

3.6 Аналіз та інтерпретація результатів

3.6.1 Важливість ознак та клінічна інтерпретація

Одним із ключових критеріїв якості медичної системи підтримки рішень є не лише числові метрики якості прогнозу, але й змістова відповідність виявлених моделлю залежностей клінічній логіці та встановленим протоколам фізичної реабілітації. Система, яка досягає високого ROC-AUC завдяки випадковій

кореляції, не заслуговує на медичну довіру та практичну цінність. Навпаки, система, у якій провідними предикторами для кожної цілі виявляються саме ті клінічні показники, що є патогенетично обґрунтованими для відповідного рішення, є не просто статистично точною, але й клінічно валідною.

Аналіз важливості ознак виконано для репрезентативної вибірки з трьох функціонально різних груп цілей: локально-специфічні вправи, загальнозміцнювальні процедури та системні протипоказання.

Для цілей, пов'язаних із дихальною гімнастикою, найбільш інформативними предикторами виявилися дефіцит ЖЄЛ, наявність хронічних обструктивних захворювань дихальних шляхів та руховий режим «постільний» або «палатний». Це повністю відповідає клінічному протоколу: дихальна гімнастика є першочерговим засобом реабілітації при обмеженій рухомості та зниженні вентиляційної функції легень.

Для вправ для зміцнення м'язового корсету спини найвагомішими виявилися кут Кобба, наявність остеохондрозу поперекового та грудного відділів хребта, спондилолітезу, а також патологічних змін постави – кіфозу та лордозу. Ці показники є безпосередніми клінічними показаннями до стабілізаційних вправ для хребта відповідно до стандартів вертебрології. Включення кута Кобба як одного з провідних предикторів підтверджує, що модель коректно вловлює залежність між ступенем деформації хребта та необхідністю цілеспрямованого зміцнення м'язового корсету.

Провідними предикторами для виключення осьового навантаження на нижні кінцівки стали ступінь зносу хряща, наявність коксартрозу та стан після ендопротезування. Ця залежність є медично бездоганною: при значному зносі суглобового хряща будь-яке осьове навантаження прискорює деструкцію суглоба та посилює больовий синдром, тому його виключення є абсолютною вимогою протоколів реабілітації при дегенеративних захворюваннях суглобів нижніх кінцівок.

Для цілей, що стосуються контролю частоти серцевих скорочень та обмеження навантаження при кардіологічному ризику, провідними предикторами

стали зниження фракції викиду лівого шлуночка, наявність ішемічної хвороби серця, інфаркт міокарда в анамнезі, аритмії та показники артеріального тиску. Відповідність протоколам кардіологічної реабілітації тут є очевидною: кожен із цих показників є самостійним показанням до обмеження інтенсивності фізичного навантаження та обов'язкового кардіомоніторингу під час процедур.

Загальна закономірність, що підтверджує медичну валідність системи: для кожної цілі провідними предикторами виявляються саме ті клінічні показники, які є патогенетично пов'язаними з відповідними рішеннями згідно з чинними реабілітаційними протоколами. Це свідчить про те, що система не просто досягає статистично прийнятних показників, але й вловлює реальні клінічні закономірності.

3.6.2 Аналіз помилок моделі

У задачах реабілітаційного прогнозування два типи помилок класифікації мають принципово різне клінічне значення. Хибнопозитивна помилка – ситуація, коли система рекомендує процедуру або встановлює обмеження, що насправді не є показаним, – є менш небезпечною: зайве призначення дихальних або координаційних вправ мало ймовірно завдасть шкоди пацієнту, хоча й знижує ефективність реабілітаційного плану. Хибнонегативна помилка – ситуація, коли система не виявляє реального показання або протипоказання – є більш небезпечною, особливо для цільових груп обмежень і протипоказань. Пропуск кардіологічного ризику або неврахована необхідність виключення осьового навантаження можуть призвести до погіршення стану пацієнта або ускладнень під час реабілітації. Саме тому вибір ROC-AUC як основної метрики є клінічно обґрунтованим: ця метрика явно штрафує за пропуск позитивного класу і дозволяє за потреби коригувати поріг класифікації в бік підвищення чутливості для критичних цілей безпеки.

Серед 44 цілей найнижчі значення F1-Score зафіксовано для вправ для великих м'язових груп ($F1 = 0,882$) та теренкуру/дозованої ходьби ($F1 = 0,898$). Обидві процедури є загальнозміцнювальними за своєю природою – вони показані при широкому спектрі захворювань без чіткого та вузького набору специфічних предикторів. Клінічне правило для їхнього призначення є розмитим і залежить від складної взаємодії багатьох чинників: загального функціонального стану пацієнта, рухового режиму та сукупності конкретних протипоказань. Така дифузна клінічна логіка є об'єктивно складнішою для формалізованого моделювання порівняно з цілями, де рішення однозначно визначається наявністю конкретного діагнозу.

3.6.3 Обговорення обмежень та напрями розвитку системи

Поточна архітектура системи працює з одноразовим клінічним знімком пацієнта на момент початку або певного етапу реабілітації. Водночас реабілітаційний процес є за своєю природою динамічним: стан пацієнта змінюється від сеансу до сеансу. Перспективним напрямком розвитку є перехід від одноразового прогнозу до моделі динамічного супроводу: відстеження тенденцій ключових клінічних показників у часі та автоматичне коригування реабілітаційних рекомендацій відповідно до зафіксованих змін.

Дослідження не включає валідації на незалежній вибірці із залученням клінічних експертів – фізіотерапевтів та лікарів-реабітологів. Така валідація дозволила б не лише перевірити загальність моделі, але й оцінити медичну обґрунтованість її рекомендацій у конкретних клінічних випадках. Система без зовнішньої клінічної валідації залишається дослідницьким прототипом і не може розглядатися як готовий клінічний інструмент.

Система не забезпечує механізмів онлайн-навчання та адаптації без повного перенавчання. Оновлення клінічних протоколів або поява нових діагностичних

показників вимагають повного перезапуску пайплайну навчання всіх 44 моделей-експертів та мета-мозку.

Найбільш перспективним напрямком розвитку системи є інтеграція мовної моделі як інтерфейсного компонента між лікарем і системою прогнозування. У поточній реалізації система потребує на вході структурованого числового вектора, що вимагає ручного заповнення відповідної форми. На практиці ж лікар-реабілітолог описує стан пацієнта у вільній текстовій формі. Мовна модель, дотренована на медичних текстах, могла б автоматично розпізнавати у такому вільному тексті релевантні клінічні сутності – діагнози, числові показники, хірургічний анамнез, больові локалізації – і формувати на їх основі структурований вхідний вектор для системи прогнозування. Таке поєднання мовного інтерфейсу з дворівневою ансамблевою архітектурою утворило б повноцінну систему підтримки прийняття клінічних рішень у реабілітації.

