

БАКАЛАВРСЬКА РОБОТА

КРБ.СІ-21.00.00.000 ПЗ

Група СІ-22-1

Сергій КОБЛЬОВСЬКИЙ

2026

Міністерство освіти і науки України
Івано-Франківський національний технічний університет нафти і газу
Факультет інформаційних технологій
Кафедра інформаційно-телекомунікаційних технологій і систем

Кобльовський Сергій Юрійович

(прізвище, ім'я, по-батькові)

УДК 681.5:004.85:622.276
(індекс)

БАКАЛАВРСЬКА РОБОТА

Розроблення програмного модуля автоматизованої класифікації стану
штангових глибинно-насосних установок на основі інтелектуального аналізу
динамограм

(назва роботи)

Системна інженерія - Інтернет речей

(назва освітньої програми)

151 – Автоматизація та комп'ютерно-інтегровані технології

(шифр і назва спеціальності)

**Робота містить результати власних досліджень, використання ідей,
результатів і текстів інших авторів мають посилання на відповідне
джерело:**

Здобувач освітнього рівня _____ С. Ю. Кобльовський _____

(підпис, ініціали та прізвище здобувача)

Науковий керівник _____ к.т.н., доц. Л.О. Штаєр _____

(підпис, прізвище, ім'я, по батькові, науковий ступінь, вчене звання керівника)

Допущено до захисту

В. о. завідувача кафедри _____ к.т.н., доц. О.Л. Заміховська _____

(посада)

(підпис)

(дата)

(ініціали та прізвище)

Івано-Франківськ

2026

Івано-Франківський національний технічний університет нафти і газу

(повне найменування закладу вищої освіти)

Факультет Інформаційних технологій

Кафедра Інформаційно-телекомунікаційних технологій та систем

Освітній рівень Бакалавр

Спеціальність 151 – Автоматизація та комп'ютерно-інтегровані технології

(шифр і назва)

ЗАТВЕРДЖУЮ

В.о. завідувача кафедри ІТТС, к.т.н. доц.

О. Л. Заміховська

« _____ » _____ 2026 року

З А В Д А Н Н Я

НА БАКАЛАВРСЬКУ РОБОТУ СТУДЕНТОВІ

Кобльовському Сергію Юрійовичу

1. Тема роботи Розроблення програмного модуля автоматизованої класифікації стану штангових глибинно-насосних установок на основі інтелектуального аналізу динамограм

керівник роботи Штаєр Лідія Омелянівна, к.т.н.

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом закладу вищої освіти від “ 07 ” квітня 2026 року № 164/7

2. Строк подання студентом роботи 15.06.2026 р.

3. Вихідні дані до роботи Матеріали та результати отримані під час

проходження переддипломної практики, методи розробки програмного забезпечення, алгоритми та методичні вказівки.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

1 Аналіз методів автоматизованої класифікації стану штангових глибинно-насосних установок та обґрунтування архітектури інтелектуальної системи

2 Розроблення методичного та алгоритмічного забезпечення класифікації станів ШГНУ за формою динамограми

3 Реалізація програмного модуля автоматизованої класифікації стану ШГНУ та оцінка його ефективності

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

Узагальнена структура системи керування ШГНУ

Синтетичні динамограми

Матриці помилок алгоритмів класифікації на тестовій вибірці

Аналіз важливості складових вектора ознак

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1-3	Штаєр Л. О., к.т.н., доц. каф. ІТТС		

7. Дата видачі завдання листопад 2025 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів бакалаврської роботи	Термін виконання етапів роботи	Примітка
1	Аналіз методів автоматизованої класифікації стану штангових глибинно-насосних установок та обґрунтування архітектури інтелектуальної системи	01.11.2025-25.12.2025	Виконано
2	Розроблення методичного та алгоритмічного забезпечення класифікації станів ШГНУ за формою динамограми	26.12.2025-26.03.2026	Виконано
3	Реалізація програмного модуля автоматизованої класифікації стану ШГНУ та оцінка його ефективності	27.03.2026-30.05.2026	Виконано
4	Оформлення результатів роботи	01.06.2026-15.06.2026	Виконано

Студент

Кобльовський С. Ю.

(підпис)

(прізвище та ініціали)

Керівник роботи

Штаєр Л. О.

(підпис)

(прізвище та ініціали)

РЕФЕРАТ

Пояснювальна записка містить 69 сторінок, 11 рисунків, 16 джерел за переліком посилань.

Об'єкт дослідження – процес автоматизованої діагностики технічного стану штангових глибинно-насосних установок (ШГНУ) нафтовидобувних свердловин.

Мета роботи – підвищення ефективності та надійності автоматизованої діагностики ШГНУ шляхом розроблення методу класифікації динамограм на основі алгоритмів машинного навчання, навчених на фізично обґрунтованих синтетичних даних зі стохастичними збуреннями.

Результатом роботи є розроблення програмного модуля інтелектуальної класифікації станів ШГНУ, який включає підсистему генерації зашумлених еталонних динамограм, алгоритм конструювання інженерних метаознак та оптимізовану модель машинного навчання для ідентифікації прихованих дефектів насосного обладнання в режимі реального часу.

У першому розділі проведено аналіз предметної області, огляд існуючих технологічних рішень та методів інтерпретації динамограм ШГНУ. Сформовано загальну архітектуру розроблюваної інтелектуальної системи.

У другому розділі представлено розроблене методичне, математичне та алгоритмічне забезпечення.

У третьому розділі здійснено експериментальні дослідження розробленого модуля. Проведено порівняльний аналіз алгоритмів класифікації. Виконано візуалізацію простору ознак (t-SNE) та кількісну оцінку інформативності складових сформованого вектору ознак.

Ключові слова: ШТАНГОВА ГЛИБИННО-НАСОСНА УСТАНОВКА, ДИНАМОГРАМА, МАШИННЕ НАВЧАННЯ, МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ, ВЕКТОР ОЗНАК.

ABSTRACT

The explanatory note contains 69 pages, 11 figures, and 16 references in the bibliography.

The object of research is the process of automated technical condition diagnostics of sucker-rod pumping units (SRPUs) in oil wells.

The purpose of the work is to improve the efficiency and reliability of automated SRPU diagnostics by developing a dynamometer card classification method based on machine learning algorithms trained on physically grounded synthetic data with stochastic disturbances.

The result of the work is the development of a software module for the intelligent classification of SRPU conditions, which includes a subsystem for generating noisy reference dynamometer cards, an algorithm for engineering meta-features, and an optimized machine learning model for identifying hidden defects in pumping equipment in real time.

The first chapter analyzes the subject area, reviews existing technological solutions and methods for interpreting SRPU dynamometer cards. The overall architecture of the intelligent system being developed has been established.

The second chapter presents the developed methodological, mathematical, and algorithmic support.

The third chapter deals with the experimental studies of the developed module. A comparative analysis of classification algorithms has been carried out. A visualization of the feature space (t-SNE) and a quantitative assessment of the information content of the formulated feature vector components have been performed.

Keywords: SUCKER-ROD PUMPING UNIT, DYNAMOMETER CARD, MACHINE LEARNING, MATHEMATICAL MODELING, FEATURE VECTOR.

ЗМІСТ

ВСТУП	8
1 АНАЛІЗ МЕТОДІВ АВТОМАТИЗОВАНОЇ КЛАСИФІКАЦІЇ СТАНУ ШТАНГОВИХ ГЛИБИННО-НАСОСНИХ УСТАНОВОК ТА ОБҐРУНТУВАННЯ АРХІТЕКТУРИ ІНТЕЛЕКТУАЛЬНОЇ СИСТЕМИ	10
1.1 Аналіз проблеми контролю технічного стану штангових глибинно-насосних установок.....	10
1.2 Динамограма як основне джерело інформації про стан ШГНУ	15
1.3 Огляд і аналіз існуючих методів розпізнавання динамограм.....	21
1.4 Постановка задачі на розроблення програмного модуля автоматизованої класифікації стану штангових глибинно-насосних установок на основі інтелектуального аналізу динамограм	24
Висновки до розділу 1	26
2 РОЗРОБЛЕННЯ МЕТОДИЧНОГО ТА АЛГОРИТМІЧНОГО ЗАБЕЗПЕЧЕННЯ КЛАСИФІКАЦІЇ СТАНІВ ШГНУ ЗА ФОРМОЮ ДИНАМОГРАМИ	27
2.1 Алгоритмічні основи синтезу еталонних динамограм для різних станів ШГНУ	27
2.2 Математичне моделювання завад	30
2.3 Формування вектору ознак для моделей машинного навчання.....	32
2.4 Вибір, конфігурація моделей машинного навчання для класифікації динамограм та критерії оцінки якості класифікації	34
Висновки до розділу 2	36