Іншими перспективними напрямками є: залучення клінічних експертів для оцінки рекомендацій у рамках проспективного пілотного дослідження; застосування методів пояснюваного штучного інтелекту (SHAP-значень) для формування покрокового пояснення кожної рекомендації лікарю; розробка механізмів калібрування ймовірнісних виходів системи; а також розширення системи на суміжні задачі планування інтенсивності навантаження, тривалості курсу та частоти процедур.

Висновки до розділу 3

У третьому розділі здійснено повну програмну реалізацію та комплексне оцінювання дворівневої ансамблевої системи автоматизованого підбору реабілітаційних програм.

Описано структуру та характеристики датасету з 93 937 записів, 136 вхідними предикторами та 44 бінарними цільовими змінними, що охоплюють як

рекомендовані вправи, так і клінічні обмеження і протипоказання. Проаналізовано демографічний і нозологічний профіль когорти, виявлено виражений дисбаланс класів із середньою часткою позитивного класу 20,1%, що визначило вибір ROC-AUC та F1-Score як основних метрик оцінювання.

Розроблено диференційований підхід до передобробки даних: стандартизація неперервних ознак, One-Hot Encoding для рухового режиму та збереження бінарних діагностичних показників без перетворень. Реалізовано алгоритм динамічного відбору ознак, що для кожної з 44 цілей формує індивідуальний набір предикторів (від 5 до 109), кількість яких безпосередньо відображає клінічну складність відповідного рішення.

Навчено ансамбль із 44 моделей-експертів, де для кожної цілі змагалися п'ять алгоритмів-кандидатів. Встановлено, що логістична регресія або випадковий ліс є найкращим вибором для 86,4% цілей, а середній ROC-AUC найкращих моделей-експертів становить 0,9991. Домінування цих алгоритмів отримало клінічне обґрунтування через переважно бінарний та ієрархічно-кон'юнктивний характер реабілітаційних правил.

На основі виходів моделей першого рівня побудовано матрицю мета-ознак розміром $93\ 937 \times 49$ та навчено мета-мозок на базі нейронної мережі з ранньою зупинкою. Фінальна дворівнева система досягла ROC-AUC (Macro) = 0,9666 та F1-Score (Macro) = 0,9603. Порівняння з базовими моделями підтвердило перевагу запропонованої архітектури: покращення F1-Score становить +0,1555 відносно MLP мультивиходу (+19,3%) та +0,4296 відносно мультивихідного випадкового лісу (+80,9%).

Аналіз важливості ознак підтвердив медичну валідність системи: для кожної цілі провідними предикторами виявляються саме ті клінічні показники, що є патогенетично обґрунтованими для відповідного реабілітаційного рішення. Визначено напрями подальшого розвитку системи, зокрема інтеграцію мовної моделі для автоматичного структурування вільнотекстового клінічного опису та перехід до динамічного відстеження змін стану пацієнта в процесі реабілітації.

ВИСНОВКИ

У рамках цієї роботи проведено комплексне дослідження задачі автоматизованого формування індивідуальних програм фізичної реабілітації із застосуванням методів машинного навчання. Здійснено аналіз предметної області, з урахуванням принципів фізичної реабілітації, категорій медичних показників та сучасних підходів до комп'ютеризованого прийняття клінічних рішень. Охарактеризовано медичні дані з гетерогенних відкритих джерел, виявлено їх особливості, зокрема коморбідність, дисбаланс класів та неоднорідність показників, а також потенціал для побудови прогностичних моделей.

На основі порівняльного аналізу алгоритмів машинного навчання (логістична регресія, випадковий ліс, градієнтний бустинг, MLP, SVM) обґрунтовано доцільність ансамблевого підходу. Розроблено дворівневу ансамблеву архітектуру: перший рівень – 44 незалежні моделі-експерти з динамічним відбором ознак для кожної цільової змінної, другий – нейронна мережа-мета-синтезатор, що враховує взаємозв'язки між реабілітаційними рішеннями. Проведено підготовку даних, відбір інформативних ознак, навчання та оцінку моделей за метриками F1-Score та ROC-AUC.

Отримані результати підтверджують високу аналітичну та клінічну цінність розробленої системи, яка досягла F1-Score 0,9603 та ROC-AUC 0,9666, що значно перевищує показники базових підходів. Виявлено, що визначальні ознаки для кожного медичного рішення повністю відповідають чинним клінічним протоколам, що підтверджує медичну обґрунтованість системи. Перспективою подальших досліджень є розширення датасету, врахування динаміки стану пацієнта в часі та інтеграція системи з електронними медичними записами для підвищення точності та практичної застосовності рекомендацій.

ПЕРЕЛІК ПОСИЛАНЬ

1. Rehabilitation // World Health Organization : офіційний вебсайт. URL: <https://www.who.int/health-topics/rehabilitation> (дата звернення: 10.04.2026).
2. World Health Organization. Rehabilitation in Health Systems. Geneva : WHO, 2017. 75 p. URL: <https://www.who.int/publications/i/item/rehabilitation-in-health-systems> (дата звернення: 12.04.2026).
3. Liguori G. et al. ACSM's Guidelines for Exercise Testing and Prescription. 11th ed. Philadelphia : Wolters Kluwer, 2022. 624 p. URL: <https://www.wolterskluwer.com/en/know/acsm/guidelines-for-exercise-testing-and-prescription> (дата звернення: 14.04.2026).
4. Мухін В. М. Фізична реабілітація : підручник. 3-тє вид., переробл. та доповн. Київ : Олімпійська література, 2009. 488 с. URL: <https://sprotyvg7.com.ua/wp-content/uploads/2023/09/Мухін-фізична-реабілітація.pdf> (дата звернення: 14.04.2026).
5. van Buuren S., Groothuis-Oudshoorn K. mice: Multivariate Imputation by Chained Equations in R // Journal of Statistical Software. 2011. Vol. 45, iss. 3. pp. 1–67. DOI: 10.18637/jss.v045.i03. URL: <https://www.jstatsoft.org/v45/i03/> (дата звернення: 15.04.2026).
6. Campagnini S. et al. Machine learning methods for functional recovery prediction and prognosis in post-stroke rehabilitation: a systematic review // Journal of NeuroEngineering and Rehabilitation. 2022. Vol. 19, iss. 1. Article 54. DOI: 10.1186/s12984-022-01032-4. URL: <https://jneuroengrehab.biomedcentral.com/articles/10.1186/s12984-022-01032-4> (дата звернення: 17.04.2026).
7. Ghassemi M. et al. A Review of Challenges and Opportunities in Machine Learning for Health // AMIA Joint Summits on Translational Science proceedings. 2020. Vol. 2020. pp. 191–200. URL: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7233077/> (дата звернення: 17.04.2026).