					КРБ.СІ-21.00.00.000 ПЗ			
Зм.	Арк.	№ докум.	Підпис	Дата				
Розроб.		Кобльовський.			Розроблення програмного модуля автоматизованої класифікації стану штангових глибинно-насосних установок на основі інтелектуального аналізу динамограм	Літ.	Арк.	Аркушів
Перевір.		Штаер Л. О.					6	69
Реценз.						ІФНТУНГ СІ-22-1		
Н. Контр.		Возний А.В.						
Затверд.		Заміховська						

3 РЕАЛІЗАЦІЯ ПРОГРАМНОГО МОДУЛЯ АВТОМАТИЗОВАНОЇ КЛАСИФІКАЦІЇ СТАНУ ШГНУ ТА ОЦІНКА ЙОГО ЕФЕКТИВНОСТІ	38
3.1 Оцінка надійності моделей на тестовій вибірці.....	38
3.2 Аналіз ефективності моделей з використанням алгоритму t-SNE та аналізу матриць помилок.....	40
3.3 Дослідження інформативності складових вектора ознак	45
Висновки до розділу 3	49
ВИСНОВКИ.....	51
ПЕРЕЛІК ПОСИЛАНЬ НА ДЖЕРЕЛА	53
Додаток А Програмний код синтезу динамограм з шумом.....	56
Додаток Б Програмний код модуля інженерії ознак	60
Додаток В Програмний код модуля порівняння ефективності алгоритмів класифікації станів ШГНУ	62
Додаток Г Програмний код модуля аналізу важливості складових вектору ознак.....	67

ВСТУП

Актуальність теми. Штангові глибинно-насосні установки (ШГНУ) є найбільш поширеним у світі способом механізованого видобутку нафти. Економічна рентабельність нафтових родовищ критично залежить від безперебійної роботи цього обладнання. Несвоєчасне виявлення таких дефектів, як витоки в клапанах, газові блокування чи удари плунжера, призводить до прискореного зносу обладнання, аварійних зупинок та значних фінансових втрат. Традиційна діагностика технічного стану ШГНУ здійснюється шляхом візуального аналізу динамограм (графіків залежності навантаження від переміщення штока) оператором або за допомогою простих евристичних алгоритмів. Проте в умовах реальних промислових IoT-мереж динамограми зазнають суттєвих стохастичних спотворень: високочастотних вібрацій, температурного дрейфу нуля датчиків, втрати пакетів даних та фазового зсуву. За таких умов класичні методи втрачають свою ефективність.

Вирішенням цієї проблеми є впровадження систем предиктивного обслуговування на базі штучного інтелекту та машинного навчання (ML). Однак їх розробка сповільнюється дефіцитом відкритих і збалансованих промислових баз даних, які б містили достатню кількість прикладів аварійних режимів. Тому розроблення комплексного методу, який поєднує фізичне моделювання динамограм, генерацію завад та навчання нелінійних ML-алгоритмів для автоматизованої діагностики ШГНУ, є актуальним науково-прикладним завданням.

Мета дослідження — підвищення ефективності та надійності автоматизованої діагностики ШГНУ шляхом розроблення методу класифікації динамограм на основі алгоритмів машинного навчання, навчених на фізично обґрунтованих синтетичних даних зі стохастичними збуреннями.

Ідея роботи полягає у поєднанні фізичного моделювання хвильових процесів у колоні штанг (для подолання проблеми дефіциту даних) із

					КРБ.СІ - 21.00.00.000 ПЗ	Арк.
						8
Змн.	Арк.	№ докум.	Підпис	Дата		

конструюванням вектору ознак та використанням нелінійних ансамблевих і нейромережових алгоритмів для створення системи діагностики, стійкої до інтенсивних промислових завад телеметрії.

Об'єкт дослідження — процес автоматизованої діагностики технічного стану штангових глибинно-насосних установок (ШГНУ) нафтовидобувних свердловин.

Предмет дослідження — методи, алгоритми та моделі машинного навчання для класифікації просторово-часових форм динамограм в умовах стохастичних спотворень.

Методи дослідження: математичне та імітаційне моделювання, теорія ймовірностей та математична статистика, методи машинного навчання та штучного інтелекту, методи оцінки ефективності алгоритмів.

					КРБ.СІ - 21.00.00.000 ПЗ	
Змн.	Арк.	№ докум.	Підпис	Дата		9

1 АНАЛІЗ МЕТОДІВ АВТОМАТИЗОВАНОЇ КЛАСИФІКАЦІЇ СТАНУ ШТАНГОВИХ ГЛИБИННО-НАСОСНИХ УСТАНОВОК ТА ОБҐРУНТУВАННЯ АРХІТЕКТУРИ ІНТЕЛЕКТУАЛЬНОЇ СИСТЕМИ

1.1 Аналіз проблеми контролю технічного стану штангових глибинно-насосних установок

Штангові глибинно-насосні установки (ШГНУ) є основним і найбільш поширеним методом механізованого видобутку нафти у світі. Оскільки за допомогою цього методу видобувається основний обсяг нафти, забезпечення надійності та ефективності таких установок має критичне значення для економічної діяльності нафтових родовищ [1].

Конструктивно ШГНУ поділяється на два взаємопов'язані комплекси: наземне обладнання (індивідуальний механічний привід та гирлове обладнання) і підземне (свердловинне) обладнання (рис. 1.1). Наземний комплекс обладнання включає такі ключові елементи:

- електродвигун – виступає первинним джерелом енергії, який передає обертовий момент через пасові передачі на редуктор;

- редуктор та кривошипно-шатунний механізм – призначені для зниження частоти обертання та перетворення обертового руху вала двигуна у зворотно-поступальний рух балансира;

- балансир із поворотною головкою – здійснює коливальні рухи навколо своєї опори. На головці балансира закріплено гнучку канатну підвіску;

- полірований (гирловий) шток – верхня посилена частина штангової колони, яка проходить крізь гирлове обладнання;

- гирловий сальник (сальниковий пристрій) – забезпечує герметизацію гирла свердловини у місці виходу полірованого штока, запобігаючи витoku нафти та газу в навколишнє середовище;

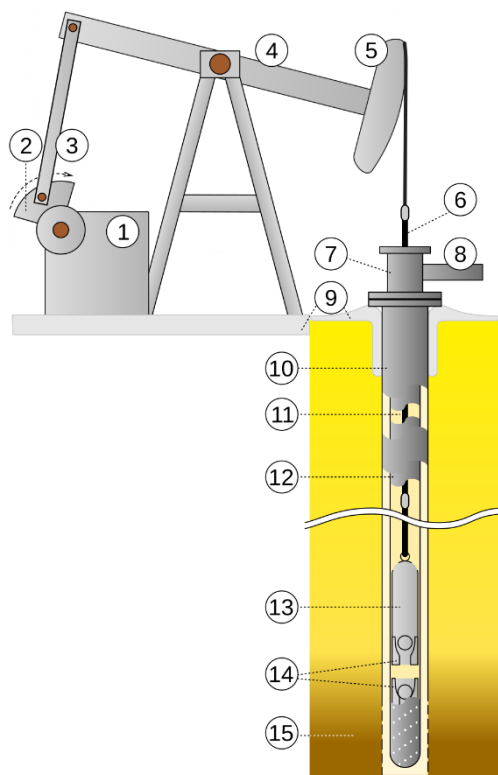


Рисунок 1.1 – Штангова насосна установка [2]: 1 – двигун; 2 – противага; 3 – кермова сошка; 4 – балка балансира; 5 – головка балансира; 6 – верхній шарнір колони штанг; 7 – гирло свердловини; 8 – нафтопровід; 9 – бетонні фундаменти; 10 – кожух; 11 – штанга насосна; 12 – трубки; 13 – насос; 14 – клапани; 15 – нафтові піски

— гирловий трійник – спрямовує потік видобутої нафтогазової суміші зі свердловини у викидну лінію та далі до промислового колектора.

Підземний (свердловинний) комплекс обладнання складається з таких компонентів:

— насосно-компресорні труби (НКТ) – колона труб, яка спускається у свердловину і слугує каналом для підйому рідини на поверхню, а також основою для кріплення циліндра насоса;

— колона насосних штанг – багатокілометрова зв'язка послідовно з'єднаних між собою штангових елементів, що розміщується всередині НКТ і виконує роль проміжної механічної ланки для передачі механічної енергії від головки балансира до підземного насоса;

— свердловинний штанговий насос – об’ємний гідроліфтовий апарат, який встановлюється під динамічним рівнем рідини. Насос складається з нерухомого циліндра, всередині якого здійснює зворотно-поступальний рух прецизійний плунжер;

— клапанна система насоса – включає два зворотні клапани: нерухомий всмоктувальний (стоячий) клапан, розташований у нижній частині циліндра, та рухомий нагнітальний (попутний) клапан, змонтований безпосередньо у верхній частині плунжера.

Робота установки за схемою має чітко виражений циклічний характер і складається з двох основних тактів: ходу вгору та ходу вниз.

Хід вгору: головка балансира піднімає полірований шток і колону штанг разом із плунжером. Під дією ваги стовпа рідини, що знаходиться в НКТ над плунжером, нагнітальний клапан щільно закривається. Рідина, що заповнює НКТ, витісняється через гирловий трійник у трубопровід. Одночасно з цим у порожнині циліндра під плунжером створюється розрідження (вакууметричний тиск). Під дією тиску стовпа рідини у свердловині (вибійного тиску) всмоктувальний клапан відкривається, і пластовий флюїд заповнює звільнений простір циліндра. На цьому такті колона штанг зазнає максимального розтягувального навантаження, оскільки вона утримує як власну вагу, так і вагу стовпа рідини над плунжером.

Хід вниз: головка балансира рухається вниз, опускаючи штанги та плунжер. Всмоктувальний клапан під дією тиску рідини всередині циліндра та сили власної ваги кульки закривається, перекриваючи зворотний хід флюїду у свердловину. Тиск у камері насоса зростає, що змушує відкритися нагнітальний клапан на плунжері. Плунжер занурюється у рідину, і вона перетікає з підплунжерного простору в надплунжерний (у колону НКТ). На цьому такті навантаження на колону штанг різко зменшується, оскільки вага стовпа рідини переноситься з плунжера через циліндр на колону НКТ.

Широке застосування ШГНУ в нафтогазовій промисловості зумовлене їхніми техніко-експлуатаційними характеристиками. До основних переваг належать відносно нескладний монтаж обладнання та доволі високий коефіцієнт корисної дії. Проте експлуатація установок супроводжується низкою суттєвих проблем, серед яких підвищений знос ключових підземних елементів — плунжерної пари, насосно-компресорних труб та штанг насоса. Крім того, недоліком є загальна складність проведення ремонту глибинного обладнання [1,3].

Складні умови експлуатації та схильність елементів ШГНУ до зносу зумовлюють необхідність своєчасного діагностування та автоматизації контролю технічного стану установок і режимів їх роботи. Систематичний контроль технічного стану обладнання безпосередньо впливає на ефективність та загальну безпеку експлуатації нафтових свердловин [1,3].

Створення систем моніторингу вирішує наступні ключові завдання:

— мінімізація простоїв: забезпечення швидкої діагностики, зводить до мінімуму час простою установки і дозволяє оператору своєчасно виконати необхідні коригувальні дії;

— економічна ефективність: розроблення точних і надійних систем діагностування несправностей дозволяє своєчасно виявити збої та знизити фінансові витрати;

— запобігання аваріям: виявлення дефектів на ранніх етапах запобігає відмовам компонентів системи та унеможливорює подальші пошкодження глибинного обладнання.

Для реалізації завдань безперервної діагностики та оптимізації видобутку сучасні нафтогазові підприємства впроваджують багаторівневі автоматизовані системи керування (АСК). Узагальнена структура системи керування та діагностики ШГНУ представлена на рисунку 1.2 [1].



Рисунок 1.2 - Узагальнена структура системи керування ШГНУ

Як видно зі схеми (рис. 1.2), архітектура системи є ієрархічною і складається з трьох базових рівнів:

— польовий рівень: охоплює безпосередньо технологічний об'єкт — свердловину, наземний привід (верстат-качалку) та комплект первинних перетворювачів (давачів). Ключову роль тут відіграють давачі навантаження (динамографи), які встановлюються на полірованому штоку, та давачі переміщення балансира. Їхнє завдання — безперервна реєстрація зусилля в колоні штанг залежно від її поточного положення. Також на цьому рівні

знаходяться датчики параметрів електродвигуна (струм, напруга) та тиску на гирлі;

— рівень локального контролера: базується на використанні спеціалізованих програмованих логічних контролерів (контролери керування відкачуванням РОС (Pump-Off Controller)). Контролер виконує первинне перетворення аналогових сигналів від датчиків у цифрову форму, формує масив точок динамограми та реалізує базові алгоритми захисту (наприклад, автоматичну зупинку двигуна у разі обриву штанг або критичного недонаповнення насоса). Зібрані та первинно оброблені дані передаються на верхній рівень за допомогою бездротових (GSM/LTE/Radio) або провідних каналів зв'язку;

— верхній рівень: централізована диспетчерська система збору та обробки даних (SCADA-система), яка розгорнута на локальних серверах підприємства або у хмарному середовищі. На цьому рівні здійснюється накопичення історії роботи фонду свердловин у базах даних та візуалізація інформації для інженерів. Саме на цьому етапі доцільно розгортати модулі машинного навчання (ML) та штучного інтелекту. Оскільки локальні контролери мають обмежені обчислювальні потужності, глибокий інтелектуальний аналіз синтезованих динамограм, розпізнавання складних (зашумлених) патернів дефектів та прогнозування технічного стану обладнання виконується високопродуктивними серверами верхнього рівня.

Таким чином, розробка методів діагностики є як технічним покращенням, так і економічно обґрунтованою необхідністю, що забезпечує безперебійний цикл механізованого видобутку нафти.

1.2 Динамограма як основне джерело інформації про стан ШГНУ

Основним, найбільш інформативним та економічно доступним методом неруйнівного контролю і предиктивної діагностики штангових глибинно-

					КРБ.СІ - 21.00.00.000 ПЗ	15
Змн.	Арк.	№ докум.	Підпис	Дата		

насосних установок є динамометрування свердловин [4]. Фізична суть цього методу полягає у безперервній реєстрації та аналізі механічних зусиль, що діють у точці підвісу штанг, протягом одного повного кінематичного циклу роботи верстата-качалки (ходу вгору та ходу вниз). Графічне відображення цієї залежності у двовимірній системі координат «навантаження на полірований шток (P) – переміщення штока (S)» називається динамограмою або динамокартою [4]. Оскільки процес відкачування рідини є циклічним, траєкторія зміни навантаження утворює замкнену криву лінію, геометрична форма якої є унікальним інтегральним патерном (відбитком) стану всієї системи «пласт – свердловина – насос – колона штанг – наземний привід».

У сучасній нафтопромисловій практиці та теорії інтелектуального аналізу даних динамограми чітко поділяють на два основних типи, кожен з яких має свій фізичний зміст та область застосування:

— наземна (поверхнева) динамограма: це первинний телеметричний сигнал, який безпосередньо реєструється давачами на усті свердловини (на рівні полірованого штока). Вона відображає реальні сили, що діють на наземний привід. Проте, як продемонстрував у своїх класичних працях С. Гіббс, форма наземної динамограми зазнає сильного викривлення через складні хвильові процеси [5]. На сигнал, що надходить з глибини, накладаються пружні деформації багатокілометрової колони штанг, сили інерції, гідродинамічний опір та коливальні процеси у стовпі рідини. Через це пряма візуальна інтерпретація наземних карт для виявлення дефектів безпосередньо в насосі є ускладненою.

— глибинна (плунжерна/насосна) динамограма: відображає чисту залежність «навантаження – переміщення» безпосередньо на рівні плунжера свердловинного насоса. Оскільки занурений насос працює в умовах відносно жорсткого циліндра, його ідеальна (еталонна) динамограма за умов нормальної роботи має чітку форму паралелограма [6]. Отримання глибинної динамограми здійснюється зазвичай розрахунковим шляхом за допомогою розв'язання оберненої крайової задачі математичної фізики для одновимірного

диференціального рівняння в частинних похідних гіперболічного типу (хвильового рівняння з урахуванням згасання коливань та сил тертя), що вперше було запропоновано у 1960-х роках [5]. Глибинна динамограма дозволяє «очистити» сигнал від впливу колони штанг та ідентифікувати дефекти самого глибинного клапанного вузла з набагато вищою точністю.

Приклади глибинних динамограм при типових станах ШГНУ показано на рис. 1.3.