8. Jung J. M. et al. Center of Pressure- and Machine Learning-based Gait Score and Clinical Risk Factors for Predicting Functional Outcome in Acute Ischemic Stroke // Archives of Physical Medicine and Rehabilitation. 2024. DOI: 10.1016/j.apmr.2024.08.006. URL: <https://pubmed.ncbi.nlm.nih.gov/39187003/> (дата звернення: 17.04.2026).
9. Lundberg S. M., Lee S.-I. A Unified Approach to Interpreting Model Predictions // Advances in Neural Information Processing Systems 30. 2017. pp. 4765–4774. URL: https://proceedings.neurips.cc/paper_files/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf (дата звернення: 18.04.2026).
10. Dreiseitl S., Ohno-Machado L. Logistic regression and artificial neural network classification models: a methodology review // Journal of Biomedical Informatics. 2002. Vol. 35, iss. 5–6. pp. 352–359. DOI: 10.1016/S1532-0464(03)00034-0. URL: <https://www.sciencedirect.com/science/article/pii/S1532046403000340> (дата звернення: 18.04.2026).
11. Breiman L. Random Forests // Machine Learning. 2001. Vol. 45, iss. 1. pp. 5–32. DOI: 10.1023/A:1010933404324. URL: <https://link.springer.com/article/10.1023/A:1010933404324> (дата звернення: 19.04.2026).
12. Chen T., Guestrin C. XGBoost: A Scalable Tree Boosting System // Proceedings of the 22nd ACM SIGKDD. New York : ACM, 2016. pp. 785–794. DOI: 10.1145/2939672.2939785.
13. Cortes C., Vapnik V. Support-vector networks // Machine Learning. 1995. Vol. 20, iss. 3. pp. 273–297. DOI: 10.1007/BF00994018. URL: <https://link.springer.com/article/10.1007/BF00994018> (дата звернення: 19.04.2026).
14. Goodfellow I., Bengio Y., Courville A. Deep Learning. Cambridge : MIT Press, 2016. 800 p. URL: <https://www.deeplearningbook.org> (дата звернення: 20.04.2026).
15. Chawla N. V., Bowyer K. W., Hall L. O., Kegelmeyer W. P. SMOTE: Synthetic Minority Over-sampling Technique // Journal of Artificial Intelligence Research. 2002. Vol. 16. pp. 321–357. DOI: 10.1613/jair.953. URL: <https://www.jair.org/index.php/jair/article/view/10302> (дата звернення: 20.04.2026).

16. Pedregosa F. et al. Scikit-learn: Machine Learning in Python // Journal of Machine Learning Research. 2011. Vol. 12. pp. 2825–2830. URL: <https://jmlr.org/papers/v12/pedregosa11a.html> (дата звернення: 20.04.2026).
17. MIMIC-IV: A freely accessible electronic health record dataset // PhysioNet : вебсайт. URL: <https://physionet.org/content/mimiciv> (дата звернення: 22.04.2026).
18. Lohse K. R., Velez T., Schaefer S. Y., Boyd L. A., Lang C. E., Barnard R. Asking New Questions with Old Data: The Centralized Open-Access Rehabilitation Database for Stroke // Frontiers in Neurology. 2016. Vol. 7. Article 153. DOI: 10.3389/fneur.2016.00153. URL: <https://www.frontiersin.org/articles/10.3389/fneur.2016.00153/full> (дата звернення: 22.04.2026).
19. Seccia R. et al. Data of patients undergoing rehabilitation programs // Data in Brief. 2020. Vol. 30. Article 105419. DOI: 10.1016/j.dib.2020.105419. URL: <https://pmc.ncbi.nlm.nih.gov/articles/PMC7110331/> (дата звернення: 23.04.2026).
20. El Kharoua R. Physical Therapy Exercises Dataset // Kaggle. 2024. URL: <https://www.kaggle.com/datasets/rabieelkharoua/physical-therapy-exercises-dataset> (дата звернення: 23.04.2026).
21. Teboul A. Heart Disease Health Indicators Dataset // Kaggle. 2020. URL: <https://www.kaggle.com/datasets/alexteboul/heart-disease-health-indicators-dataset> (дата звернення: 24.04.2026).

ДОДАТОК А ЛІСТИНГ ПРОГРАМИ

```
# Стандартні бібліотеки Python
```

```
import os
```

```
import json
```

```
import gc
```

```
import time
```

```
import warnings
```

```
# Обробка даних та візуалізація
```

```
import pandas as pd
```

```
import numpy as np
```

```
import matplotlib.pyplot as plt
```

```
import seaborn as sns
```

```
# Збереження моделей та специфічні інструменти Colab
```

```
import joblib
```

```
from google.colab import drive
```

```
from xgboost import XGBClassifier
```

```
# Scikit-learn: Передобробка та валідація
```

```
from sklearn.preprocessing import StandardScaler
```

```
from sklearn.feature_selection import SelectFromModel
```

```
from sklearn.model_selection import train_test_split
```

```
from sklearn.metrics import accuracy_score, roc_auc_score, f1_score,  
classification_report
```

```
# Scikit-learn: Моделі машинного навчання
```

```
from sklearn.ensemble import ExtraTreesClassifier, RandomForestClassifier
```

```

from sklearn.linear_model import LogisticRegression
from sklearn.svm import SVC
from sklearn.neural_network import MLPClassifier

warnings.filterwarnings('ignore')

SEED = 42
os.environ['PYTHONHASHSEED'] = str(SEED)
np.random.seed(SEED)

"""# **Попередній аналіз датасету**"""

# Завантаження даних
df = pd.read_csv('rehab_dataset_master_v2.csv')

# Визначаємо цільові колонки
start_idx = df.columns.get_loc('Вправи_дихальні_динамічні')
target_cols = df.columns[start_idx:]

# --- ПЕРЕВІРКА УНІКАЛЬНИХ ЗНАЧЕНЬ У ЦІЛЬОВИХ ---
print("=== Перевірка унікальних значень у цільових колонках ===")
anomalies_found = False

for col in target_cols:
    unique_vals = df[col].dropna().unique()
    unique_vals.sort()

    # Перевіряємо, чи є щось окрім 0, 1, 0.0, 1.0
    is_strictly_binary = all(val in [0, 1, 0.0, 1.0] for val in unique_vals)

```

```

if not is_strictly_binary:
    print(f'АНОМАЛІЯ в '{col}': {unique_vals}')
    anomalies_found = True

if not anomalies_found:
    print("Усі цільові колонки містять виключно бінарні значення (0 та 1). Аномалій  
не знайдено!\n")

# Приводимо всі таргет-колонки до int
df[target_cols] = df[target_cols].astype(int)

target_start = df.columns.get_loc('Вправи_дихальні_динамічні')

target_cols = df.columns[target_start:]
feature_cols = df.columns[:target_start]

num_targets = len(target_cols)
num_features = len(feature_cols)

# Змінні для підрахунку типів (виключно для ознак)
categorical = 0
boolean = 0
integer = 0
floating = 0

for col in feature_cols:
    dtype = df[col].dtype

    # 1. Перевірка на категоріальні (рядки)