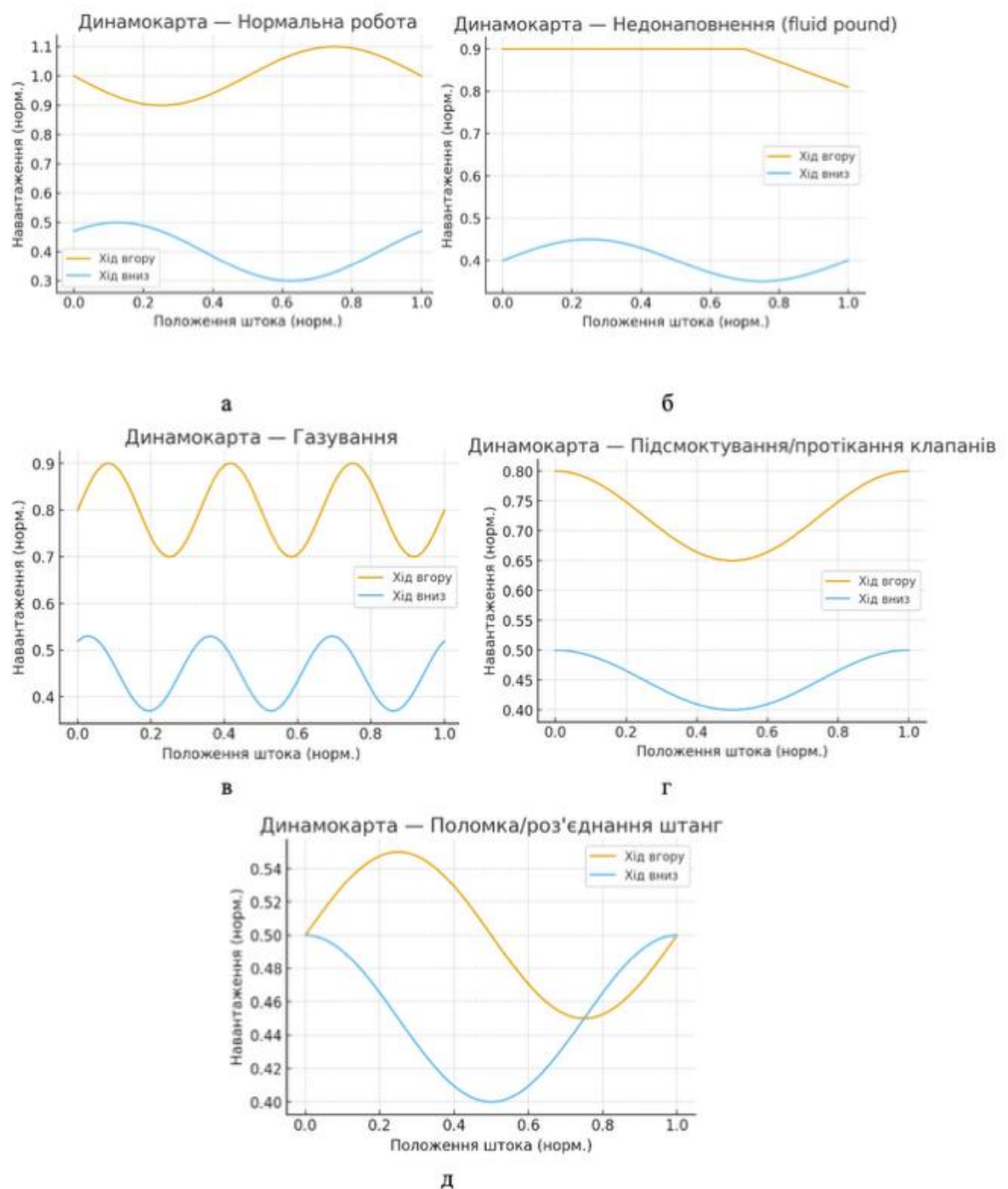


Рисунок 1.3 – Динамограми при типових станах ШГНУ

Математичний та геометричний аналіз форми динамограми передбачає виділення низки ключових кількісних параметрів, які використовуються як інформативні ознаки в алгоритмах класифікації (приклади спотворень форми динамограм показано на рис. 1.4):

— максимальне навантаження (P_{max}): фіксується під час ходу вгору, коли штанги сприймають вагу стовпа рідини та власну вагу в рідині. Перевищення критичного рівня свідчить про заклинювання або аномальне тертя [7];

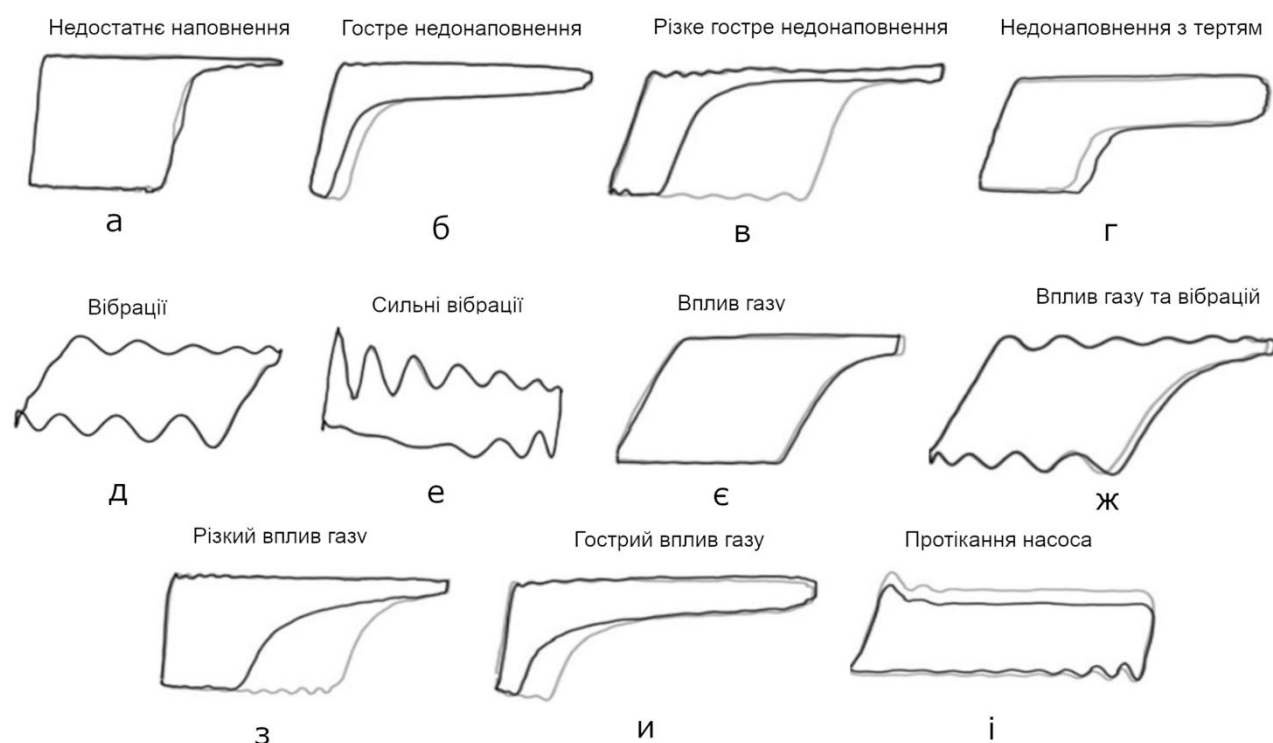


Рисунок 1.4 - Динамограми при різних типових дефектах ШГНУ

— мінімальне навантаження (P_{min}): фіксується під час ходу вниз, коли вага рідини переноситься на НКТ. Аномальне падіння P_{min} до нуля або від'ємних значень може вказувати на удар плунжера об рідину або посадку штанг [4];

— довжина ходу штока (S): визначає амплітуду коливань кривої по осі абсцис;

— площа динамограми (W): інтеграл навантаження за переміщенням, який фізично чисельно дорівнює механічній роботі, виконаній установкою за

один крок ходу. Зміна площі є індикатором зміни коефіцієнта наповнення насоса та його подачі [8];

— кути нахилу ліній сприйняття та скидання навантаження: геометрія ліній у лівій та правій частинах динамокарти відображає швидкість відкриття та закриття всмоктувального і нагнітального клапанів, що є критично важливим для предиктивної діагностики витоків [6,9].

Будь-яке відхилення від нормального режиму роботи свердловинного обладнання викликає перерозподіл сил у системі, що миттєво змінює геометрію замкненого контуру динамокарти. Графічні шаблони типових несправностей підземної частини ШГНУ наведено на рис. 1.4 та пояснено в табл. 1.1.

Таблиця 1.1 – Класифікація технічних станів ШГНУ за геометричними патернами динамограм

№	Технічний стан / Дефект	Геометричний патерн (форма динамограми)	Фізична суть процесу
1	Нормальна робота (ідеальний стан)	Правильний паралелограм з чіткими кутами. Лінії сприйняття (ліворуч) та скидання навантаження (праворуч) є прямими і паралельними.	Нормальний цикл відкачування. Кут нахилу бічних ліній визначається виключно теоретичною пружною деформацією штанг.
2	Недостатнє наповнення насоса (вплив низького дебіту)	Крива ходу вниз спочатку повторює нормальну лінію, але в середині ходу різко вертикально падає (правий верхній кут «відсікається»).	Плунжер рухається у напівпорожньому циліндрі і здійснює жорсткий гідравлічний удар об рівень рідини [4]
3	Газова інтерференція (вплив вільного газу)	Лінія ходу вниз падає не стрімко, а утворює плавну увігнуту гіперболічну криву (закруглення правого кута).	Відбувається стиснення газової «подушки», що виділилася з нафти. Скидання навантаження йде із запізненням, знижуючи ефективний хід плунжера [6]
4	Витік у нагнітальному (рухомому) клапані	Верхня лінія (хід вгору) втрачає горизонтальність і плавно згинається донизу.	Через негерметичність рідина під високим тиском перетікає крізь плунжер назад у циліндр, частково