```

```

if dtype == 'object' or str(dtype) == 'category':
    categorical += 1
    continue

# 2. Перевірка на булеві (0 та 1) - навіть якщо вони записані як int або float
unique_vals = df[col].dropna().unique()
is_bool = True
for val in unique_vals:
    if val not in [0, 1, 0.0, 1.0]:
        is_bool = False
        break

if is_bool or str(dtype) == 'bool':
    boolean += 1

# 3. Перевірка на цілочислові
elif pd.api.types.is_integer_dtype(dtype):
    integer += 1

# 4. Перевірка на числа з плаваючою комою
elif pd.api.types.is_float_dtype(dtype):
    floating += 1

print(f"1. Загальна кількість ознак (Features/Predictors): {num_features}")
print(f" - Кількість категоріальних ознак (рядки/текст): {categorical}")
print(f" - Кількість булевих ознак (тільки 0 та 1): {boolean}")
print(f" - Кількість цілочислових ознак (int): {integer}")
print(f" - Кількість ознак із плаваючою комою (float): {floating}")
print(f"\n2. Кількість цільових ознак (Targets): {num_targets}")
print(f"\nЗагальна кількість колонок у датасеті: {len(df.columns)}")

# --- ВІЗУАЛІЗАЦІЯ ДИСБАЛАНСУ (ЗВЕДЕНИЙ ГРАФІК) ---

```

```

# Рахуємо відсоток 1 (оскільки значення 0 і 1, середнє значення * 100 = відсоток
одиниць)
positive_rates = df[target_cols].mean() * 100
positive_rates = positive_rates.sort_values(ascending=False)

plt.figure(figsize=(14, 14))
ax = sns.barplot(x=positive_rates.values, y=positive_rates.index, palette="viridis")

# Додаємо підписи відсотків біля кожного бару
for p in ax.patches:
    width = p.get_width()
    plt.text(width + 0.5,
             p.get_y() + p.get_height() / 2.,
             f'{width:.1f}%',
             ha="left", va="center", fontsize=10)

plt.title('Відсоток позитивного класу (1) для кожної цільової змінної', fontsize=16,
pad=20)
plt.xlabel('Відсоток присутності ознаки (%)', fontsize=12)
plt.ylabel('Цільові змінні', fontsize=12)
plt.xlim(0, 105) # Запас місця для тексту
plt.tight_layout()
plt.show()

"""# **Нормалізація датасету**

```

Оскільки ми вже знаємо структуру датасету, логіка перетворень буде такою:

1. **Цільові ознаки (Таргети):** Залишаємо як є (це наші мітки 0 та 1).
2. **Булеві ознаки (Діагнози, ампутації):** Залишаємо як є (вони вже в масштабі

0-1, масштабування їх тільки зіпсує і зробить неінтерпретованими).

3. ****Категоріальні ознаки ('Motor_Regime'):** Застосовуємо One-Hot Encoding ('pd.get_dummies'), щоб перетворити текст на додаткові булеві колонки з 0 та 1.

4. ****Неперервні числові ознаки (Вік, Тиск, Кут Кобба тощо):** Застосовуємо 'StandardScaler' (стандартизацію), щоб привести їх до середнього = 0 та стандартного відхилення = 1.

"""

1. Завантаження даних

```
print("Завантаження даних...")
```

```
df = pd.read_csv('rehab_dataset_master_v2.csv')
```

2. Розділення на фічі (X) та цільові змінні (y)

```
target_start = df.columns.get_loc('Вправи_дихальні_динамічні')
```

```
X = df.iloc[:, :target_start].copy()
```

```
y = df.iloc[:, target_start:].copy()
```

=====

3. ВИПРАВЛЕННЯ ТИПІВ ДЛЯ ЦІЛЬОВИХ КОЛОНОК

=====

Примусово конвертуємо ВСІ цільові змінні (включно з тими 8 float-колонками) в integer.

Тепер всі 0.0 та 1.0 стануть чистими 0 та 1.

```
y = y.astype(int)
```

```
print(f"Усі {y.shape[1]} цільових колонок успішно приведено до цілочисельного типу (int).")
```

4. Інтелектуальне визначення типів колонок у фічах (X)

```
categorical_cols = X.select_dtypes(include=['object', 'category']).columns.tolist()
```

```

boolean_cols = []
continuous_cols = []

for col in X.columns:
    if col in categorical_cols:
        continue

    # Перевірка на булеві значення серед фіч
    unique_vals = X[col].dropna().unique()
    if set(unique_vals).issubset({0, 1, 0.0, 1.0}):
        boolean_cols.append(col)
    else:
        continuous_cols.append(col)

print(f"\nЗнайдено ознак (фіч):")
print(f" - Категоріальних: {len(categorical_cols)} {categorical_cols}")
print(f" - Булевих: {len(boolean_cols)}")
print(f" - Неперервних (для масштабування): {len(continuous_cols)}")

# 5. Обробка категоріальної ознаки (One-Hot Encoding)
if categorical_cols:
    X = pd.get_dummies(X, columns=categorical_cols, drop_first=True, dtype=int)

# 6. Масштабування неперервних числових ознак
scaler = StandardScaler()
X[continuous_cols] = scaler.fit_transform(X[continuous_cols])

# 7. З'єднання фіч та таргетів назад в один очищений датасет
df_prepared = pd.concat([X, y], axis=1)

```

```

print(f"\nСтарий розмір датасету: {df.shape}")
print(f"Новий розмір датасету: {df_prepared.shape}")

# 8. Збереження готового до ML датасету
df_prepared.to_csv('rehab_dataset_prepared.csv', index=False)
print("\nГотовий, повністю стандартизований та типізований датасет збережено як
'rehab_dataset_prepared.csv")

"""# **Розбиття датасету на підзадачі**"""

df = pd.read_csv('rehab_dataset_prepared.csv')

# === РОЗБИТТЯ НА X ТА Y (БЕЗ ДУБЛЮВАННЯ ДАНИХ) ===
target_start = df.columns.get_loc('Вправи_дихальні_динамічні')
X = df.iloc[:, :target_start]
Y = df.iloc[:, target_start:]

print(f"Розмір матриці ознак X: {X.shape}")
print(f"Кількість цільових змінних Y: {Y.shape[1]}")

# Оптимізація типів даних для X (зниження використання пам'яті на ~50%)
for col in X.columns:
    if X[col].dtype == 'float64':
        X[col] = X[col].astype('float32')
    elif X[col].dtype == 'int64':
        X[col] = X[col].astype('int32')

print("Оптимізація пам'яті завершена.")

target_features_map = {}

```

```

# Залишаємо дерева, бо вони чудово ловлять нелінійні зв'язки в медичних даних
base_estimator = ExtraTreesClassifier(
    n_estimators=100,      # Збільшили до 100 для більш точної оцінки важливості
    max_depth=7,
    class_weight='balanced', # Все ще враховуємо дисбаланс класів (2-4%)
    random_state=SEED,
    n_jobs=-1
)

for idx, target_col in enumerate(Y.columns, 1):
    y_target = Y[target_col]

    # 1. Навчаємо модель на одному таргеті
    base_estimator.fit(X, y_target)

    # 2. Отримуємо "вагу" кожної ознаки
    importances = base_estimator.feature_importances_

    # 3. Сортуємо індекси фіч від найважливішої до найменш важливої
    indices = np.argsort(importances)[::-1]

    # 4. Рахуємо кумулятивну (накопичувальну) суму важливостей
    cum_importances = np.cumsum(importances[indices])

    # 5. Знаходимо, скільки фіч потрібно, щоб зібрати 95% всієї важливості
    # (Можна змінити на 0.90, якщо хочеш ще менше фіч залишати)
    num_features = np.argmax(cum_importances >= 0.95) + 1

    # 6. Встановлюємо мінімальний поріг (щоб не було таргета з 1-2 фічами, які

```

можуть перенавчитись)

```
if num_features < 5:
    num_features = 5
```

7. Відбираємо імена колонок

```
selected_indices = indices[:num_features]
selected_features = X.columns[selected_indices].tolist()
```

```
target_features_map[target_col] = selected_features
```

```
print(f"[{idx:02d}/{Y.shape[1]}] {target_col[:40]:<42} -> Відібрано ознак:
{len(selected_features)}")
```

```
gc.collect()
```

Збереження результатів

```
with open('target_features_map_dynamic.json', 'w', encoding='utf-8') as f:
    json.dump(target_features_map, f, ensure_ascii=False, indent=4)
```

"""Динамічний відбір ознак спрацював ідеально, адаптувавши розмірність даних (від 5 до 109 ознак) під індивідуальну складність кожної цільової змінної. Це логічно підтверджує медичну специфіку датасету: прості локальні вправи залежать лише від кількох ключових факторів, тоді як комплексні системні обмеження вимагають аналізу десятків показників пацієнта.

Експеримент із прибиранням дублікатів

```
"""
```

```
df = pd.read_csv('rehab_dataset_prepared.csv')
```

```
with open('target_features_map_dynamic.json', 'r', encoding='utf-8') as f:
    features_map = json.load(f)
```

```
results = []
```

```
print("Аналіз дедуплікації по 44 піддатасетах...\n")
```

```
for target, features in features_map.items():
```

```
    # Формуємо піддатасет
```

```
    cols = features + [target]
```

```
    sub_df = df[cols]
```

```
    # СТАТИСТИКА ДО
```

```
    n_before = len(sub_df)
```

```
    pos_percent_before = sub_df[target].mean() * 100
```

```
    # ДЕДУПЛІКАЦІЯ
```

```
    # Ми видаляємо повні дублікати (ідентичні і фічі, і таргет)
```

```
    sub_df_dedup = sub_df.drop_duplicates()
```

```
    # СТАТИСТИКА ПІСЛЯ
```

```
    n_after = len(sub_df_dedup)
```

```
    pos_percent_after = sub_df_dedup[target].mean() * 100
```

```
    # РАХУЄМО КОНФЛІКТИ (однакові фічі, але різні таргети 0 та 1)
```

```
    # Згрупуємо по фічах і подивимось, скільки унікальних таргетів вони мають
```

```
    conflicts = sub_df_dedup.groupby(features)[target].nunique()
```

```
    conflict_rows = (conflicts > 1).sum()
```

```
    results.append({
```

```

'Target': target,
'N_Features': len(features),
'Rows_Before': n_before,
'Rows_After': n_after,
'Removed_%': (n_before - n_after) / n_before * 100,
'Pos_Class_Before_%': pos_percent_before,
'Pos_Class_After_%': pos_percent_after,
'Conflicting_Profiles': conflict_rows
}))

# Перетворюємо в DataFrame для зручності
results_df = pd.DataFrame(results)

# Сортуюмо за найбільшим відсотком видалених рядків
results_df = results_df.sort_values('Removed_%', ascending=False)

# Виводимо топ-5 найбільш "схлопнутих" піддасетів
print("=== ТОП-5 ПІДДАТАСЕТІВ З НАЙБІЛЬШОЮ КІЛЬКІСТЮ ДУБЛІКАТІВ ===")

display_cols = ['Target', 'N_Features', 'Rows_After', 'Removed_%',
'Pos_Class_Before_%', 'Pos_Class_After_%', 'Conflicting_Profiles']
print(results_df[display_cols].to_string(index=False))

# === ВІЗУАЛІЗАЦІЯ (Dumbbell Plot / Гантельковий графік) ===
# Візуалізуємо зміну дисбалансу класів
plot_data = results_df.head(44).copy()

plt.figure(figsize=(12, 10))
# Малюємо лінії між "До" і "Після"
plt.hlines(y=range(len(plot_data)), xmin=plot_data['Pos_Class_Before_%'],

```

```

xmax=plot_data['Pos_Class_After_%'], color='grey', alpha=0.4)

# Малюємо точки
plt.scatter(plot_data['Pos_Class_Before_%'], range(len(plot_data)), color='skyblue',
label='До дедуплікації', s=100)
plt.scatter(plot_data['Pos_Class_After_%'], range(len(plot_data)), color='red',
label='Після дедуплікації', s=100)

plt.yticks(range(len(plot_data)), plot_data['Target'].str[:45] + '...')
plt.xlabel('Відсоток позитивного класу (Target = 1), %')
plt.title('Зміна дисбалансу класів після видалення дублікатів')
plt.legend()
plt.grid(axis='x', linestyle='--', alpha=0.5)
plt.tight_layout()
plt.show()

"""# **Навчання моделей-експертів**"""

# === 1. ПІДКЛЮЧЕННЯ GOOGLE DRIVE ТА НАЛАШТУВАННЯ ПАПОК ===
drive.mount('/content/drive')

BASE_DIR = '/content/drive/MyDrive/Rehab_Models_Ensemble'
MODELS_DIR = os.path.join(BASE_DIR, 'saved_experts')
os.makedirs(MODELS_DIR, exist_ok=True)

LOG_FILE = os.path.join(BASE_DIR, 'training_log.csv')
MAPPING_FILE = os.path.join(BASE_DIR, 'target_mapping.json')

# === 2. ЗАВАНТАЖЕННЯ ДАНИХ ===
print("Завантаження даних...")