		Правий верхній кут стає округлим.	розвантажуючи штанги ще до початку ходу вниз [9]
5	Витік у всмоктувальному (стоячому) клапані	Нижня лінія (хід вниз) згинається догори, а лівий нижній кут утворює плавне закруглення.	При опусканні штанг закритий клапан пропускає рідину в пласт. Тиск під плунжером падає повільніше, вага рідини не може повністю перенестися на НКТ [7]
6	Обрив колони насосних штанг	Вузька, витягнута по горизонталі петля або тонка стрічка в нижній частині графіка (поблизу лінії мінімального статичного навантаження).	Зусилля на штоку залишається постійним і мінімальним протягом усього циклу, оскільки привід переміщує лише залишок обірваної колони [4]
7	Аномально високе тертя / висока в'язкість	Контур рівномірно розширюється по вертикалі (лінія ходу вгору зміщується вище, а ходу вниз — нижче), утворюючи «роздуту» форму.	Виникають значні додаткові сили опору під час руху штанг всередині НКТ (через викривлення стовбура або відкладення парафіну) [1]
8	Удар плунжера об стоячий клапан / посадка штанг	Різкі, високочастотні амплітудні коливання (вертикальні «шпильки» або зазубрини) у найнижчій точці ходу.	Виникають хвилі механічних напружень внаслідок фізичного зіткнення підземних металевих конструкцій під час зміни напрямку руху [8]

Головна складність практичного застосування динамометричного контролю полягає в тому, що на реальних нафтових промислах сигнали давачів ніколи не бувають ідеальними [4,10]. На форму кривої накладаються випадкові промислові завади, вібрації наземного приводу, нестабільність напруги живильної мережі електродвигуна, випадкові викиди через гідродинамічні збурення або пульсації тиску у викидній лінії [1].

Внаслідок цього реальна промислова динамограма є сильно зашумленою. Традиційні експертні системи, що спираються на жорсткі аналітичні правила або геометричні шаблони, часто дають збої або помилкові спрацьовування при появі шумів різної потужності. Цей фактор обумовлює високу актуальність

розроблення зашумлених синтетичних моделей для навчання гнучких стохастичних моделей штучного інтелекту, здатних ефективно фільтрувати корисний сигнал та безпомилково класифікувати стан ШГНУ в умовах реальних промислових перешкод.

1.3 Огляд і аналіз існуючих методів розпізнавання динамограм

Автоматизація розпізнавання та класифікації технічного стану ШГНУ за формою динамограми пройшла еволюцію від суб'єктивного візуального аналізу, що здійснювався інженерно-технічним персоналом безпосередньо на промислах, до розробки сучасних автоматизованих систем на базі штучного інтелекту та глибокого машинного навчання [3]. На сьогодні існуючі методи класифікації динамокарток можна розділити на три основні групи: традиційні аналітичні (експертні) методи, методи класичного машинного навчання з ручним проектуванням ознак та сучасні методи глибокого навчання [3,4].

Першими автоматизованими інструментами аналізу стали експертні системи, засновані на жорстких логічних правилах та детермінованих геометричних шаблонах [3,6]. В основі таких систем лежали фундаментальні праці С. Гіббса щодо математичного опису коливань колони штанг за допомогою хвильового рівняння [5], а також класичні методики візуальної інтерпретації [6,7]. Принцип дії таких систем базується на розрахунку набору реперних точок (мінімальне/максимальне навантаження, довжина ходу, площа) та порівнянні отриманого контуру з еталонними геометричними фігурами (паралелограм, трапеція, лінійна петля). Незважаючи на високу швидкість роботи, традиційні аналітичні системи мають суттєві недоліки. Вони мають низьку адаптивність і вимагають постійної ручної корекції порогових значень для кожної конкретної свердловини через унікальні геолого-технічні умови. Вони виявляються нездатними коректно працювати в умовах реальних промислових перешкод. Як зазначають в [1,4] наявність високочастотних шумів, вібрацій приводу та

випадкових викидів призводить до того, що жорсткі аналітичні правила дають велику кількість помилкових спрацьовувань або взагалі не розпізнають спотворений дефект.

З розвитком обчислювальної техніки фокус досліджень змістився у бік використання методів класичного машинного навчання. Для подолання чутливості до шумів дослідники почали застосовувати попередню математичну обробку та параметризацію замкненого контуру динамограми [11] – було запропоновано перетворення графічного образу динамокарти у стійкі вектори цифрових ознак. Для цього використовувалися такі математичні інструменти, як коефіцієнти розкладу в ряди Фур'є, інтегральні інваріанти кривих, моменти інерції геометричної фігури та методи дефектної декомпозиції сигналу [4,8,11]. Сформовані таким чином вектори геометричних ознак у подальшому подавалися на вхід класичних класифікаторів машинного навчання, таких як:

- метод опорних векторів (SVM): ефективно розділяє класи станів у просторі ознак високої розмірності;
- приховані моделі Маркова (HMM): застосовуються для аналізу динамограми як послідовності часових інтервалів, що дозволяє враховувати динаміку зміни сигналів під час циклу [4];
- випадкові ліси (Random Forest) та метод k -найближчих сусідів (k -NN): використовуються для багатокласової класифікації.

Використання класичного ML дозволило значно підвищити стійкість алгоритмів до зашумлення сигналів [3]. Проте цей підхід має критичне обмеження — етап виділення ознак виконується вручну. Якщо інженер-розробник не врахував певний геометричний інваріант або стохастичну деформацію кривої, точність класифікації моделі різко падає.

Сучасним етапом розвитку систем технічної діагностики ШГНУ є впровадження глибоких штучних нейронних мереж (DNN), зокрема згорткових нейронних мереж (CNN). Головна перевага глибокого навчання полягає в

автоматичному витягуванні ієрархічних просторових та текстурних ознак безпосередньо з сирих даних без необхідності ручного проектування ознак [3].

У таких підходах динамограма може розглядатися у двох форматах [3]:

– як двовимірне бінарне або градієнтне зображення: графік контуру перетворюється на матрицю пікселів фіксованого розміру, яка обробляється стандартними архітектурами CNN (LeNet, ResNet, VGG) аналогічно до задач комп'ютерного зору;

– як послідовність двовимірних сигналів: масиви точок навантаження та переміщення (P_i, S_i) обробляються одновимірними згортками (1D-CNN) або рекурентними мережами (LSTM/GRU), що дозволяє зберігати часову послідовність кроків [4].

Нейромережеві класифікатори демонструють найвищу точність розпізнавання дефектів підземного обладнання на великих обсягах даних, успішно ігноруючи дрібні випадкові коливання та спотворення форми кривої.

Попри високий потенціал моделей глибокого навчання (CNN), їхнє успішне практичне впровадження в реальних промислових системах моніторингу (таких як WellPilot [12], XSPOC [13] або SAM Well Manager [14]) стикається із фундаментальним бар'єром — проблемою дефіциту та дисбалансу навчальних даних. У реальних виробничих умовах нафтових промислів понад 90–95% часу свердловини працюють у штатному режимі. Наземні контролери безперервно реєструють тисячі динамограм нормального стану обладнання. Водночас динамокарти, що відповідають критичним аварійним ситуаціям або рідкісним дефектам (наприклад, обрив колони штанг, сильний витік стоячого клапана, піщаний корок або вибійне газове блокування), становлять менше 1–3% від загального обсягу промислових баз даних [15]. Крім того, отримання розмічених реальних даних про аварії ускладнене тим, що оператори намагаються негайно зупинити свердловину при підозрі дефекту для запобігання руйнуванню обладнання, не встигаючи накопичити статистичну вибірку сигналів. Як вказано в [15], навчання глибоких нейромереж на таких

незбалансованих вибірках призводить до ефекту перенавчання: модель ідеально розпізнає нормальну роботу, але повністю втрачає здатність ідентифікувати аварійні стани.

Рішенням цієї проблеми, яке лежить в основі даної бакалаврської роботи, є розробка математичних та програмних моделей синтезу динамограм [3,15]. Написаний у межах роботи програмний модуль мовою Python дозволяє за допомогою чисельного моделювання згенерувати необмежену та абсолютно збалансовану кількість еталонних динамокарт для вказаного класу дефектів. Проте ідеальні синтетичні криві не відображають реальний "зашумлений" промисловий сигнал. Тому важливим науково-практичним кроком даного дослідження є алгоритмічне моделювання та накладання на синтезовані динамограми випадкових шумів різної природи, конфігурації та амплітудної потужності. Це дозволяє штучно наблизити згенерований датасет до реальних умов експлуатації свердловин, забезпечуючи високу стійкість та ефективність навченої інтелектуальної моделі під час її подальшої роботи на реальних нафтогазових родовищах.

1.4 Постановка задачі на розроблення програмного модуля автоматизованої класифікації стану штангових глибинно-насосних установок на основі інтелектуального аналізу динамограм

Аналіз предметної області та сучасних технологічних рішень (п. 1.1–1.3) засвідчив, що впровадження методів інтелектуального аналізу динамограм є найбільш перспективним напрямом для підвищення ефективності експлуатації ШГНУ. Традиційні експертні системи, засновані на жорстких математичних правилах, виявляються недостатньо стійкими до реальних промислових шумів. Водночас використання сучасних методів машинного навчання (ML) та глибоких нейронних мереж (DNN) значно обмежується проблемою дефіциту та значного дисбалансу навчальних даних.