```

```

df = pd.read_csv('rehab_dataset_prepared.csv')
with open('target_features_map_dynamic.json', 'r', encoding='utf-8') as f:
    features_map = json.load(f)

# Знаходимо всі цільові змінні
target_start = df.columns.get_loc('Вправи_дихальні_динамічні')
targets = df.columns[target_start:].tolist()
X_full = df.iloc[:, :target_start]
Y_full = df.iloc[:, target_start:]

target_mapping = {f'y_{i}": target for i, target in enumerate(targets)}
reverse_mapping = {target: f'y_{i}" for i, target in enumerate(targets)}

with open(MAPPING_FILE, 'w', encoding='utf-8') as f:
    json.dump(target_mapping, f, ensure_ascii=False, indent=4)
print(f"Створено мапінг для {len(target_mapping)} цільових змінних. Збережено в {MAPPING_FILE}")

if not os.path.exists(LOG_FILE):
    pd.DataFrame(columns=[
        'Alias', 'Target_Name', 'Model_Name', 'Accuracy', 'ROC_AUC', 'F1_Score',
        'Is_Best'
    ]).to_csv(LOG_FILE, index=False)
    processed_targets = set()
else:
    # Якщо лог існує, читаємо, які моделі вже повністю оброблені
    log_df = pd.read_csv(LOG_FILE)
    processed_targets = set(log_df[log_df['Is_Best'] == True]['Alias'].unique())
    print(f"Знайдено попередній лог. Вже оброблено експертів: {len(processed_targets)}/{len(targets)}")

```

```

# === ГОЛОВНИЙ ЦИКЛ НАВЧАННЯ ===
for alias, target_name in target_mapping.items():
    if alias in processed_targets:
        continue # Пропускаємо вже навчені

    print(f'\n[ {alias} ] Починаємо навчання експерта для: {target_name[:40]}...')

    # 1. Формуємо піддатасет
    selected_features = features_map[target_name]
    X = X_full[selected_features].astype('float32') # Оптимізація пам'яті
    y = Y_full[target_name].astype('int8')

    # 2. Розбиття на train/test (80/20) із збереженням пропорції класів (stratify)
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42, stratify=y)

    # Розрахунок ваги для XGBoost (кількість нулів / кількість одиниць)
    pos_count = sum(y_train == 1)
    neg_count = sum(y_train == 0)
    xgb_scale = neg_count / pos_count if pos_count > 0 else 1.0

    # 3. Ініціалізація зоопарку моделей
    models = {
        'LogisticRegression': LogisticRegression(class_weight='balanced', max_iter=500,
random_state=42),
        'RandomForest': RandomForestClassifier(n_estimators=100,
class_weight='balanced', max_depth=10, random_state=42, n_jobs=-1),
        'XGBoost': XGBClassifier(scale_pos_weight=xgb_scale, eval_metric='logloss',
random_state=42, n_jobs=-1),

```

```

'MLP_NeuralNet': MLPClassifier(hidden_layer_sizes=(64, 32), max_iter=200,
early_stopping=True, random_state=42),
'SVM_RBF': SVC(kernel='rbf', class_weight='balanced', probability=True,
max_iter=1000, random_state=42)
}

```

```
best_model_name = ""
```

```
best_score = -1
```

```
best_model = None
```

```
target_results = []
```

```
# 4. Навчання та оцінка кожної моделі
```

```
for model_name, model in models.items():
```

```
    print(f" -> Навчання {model_name}...")
```

```
    try:
```

```
        model.fit(X_train, y_train)
```

```
    # Прогнози
```

```
    y_pred = model.predict(X_test)
```

```
    if hasattr(model, "predict_proba"):
```

```
        y_prob = model.predict_proba(X_test)[:, 1]
```

```
    else:
```

```
        y_prob = model.decision_function(X_test) # Для SVM без probability
```

```
    # Метрики
```

```
    acc = accuracy_score(y_test, y_pred)
```

```
    roc = roc_auc_score(y_test, y_prob)
```

```
    f1 = f1_score(y_test, y_pred)
```

```
    target_results.append({
```

```

        'Alias': alias, 'Target_Name': target_name, 'Model_Name': model_name,
        'Accuracy': acc, 'ROC_AUC': roc, 'F1_Score': f1, 'Is_Best': False
    })

    # Обираємо найкращу за ROC_AUC (можна змінити на f1, якщо
критичніше точність позитивного класу)
    if roc > best_score:
        best_score = roc
        best_model_name = model_name
        best_model = model

except Exception as e:
    print(f" Помилка при навчанні {model_name}: {e}")

# 5.5. Відмічаємо найкращу модель і зберігаємо логи
print(f" Найкраща модель для {alias}: {best_model_name} (ROC-AUC =
{best_score:.4f})")

for res in target_results:
    if res['Model_Name'] == best_model_name:
        res['Is_Best'] = True

# Допишуємо результати в CSV
res_df = pd.DataFrame(target_results)
res_df.to_csv(LOG_FILE, mode='a', header=False, index=False)

# 5.6. Зберігаємо ваги (файл) найкращої моделі на Google Drive
model_filename = os.path.join(MODELS_DIR,
f"{alias}_best_{best_model_name}.pkl")
joblib.dump(best_model, model_filename)

```

```

# Примусове очищення пам'яті
del X, y, X_train, X_test, y_train, y_test, models, best_model, target_results
gc.collect()

print("\nУСІ 44 ЕКСПЕРТИ УСПІШНО НАВЧЕНІ ТА ЗБЕРЕЖЕНІ!")

# Шлях до лог-файлу на Google Drive
LOG_FILE = '/content/drive/MyDrive/Rehab_Models_Ensemble/training_log.csv'

if os.path.exists(LOG_FILE):
    # 1. Завантажуємо логи
    log_df = pd.read_csv(LOG_FILE)

    # 2. Фільтруємо лише ті рядки, де модель була обрана як найкраща
    # На всяк випадок переводимо в рядок, щоб уникнути помилок між True та
    "True"
    best_models_df = log_df[log_df['Is_Best'].astype(str).str.lower() == 'true']

    # 3. Підраховуємо, скільки разів кожна модель стала найкращою
    model_counts = best_models_df['Model_Name'].value_counts()

    print("=== Статистика вибору моделей ===")
    print(model_counts)
    print(f"\nЗагальна кількість обраних експертів: {model_counts.sum()} / 44")

    # 4. Побудова кругової діаграми
    plt.figure(figsize=(9, 9))

    # Беремо красиву палітру від Seaborn

```

```

colors = sns.color_palette("Set2")[0:len(model_counts)]

# Малюємо діаграму (autopct виводить відсотки, explode робить невеликий
відступ)
wedges, texts, autotexts = plt.pie(
    model_counts.values,
    labels=model_counts.index,
    colors=colors,
    autopct='%1.1f%%',
    startangle=140,
    explode=[0.02] * len(model_counts), # розділяємо шматочки для краси
    wedgeprops={'edgecolor': 'white', 'linewidth': 1.5},
    textprops={'fontsize': 12, 'fontweight': 'bold'}
)

plt.title('Розподіл обраних алгоритмів для 44 моделей-експертів', fontsize=16,
pad=20, fontweight='bold')
plt.axis('equal') # Щоб коло не сплюснулося
plt.tight_layout()
plt.show()

else:
    print(f"Файл {LOG_FILE} не знайдено.")

BASE_DIR = '/content/drive/MyDrive/Rehab_Models_Ensemble'
MODELS_DIR = os.path.join(BASE_DIR, 'saved_experts')
MAPPING_FILE = os.path.join(BASE_DIR, 'target_mapping.json')
LOG_FILE = os.path.join(BASE_DIR, 'training_log.csv')

print("Завантаження базового датасету...")