					КРБ.СІ - 21.00.00.000 ПЗ	
Змн.	Арк.	№ докум.	Підпис	Дата		24

Для подолання цих обмежень виникає науково-прикладна необхідність у створенні спеціалізованого програмного інструментарію. Цей інструментарій повинен класифікувати динамограми за допомогою алгоритмів машинного навчання та самостійно генерувати (синтезувати) збалансовані навчальні вибірки, максимально наближені до реальних промислових умов шляхом накладання стохастичних шумів різної природи.

Метою роботи є розроблення програмного модуля (на базі мови програмування Python), здатного автоматизовано синтезувати еталонні та зашумлені динамограми ШГНУ, а також здійснювати класифікацію технічних станів підземного обладнання за допомогою навчання моделі машинного навчання.

Об'єктом дослідження є процес автоматизованої технічної діагностики підземного обладнання штангових глибинно-насосних установок.

Предметом дослідження є методи математичного моделювання (синтезу) динамограм, алгоритми генерації завад та моделі машинного навчання для класифікації просторово-часових сигналів.

Для досягнення поставленої мети в межах даної роботи необхідно вирішити такі завдання:

- розробити алгоритмічне забезпечення для синтезу еталонних динамограм. Програмний модуль повинен генерувати базові (ідеальні) геометричні контури динамограм для нормальної роботи та для типових дефектів;

- розробити модуль генерації та накладання шумів. Для наближення синтетичних даних до реальних телеметричних сигналів необхідно реалізувати математичні алгоритми накладання перешкод (білий гаусівський шум, випадкові викиди амплітуди, гармонійні вібрації тощо). Модуль повинен дозволяти керувати рівнем зашумлення;

- сформувати навчальні та тестові набори даних. Забезпечити автоматичну генерацію великих, збалансованих наборів даних (включаючи

графічні зображення динамокарт), придатних для подачі на вхід ML-алгоритмів, із відповідною їх попередньою обробкою (нормалізація, масштабування);

- спроектувати та навчити класифікатор машинного навчання. Обрати оптимальну архітектуру ML-моделі. Здійснити навчання моделі на згенерованих даних з використанням відповідних функцій втрат та оптимізаторів;

- провести експериментальну оцінку ефективності та стійкості моделі. Оцінити якість розпізнавання за стандартними метриками (Accuracy, Precision, Recall, F1-Score, побудова матриці помилок). Проаналізувати, як змінюється точність класифікації при поступовому збільшенні амплітуди (потужності) накладених шумів.

Реалізація програмного модуля має бути виконана з використанням сучасних інструментів для аналізу даних та машинного навчання:

- мова програмування: Python (версії 3.x) – галузевий стандарт для завдань Data Science;

- бібліотеки для математичного моделювання та маніпуляції даними: NumPy (для матричних обчислень) та Pandas (для формування датасетів);

- бібліотеки для генерації шумів та обробки сигналів: SciPy;

- бібліотеки машинного навчання: Scikit-learn.

- інструменти візуалізації: Matplotlib, Seaborn для відображення синтезованих динамограм та графіків процесу навчання.

Висновки до розділу 1

Проведений огляд літератури та аналіз предметної області підтверджують високу актуальність обраної теми. Сформульована постановка задачі визначає перехід від теоретичного аналізу (розділ 1) до практичної реалізації та вимагає детального розроблення математичних алгоритмів генерації сигналів, синтезу шумів та архітектури ML-моделі, що буде висвітлено в розділі 2.

					КРБ.СІ - 21.00.00.000 ПЗ	26
Змн.	Арк.	№ докум.	Підпис	Дата		

2 РОЗРОБЛЕННЯ МЕТОДИЧНОГО ТА АЛГОРИТМІЧНОГО ЗАБЕЗПЕЧЕННЯ КЛАСИФІКАЦІЇ СТАНІВ ШГНУ ЗА ФОРМОЮ ДИНАМОГРАМИ

2.1 Алгоритмічні основи синтезу еталонних динамограм для різних станів ШГНУ

Для розв'язання задачі класифікації технічних ШГНУ методами машинного навчання необхідна наявність репрезентативної та збалансованої навчальної вибірки. З огляду на дефіцит розмічених промислових даних щодо аварійних режимів, у даній роботі застосовано підхід синтезу еталонних динамограм на основі компонентно-орієнтованого фізичного моделювання [16].

В якості математичного ядра використано метод розв'язку одновимірного хвильового рівняння з демпфуванням (рівняння Гіббса), що описує поздовжні коливання пружної колони штанг [5]:

$$\frac{\partial^2 u}{\partial t^2} = a^2 \frac{\partial^2 u}{\partial x^2} - c \frac{\partial u}{\partial t} + g, \quad (2.1)$$

де $u(x, t)$ — переміщення перерізу колони штанг, a — швидкість поширення пружної хвилі в матеріалі, c — коефіцієнт еквівалентного в'язкого тертя (демпфування), g — прискорення вільного падіння.

Прямий аналітичний розв'язок диференційного рівняння (2.1) в частинних похідних для реальних умов зі складною геометрією свердловини та змінними граничними умовами є неможливим. Тому в роботі використано чисельний метод дискретизації, реалізований за допомогою відкритої Python-бібліотеки `ruscodyn`. У розробленому програмному забезпеченні (на базі середовища Python та бібліотеки `ruscodyn` [16]) установку подано як систему взаємопов'язаних зосереджених мас (Mass), пружно-в'язких елементів (SpringDamper) та зовнішніх зусиль (Force).

Згідно з методологією `ruscodyn`, неперервна колона штанг апроксимується еквівалентною системою із зосередженими параметрами. У рамках розробленого

					КРБ.СІ - 21.00.00.000 ПЗ	27
Змн.	Арк.	№ докум.	Підпис	Дата		

алгоритму (клас PumpSimulator) систему змодельовано за допомогою трьох базових абстракцій [16]:

— Mass (M_{rod}) — зосереджена еквівалентна маса рухомих частин колони штанг. Відповідає за інерційну складову ($F = ma$);

— SpringDamper (C_{rod}, D_{rod}) — пружно-в'язкий елемент, що моделює жорсткість колони за законом Гука та гідродинамічне тертя за допомогою демпфера;

— Force (F_{base}) — генератор зовнішнього зусилля, що діє на нижній кінець колони (плунжер) і залежить від гідростатичного тиску стовпа рідини та поточного технічного стану насоса.

Така топологія дозволяє автоматично формувати систему звичайних диференціальних рівнянь стану за допомогою матричних методів та розв'язувати її інтегратором solveDyn у часовій області.

Кінематика точки підвісу штанг (ТПШ) задається гармонійним законом руху, що імітує роботу кривошипно-шатунного механізму верстата-качалки:

$$x(t) = A \sin(2\pi ft),$$

де A — амплітуда ходу ТПШ, f — частота хитань. Для підвищення інваріантності майбутньої моделі машинного навчання ці параметри рандомізуються в межах $\pm 5\text{--}10\%$ від номінальних значень для кожного згенерованого циклу.

Динамічне навантаження на ТПШ формується як сума статичної ваги колони штанг (F_{rods}) та ваги стовпа рідини (F_{fluid}), яка прикладена до плунжера. Алгоритм синтезу різних станів базується на зміні логіки взаємодії плунжера з рідиною залежно від фази циклу (хід вгору/вниз) та швидкості руху плунжера. Згладжування розривних функцій у точках зміни напрямку руху реалізується через гіперболічний тангенс $\tanh(|v|/0,01)$.

Розроблено алгоритмічні правила для чотирьох базових класів:

— нормальна робота (normal). В ідеальному стані клапани працюють синхронно з рухом плунжера. На ході вгору ($v > 0$) плунжер сприймає вагу

рідини, на ході вниз ($v < 0$) навантаження знімається. Алгоритмічно це описується умовою:

— якщо $v > 0$, тоді $F = F_{rods} + F_{fluid}$;

— якщо $v < 0$, тоді $F = F_{rods}$.

Таким чином формується еталонна динамограма у вигляді наближеного паралелограма;

— удар плунжера об рідину (fluid pound). Цей стан виникає при неповному заповненні циліндра насоса рідиною через недостатній приплив зі свердловини. На початку ходу вниз плунжер рухається у газовому середовищі (зберігаючи частину навантаження через тертя), після чого відбувається різкий удар об рівень рідини. Для симуляції цього стану введено параметр важкості $S_{pound} \in [0,5; 0,95]$, який визначає момент удару. Алгоритмічно навантаження на ході вниз спадає не одразу:

— якщо $|\sin(2\pi t_{cycle})| > S_{pound}$, навантаження зберігається;

— в іншому випадку F різко зменшується до базової ваги штанг, формуючи на динамограмі характерну сходинку (провал);

— вплив газу (gas lock). При потраплянні вільного газу під плунжер відбувається його пружне стиснення (хід вниз) та розширення (хід вгору), що затримує момент відкриття клапанів. Алгоритм імітує цей процес шляхом плавної, а не стрибкоподібної зміни зусилля, використовуючи властивості функції $\tanh$:

— на ході вгору ($v > 0$): $F = F_{rods} + F_{fluid} \tanh(v/0,5)$;

— на ході вниз ($v < 0$): зусилля зменшується пропорційно $\tanh(v/0,5)$. Графічно це призводить до суттєвого "скруглення" кутів паралелограма динамограми;

— витік клапана (valve leak). негерметичність клапанного вузла призводить до часткової втрати стовпа рідини та перетікання речовини під час робочого ходу. У математичну модель введено коефіцієнт негерметичності $S_{leak} \in [0,1; 0,4]$ (від 10% до 40% втрати тиску):

					КРБ.СІ - 21.00.00.000 ПЗ	
Змн.	Арк.	№ докум.	Підпис	Дата		29

— на ході вгору зусилля зменшується: $F = F_{rods} + F_{fluid}(1 - S_{leak})$;
 — на ході вниз виникає зустрічний опір: $F = F_{rods} + F_{fluid} * 0,1 * S_{leak}$.
 Зміна цих параметрів генерує нахил верхньої та нижньої ліній динамограми, що є головною діагностичною ознакою витоку.

Таким чином, розроблений алгоритм дозволяє синтезувати математично точні еталони динамограм, які відображають фізичну природу процесів у свердловині і слугують базою для подальшого накладання стохастичних завад.

2.2 Математичне моделювання завад

Для забезпечення високої здатності моделі машинного навчання до узагальнення та стійкості до реальних промислових умов, еталонні динамограми було піддано процедурі комплексного зашумлення.

На відміну від стандартних підходів, де використовується лише базовий білий шум, у даній роботі розроблено алгоритм стохастичного спотворення, який діє на вже нормалізований вектор ознак (100 точок). Модель завад включає п'ять незалежних компонентів, кожен з яких імітує специфічні фізичні або апаратні дефекти системи телеметрії.

Математична модель спотвореного сигналу S_{final} для i -ї точки циклу має вигляд:

$$S_{final}[i] = (S_{ideal}[(i - \Delta_{phase})(\text{mod } N)]K_{scale} + Trend(i)) + Spike(i) + \Delta_{bias} + N_{gauss}(i), \quad (2.2)$$

де кожен компонент генерується програмним шляхом за наступною логікою:

— фазовий зсув (np.roll): циклічний зсув масиву даних на випадкову величину $\Delta_{phase} \in [-20, 20]$ точок. Імітує розсинхронізацію між датчиком положення (кінцевим вимикачем) та датчиком зусилля. У реальних системах цикл запису може початися не строго з нижньої чи верхньої мертвої точки. Алгоритмічно такий підхід змушує модель машинного навчання (зокрема згорткові мережі або дерева рішень) вивчати топологічну форму кривої, а не

прив'язуватися до абсолютних індексів колонок (що є головною вразливістю простої лінійної регресії);

— імпульсні викиди (*Spike*): з імовірністю 50% у випадкові точки масиву (від 1 до 3 точок) додаються гострі піки з амплітудою 15...30 кН випадкового знаку. Так моделюються короточасні збої АЦП контролера, втрату пакетів даних при бездротовій передачі (радіоканал або GSM) або різкі стрибки напруги в мережі живлення свердловин. Алгоритмічно це перевірка стійкості моделі до аномалій;

— трендовий нахил (*Trend*): додавання лінійного масиву pr.linspace , що монотонно зростає або спадає в діапазоні ± 5 кН від початку до кінця циклу. Імітує динамічну зміну рівня рідини у свердловині безпосередньо під час зняття динамограми або вплив температурної деградації тензомоста протягом кількох секунд вимірювання. Алгоритмічна мета: ускладнює класифікацію "Витоку клапана", оскільки штучний тренд може візуально викривити нормальну динамограму, роблячи її схожою на витік;

— дрейф нуля: адитивне додавання випадкової константи $\Delta_{bias} \in [-10, 10]$ кН до всього циклу. Статична помилка тарування датчика ваги внаслідок налипання парафіну на шток, зміни натягу ременів верстата-качалки або сезонних температурних коливань. Алгоритмічна мета: деградація ефективності алгоритмів, що спираються на абсолютні пороги навантаження;

— мультиплікативний та фоновий адитивний шум: множення на випадковий коефіцієнт $K_{scale} \in [0,9; 1,1]$ (похибка масштабу) та додавання білого гауссового шуму $N_{gauss} \sim N(0; 0,2)$. Фізичний зміст: високочастотні механічні вібрації колони штанг, ефект «тертя-ковзання» муфт об НКТ та стандартна похибка вимірювального тракту.

Приклад згенерованих синтетичних динамограм для 4х обраних типів дефектів ШГНУ разом із характерною не зашумленою динамограмою для кожного прояву дефекту на динамограмі наведено на рис. 2.1.

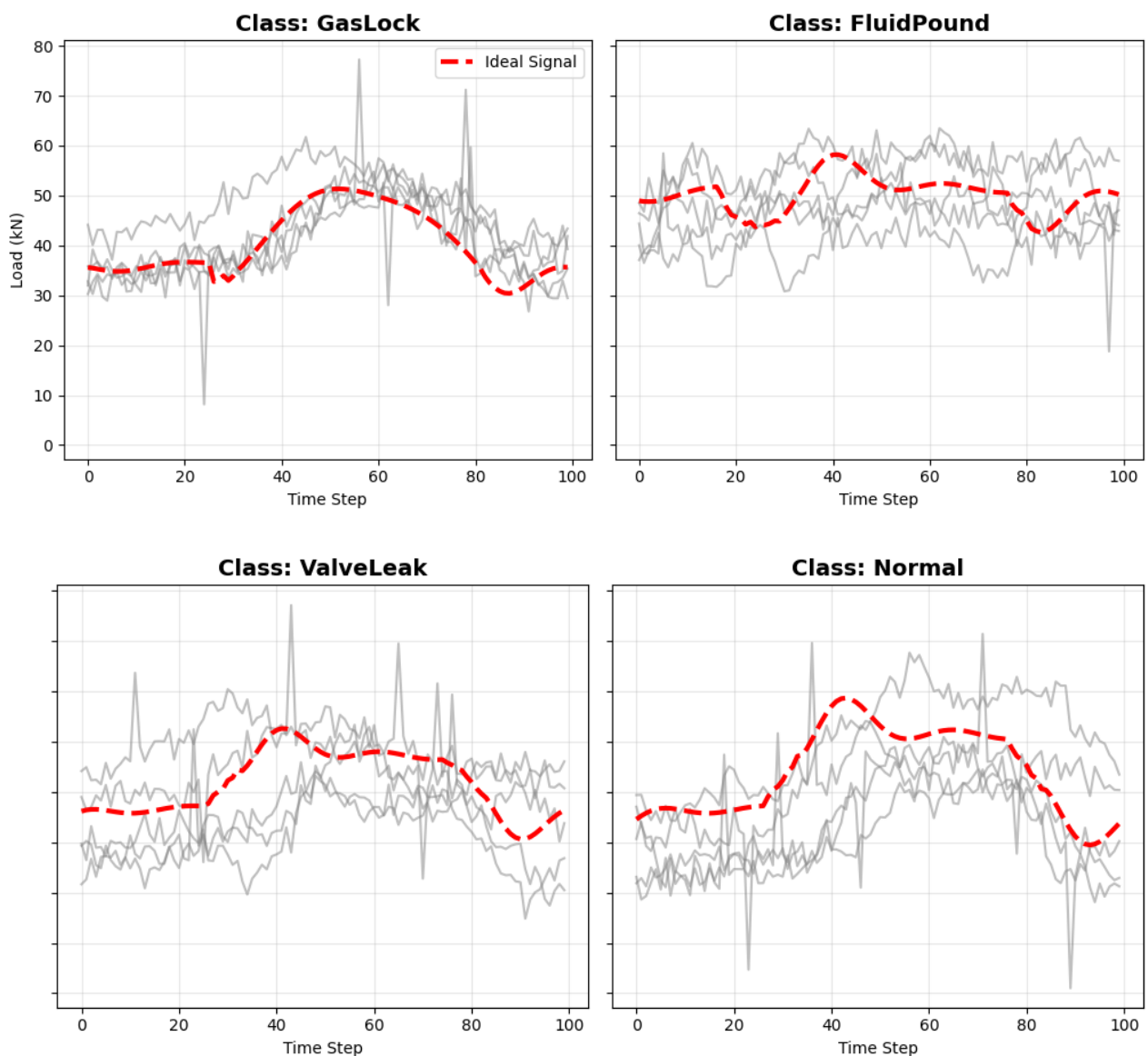


Рисунок 2.1 - Порівняння ідеальних динамограм та зашумлених сигналів

Код модуля, який генерує зображені на рис. 2.1 синтетичні динамограми для розглянутих дефектів ШГНУ наведено в додатку А.