```

```

df = pd.read_csv('rehab_dataset_prepared.csv')

with open('target_features_map_dynamic.json', 'r', encoding='utf-8') as f:
    features_map = json.load(f)

with open(MAPPING_FILE, 'r', encoding='utf-8') as f:
    target_mapping = json.load(f)

target_start = df.columns.get_loc('Вправи_дихальні_динамічні')
X_full = df.iloc[:, :target_start]
Y_full = df.iloc[:, target_start:]

meta_features = pd.DataFrame(index=df.index)

# Читаємо лог, щоб дізнатися, яка модель для якого таргета була найкращою
log_df = pd.read_csv(LOG_FILE)
best_models_info = log_df[log_df['Is_Best'].astype(str).str.lower() ==
'true'].set_index('Alias')

for alias, target_name in target_mapping.items():
    # Знаходимо ім'я найкращої моделі для цього таргета
    if alias not in best_models_info.index:
        print(f'Пропущено {alias} (не знайдено в логах як Is_Best)')
        continue

    best_model_name = best_models_info.loc[alias, 'Model_Name']
    model_filename = os.path.join(MODELS_DIR,
f'{alias}_best_{best_model_name}.pkl')

    # Завантажуємо модель-експерта

```

```

expert_model = joblib.load(model_filename)

# Витягуємо необхідні фічі для цього експерта
selected_features = features_map[target_name]
X_expert = X_full[selected_features].astype('float32')

# Просимо експерта дати ймовірність (probability) позитивного класу
if hasattr(expert_model, "predict_proba"):
    prob = expert_model.predict_proba(X_expert)[:, 1]
else:
    # Fallback для моделей, які не мають predict_proba (напр. деякі SVM)
    dec_func = expert_model.decision_function(X_expert)
    prob = 1 / (1 + np.exp(-dec_func)) # Сигмоїда для переведення в 0..1

# Зберігаємо ймовірність у новий датасет як "prob_y_0", "prob_y_1" тощо
meta_features[f"prob_{alias}"] = prob

if int(alias.split('_')[1]) % 10 == 0:
    print(f" Опитано експертів: {alias}...")

print(f"Успішно зібрано матрицю ймовірностей. Розмір: {meta_features.shape}")

# Додаємо кілька базових фіч, щоб мета-мозок бачив загальну картину (наприклад
вік, зріст, вага)
# Оскільки ми вже зробили StandardScaler раніше, вони готові до використання
context_columns = ['Age', 'Sex', 'Height_cm', 'Weight_kg', 'BMI']
X_meta = pd.concat([meta_features, X_full[context_columns]], axis=1)

print(f"Додано контекстні фічі. Фінальний розмір X_meta: {X_meta.shape}")

```

```

# Розбиваємо дані. Y_full - це матриця з 44 колонок!
X_meta_train, X_meta_test, Y_train, Y_test = train_test_split(
    X_meta, Y_full, test_size=0.2, random_state=42
)

# Створюємо Багатошаровий Перцептрон (DNN)
# Архітектура: Вхід(49) -> Прихований(128) -> Прихований(64) -> Вихід(44)
meta_brain = MLPClassifier(
    hidden_layer_sizes=(128, 64),
    activation='relu',      # Відкидає від'ємні значення
    solver='adam',         # Сучасний оптимізатор
    max_iter=300,
    early_stopping=True,    # Зупиниться, якщо перестане покращуватись
    random_state=42,
    verbose=True           # Будемо бачити процес навчання
)

# Навчаємо мета-мозок
print("Починаємо епохи навчання (може зайняти кілька хвилин)...")
meta_brain.fit(X_meta_train, Y_train)

Y_pred = meta_brain.predict(X_meta_test)
Y_prob = meta_brain.predict_proba(X_meta_test)

roc_macro = roc_auc_score(Y_test, Y_prob, average='macro')
f1_macro = f1_score(Y_test, Y_pred, average='macro')

print(f"Мета-Мозок ROC-AUC (Macro): {roc_macro}")
print(f"Мета-Мозок F1-Score (Macro): {f1_macro}")

```

```

meta_model_path = os.path.join(BASE_DIR, 'META_BRAIN_MLP.pkl')
joblib.dump(meta_brain, meta_model_path)
print(f"Фінальний Мета-Мозок збережено у: {meta_model_path}")

"""# **Baseline**"""

# === 1. ЗАВАНТАЖЕННЯ ДАНИХ (БЕЗ ВІДБОРУ ОЗНАК) ===
print("Завантаження даних...")
df = pd.read_csv('rehab_dataset_prepared.csv')

# Ділимо на X (всі ознаки) та Y (всі 44 цілі)
target_start = df.columns.get_loc('Вправи_дихальні_динамічні')
X = df.iloc[:, :target_start].astype('float32') # Усі ~138 ознак "в лоб"
Y = df.iloc[:, target_start:].astype('int8')    # 44 цільові змінні

# Розбиття на train/test (Без складної стратифікації, бо для 44 міток одночасно це
зробити важко)
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.2,
random_state=42)

print(f"Розмір X_train: {X_train.shape}, Y_train: {Y_train.shape}")
print(f"Розмір X_test: {X_test.shape}, Y_test: {Y_test.shape}\n")

"""## **Baseline 1: Багатошаровий перцептрон**"""

mlp_baseline = MLPClassifier(
    hidden_layer_sizes=(256, 128, 64),
    activation='relu',
    solver='adam',
    early_stopping=True,

```

```

    random_state=42
)

start_time = time.time()
mlp_baseline.fit(X_train, Y_train)
mlp_time = time.time() - start_time

Y_pred_mlp = mlp_baseline.predict(X_test)
Y_prob_mlp = mlp_baseline.predict_proba(X_test)

print(f"Час навчання: {mlp_time:.2f} сек")
print(f"ROC-AUC (Macro): {roc_auc_score(Y_test, Y_prob_mlp,
average='macro'):.4f}")
print(f"F1-Score (Macro): {f1_score(Y_test, Y_pred_mlp, average='macro'):.4f}\n")

"""## **Baseline 2: Random Forest Classifier**"""

rf_baseline = RandomForestClassifier(
    n_estimators=100,
    max_depth=15,
    random_state=42,
    n_jobs=-1
)

start_time = time.time()
rf_baseline.fit(X_train, Y_train)
rf_time = time.time() - start_time

Y_pred_rf = rf_baseline.predict(X_test)
rf_probs = rf_baseline.predict_proba(X_test)

```

```
Y_prob_rf = np.array([cls_probs[:, 1] for cls_probs in rf_probs]).T
```

```
print(f"Час навчання: {rf_time:.2f} сек")
```

```
print(f"ROC-AUC (Macro): {roc_auc_score(Y_test, Y_prob_rf, average='macro'):.4f}")
```

```
print(f"F1-Score (Macro): {f1_score(Y_test, Y_pred_rf, average='macro'):.4f}\n")
```

ДОДАТОК Б ЕЛЕКТРОННА ПРЕЗЕНТАЦІЯ



КАФЕДРА
ШТУЧНОГО
ІНТЕЛЕКТУ

КПІ ІМ. ІГОРЯ СІКОРСЬКОГО
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ПРИКЛАДНОГО СИСТЕМНОГО АНАЛІЗУ
КАФЕДРА ШТУЧНОГО ІНТЕЛЕКТУ

Нейромережеві моделі для формування адаптивних комплексів фізичної реабілітації за медичними показниками

Кваліфікаційна робота на здобуття ступеня бакалавра за освітньо-професійною програмою «Системи і методи штучного інтелекту» спеціальності 122 «Комп'ютерні науки»

КІ-22

Мамонов Іван Ігорович

Керівник: доцент Пишнограєв І. О.

Рецензент: доцент, Лазаренко І.С.

2026

Рисунок Б.1 – Титульний слайд



КАФЕДРА
ШТУЧНОГО
ІНТЕЛЕКТУ

Актуальність роботи



Медико-соціальна потреба.
Існує суттєвий розрив між потребою в індивідуалізованих реабілітаційних програмах та обмеженими можливостями їхнього формування через дефіцит часу кваліфікованих кадрів.



Клінічна складність.
Ручне складання безпечного плану значно ускладнюється коморбідним профілем пацієнтів, що вимагає одночасного врахування десятків різномірних показників.



Практична значущість.
Автоматизація процесу дозволить оптимізувати прийняття клінічних рішень та знизити ймовірність пропуску критично важливих обмежень чи протипоказань.

03

Рисунок Б.2 –Слайд 2



КАФЕДРА
ШТУЧНОГО
ІНТЕЛЕКТУ



Нейромережеві моделі для формування адаптивних комплексів фізичної реабілітації за медичними показниками

Об'єкт, предмет, мета, проблема, завдання

Об'єкт: Процес автоматизованого формування індивідуальних програм фізичної реабілітації на основі комплексних медичних показників пацієнтів.

Предмет: Нейромережеві моделі та алгоритми машинного навчання для прогнозування оптимальних клінічних рішень.

Мета: Розробка та експериментальна верифікація інтелектуальної системи підтримки прийняття рішень для комплексного підбору реабілітаційних рекомендацій та обмежень.

Проблема: Складність врахування коморбідних станів та дисбалансу медичних даних при складанні адаптивних програм в умовах дефіциту часу лікарів-реабілітологів.

Завдання: Інтеграція клінічних даних, розробка механізмів відбору інформативних ознак, побудова прогностичної системи та оцінка її ефективності і медичної валідності.

02

Рисунок Б.3 –Слайд 3



КАФЕДРА
ШТУЧНОГО
ІНТЕЛЕКТУ



Особенности интегрированного датасета

Для датасета было интегрировано гетерогенные открытые источники (MIMIC-IV, CARDS, Tack et al та інші).

ЗАГАЛЬНА КІЛЬКІСТЬ ЗАПИСІВ

93 937

ЗАГАЛЬНА КІЛЬКІСТЬ КОЛОНОК

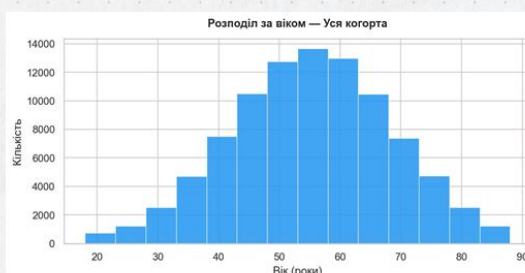
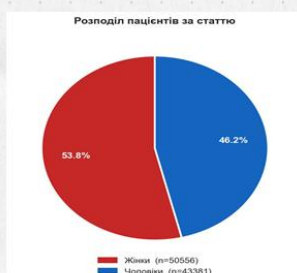
180

ВХІДНІ ОЗНАКИ

136

ЦІЛЬОВИХ ОЗНАК

44



04

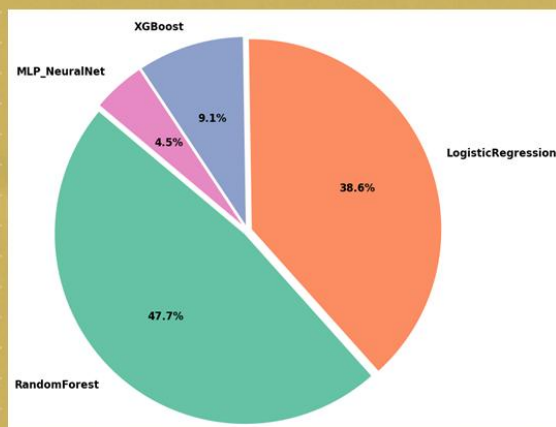
Рисунок Б.4 –Слайд 4



КАФЕДРА
ШТУЧНОГО
ІНТЕЛЕКТУ



Порівняння моделей експертів



Алгоритм	Точність	ROC-AUC	F1-міра
Логістична регресія	0,9860	0,9951	0,9506
Випадковий ліс	0,9981	0,9990	0,9918
Гرادієнтний бустинг	0,9983	0,9990	0,9919
Нейронна мережа (MLP)	0,9972	0,9988	0,9847
Метод опорних векторів	0,9768	0,9949	0,9456

07

Рисунок Б.7 –Слайд 7



КАФЕДРА
ШТУЧНОГО
ІНТЕЛЕКТУ



Результати фінальної системи

АРХІТЕКТУРА МОДЕЛІ	ROC-AUC (MACRO)	F1-SCORE (MACRO)
Ансамблева система	0.9666	0.9603
Базова модель 1 (MLP)	0.9657	0.8048
Базова модель 2 (Random Forest)	0.9325	0.5307

Приріст відносно Baseline 1

Δ F1-SCORE **0.1555**
 Δ ROC-AUC **0.0009**

Приріст відносно Baseline 2

Δ F1-SCORE **0.4296**
 Δ ROC-AUC **0.0341**

08

Рисунок Б.8 –Слайд 8



Рисунок Б.9 –Слайд 9

КАФЕДРА ШТУЧНОГО ІНТЕЛЕКТУ

Висновки

1. У результаті роботи створено дворівневу ансамблевую систему з високою точністю прогнозування реабілітаційних програм, що досягла показника F1-Score 0,9603.
2. Виявлено, що для кожного медичного рішення найважливішими є специфічні патогенетичні показники, які повністю відповідають чинним клінічним протоколам.
3. Дворівневий підхід із динамічним відбором ознак виявився значно ефективнішим за класичні мультивихідні моделі.
4. Розроблена система має доведену клінічну валідність та високу практичну цінність для підтримки прийняття рішень лікарями-реабілітологами.

10

Рисунок Б.10 –Слайд 10



Дякую за увагу!

Тема: «Нейромережеві моделі для формування адаптивних комплексів фізичної реабілітації за медичними показниками»

Виконав: Мамонов Іван Ігорович

Керівник: Пишнограєв Іван Олександрович

Рисунок Б.11 –Слайд 11