2.3 Формування вектору ознак для моделей машинного навчання

Використання виключно "сирих" точок динамограми як вхідного вектора для алгоритмів машинного навчання часто призводить до проблеми великої розмірності та перенавчання, особливо в умовах сильного фазового зсуву та імпульсних завад.

Для підвищення роздільної здатності класифікаторів (зокрема моделей на основі дерев рішень та лінійних методів) було реалізовано процедуру конструювання ознак (інженерія ознак). З кожного нормалізованого циклу динамограми (точок) за допомогою бібліотеки SciPy розраховується вектор із 12ти метаознак, які фізично описують динаміку ШГНУ. Ознаки розділено на дві групи. Група 1 — статистичні та енергетичні характеристики. Ця група описує загальний розподіл навантаження та його хаотичність:

- площа динамограми: розрахована методом трапецій (`np.trapz`). Фізично відповідає корисній роботі, витраченій на підйом рідини за один цикл. При впливі газу (Gas Lock) площа суттєво зменшується;

- середнє та стандартне відхилення (`mean`, `std`): описують базовий рівень навантаження та ступінь його коливання;

- асиметрія та ексцес: характеризують відхилення форми сигналу від нормального розподілу. Наприклад, удар плунжера (Fluid Pound) створює різку асиметрію;

- розмах: різниця між максимальним та мінімальним навантаженням;

- енергія сигналу: сума квадратів амплітуд, що відображає загальну інтенсивність процесу;

- спектральна ентропія: розраховується на основі періодограми потужності. Висока ентропія вказує на "зашумленість" або хаотичність сигналу, що є маркером механічних ускладнень.

Група 2 — геометричні ознаки. Ця група розроблена спеціально для виявлення структурних змін форми динамограми, які важко виділити звичайною статистикою:

- коефіцієнт прямокутності: відношення реальної площі динамограми до площі описаного прямокутника. Для ідеальної (нормальної) динамограми цей коефіцієнт прямує до 1.0, тоді як при впливі газу він значно падає;

— нахил ходу вгору: коефіцієнт лінійної регресії, розрахований для ділянки точок (10-40). Дозволяє математично ідентифікувати втрату герметичності клапанів, оскільки витік змінює кут нахилу цієї гілки;

— нахил ходу вниз: регресія на ділянці (60-90). Діагностує стан нагнітального клапана;

— перекіс навантаження: відношення середнього зусилля першої третини циклу до останньої. Ознака є чутливою до запізненого сприйняття навантаження при відкачуванні газованої рідини.

Сформований таким чином розширений датасет дозволяє комбінувати топологічні ознаки (сірі точки для 1D-CNN) та фізико-математичні метаознаки (для XGBoost та Random Forest), що робить систему діагностики максимально стійкою до дестабілізуючих факторів.

Програмний код етапу формування ознак наведено в додатку Б.

2.4 Вибір, конфігурація моделей машинного навчання для класифікації динамограм та критерії оцінки якості класифікації

З огляду на специфіку вхідних даних (одновимірні часові ряди з накладеним інтенсивним шумом), для розв'язання задачі мультикласової класифікації станів ШГНУ було сформовано ансамбль з семи алгоритмів різної математичної природи. Це дозволяє провести об'єктивний порівняльний аналіз та виявити архітектуру, найбільш стійку до стохастичних спотворень.

Для встановлення базового рівня ефективності використано класичні алгоритми:

— логістична регресія: лінійний класифікатор, що оптимізує логістичну функцію втрат;

— метод опорних векторів (SVM): використано нелінійне радіально-базисне ядро (RBF) для проєктування даних у простір вищої розмірності;

— K-найближчих сусідів (KNN): метричний алгоритм, що класифікує зразки за відстанню в просторі ознак;

— багатошаровий перцептрон (MLP): нейронна мережа прямого поширення з двома прихованими шарами (100 та 50 нейронів відповідно).

Ансамблеві алгоритми на основі дерев рішень:

— Random Forest: ансамбль зі 100 незалежних дерев рішень, що знижує дисперсію прогнозів;

— XGBoost (Extreme Gradient Boosting): алгоритм градієнтного бустингу. Використовує метрику багатокласової логарифмічної втрати та оптимізовану побудову послідовності дерев для мінімізації похибки попередніх ітерацій.

Архітектура згорткової нейронної мережі (1D-CNN): оскільки динамограма є часовим рядом (послідовністю зі 100 значень), для просторового вилучення локальних ознак (паттернів нахилу, екстремумів) було розроблено архітектуру одновимірної згорткової нейромережі на базі фреймворку PyTorch. Мережа складається з двох згорткових блоків та повнозв'язного класифікатора:

— блок 1: одновимірна згортка (Conv1d: 1 вхідний канал, 16 вихідних фільтрів, розмір ядра $K = 5$, $\text{padding}=2$), нормалізація батчу (BatchNorm1d), функція активації ReLU, субдискретизація (MaxPool1d, $\text{stride} = 2$);

— блок 2: Conv1d (16 вхідних, 32 вихідних фільтри, $K = 5$), BatchNorm1d, ReLU, MaxPool1d(2);

— регуляризація: для запобігання перенавчанню перед лінійним шаром застосовано метод Dropout з імовірністю відключення нейронів $p = 0,3$;

— класифікатор: лінійний шар, що відображає згладжений тензор ознак у вектор імовірностей для 5 класів станів ШГНУ.

Для забезпечення коректності навчання та об'єктивності результатів було розроблено процедуру підготовки даних та валідації. Оскільки вхідні ознаки та згенерований шум можуть мати різний масштаб, до всього масиву даних застосовано стандартизацію:

$$z = \frac{x - \mu}{\sigma}, \quad (2.3)$$

де μ — середнє значення, σ — стандартне відхилення. Це необхідно для забезпечення збіжності метричних алгоритмів (SVM, KNN) та градієнтного спуску в нейронних мережах (MLP, 1D-CNN).

Навчання згорткової мережі здійснювалося протягом 100 епох. В якості функції втрат для багатокласової класифікації використано перехресну ентропію. Мінімізація функції втрат реалізована за допомогою адаптивного оптимізатора Adam зі швидкістю навчання $\alpha = 0,001$.

Масив даних розділено на навчальну (80%) та тестову (20%) вибірки. Для підтвердження статистичної значущості результатів (уникнення випадкових сплесків точності на вдалому спліті), окрім тестування на відкладеній вибірці, застосовано метод k -кратної перехресної перевірки (5-fold Cross-Validation).

Ефективність моделей оцінювалася за допомогою побудови матриць помилок (Confusion Matrix) та розрахунку двох головних метрик:

— Accuracy (загальна точність): відсоток правильно класифікованих динамограм серед загальної кількості;

— F1-Score (зважена F-міра): гармонічне середнє між точністю (Precision) та повнотою (Recall). Використання параметра `average='weighted'` дозволяє врахувати мікрозміни у балансі класів, забезпечуючи об'єктивну оцінку здатності моделі уникати помилок першого та другого роду.

Додатково, для аналізу ступеня лінійної нероздільності класів в умовах складних стохастичних завад, впроваджено метод стохастичного вкладення сусідів з t -розподілом (t-SNE), який проєктує багатовимірний простір ознак у двовимірну площину для візуалізації кластерів.

Висновки до розділу 2

У розділі виконано розроблення методичного, математичного та алгоритмічного забезпечення для автоматизованої класифікації технічних станів ШГНУ за формою динамограми. Отримано такі основні результати: обґрунтовано та реалізовано методику фізично адекватного синтезу еталонних

					КРБ.СІ - 21.00.00.000 ПЗ	
Змн.	Арк.	№ докум.	Підпис	Дата		36

динамограм на основі чисельного розв'язання одновимірного хвильового рівняння поздовжніх коливань пружної колони штанг із в'язким демпфуванням (рівняння Гіббса) за допомогою об'єктно-орієнтованого підходу та компонентного моделювання ядра бібліотеки `rusodyn`; запропоновано модель комплексної стохастичної деградації сигналу, яка діє на фіксований вектор часового ряду та імітує п'ять незалежних типів завад: фазовий зсув (розсинхронізація датчиків), гострі імпульсні викиди (збої апаратного тракту вимірювання та втрата пакетів даних), трендовий нахил похибки вимірювання, статичний дрейф нуля (температурне зміщення тензомоста) та мультиплікативний білий шум; розроблено та реалізовано алгоритм конструювання ознак; сформовано та сконфігуровано дослідницький ансамбль моделей класифікації та визначено методологічну базу валідації та оцінки якості моделей.

					КРБ.СІ - 21.00.00.000 ПЗ	
Змн.	Арк.	№ докум.	Підпис	Дата		37

3 РЕАЛІЗАЦІЯ ПРОГРАМНОГО МОДУЛЯ АВТОМАТИЗОВАНОЇ КЛАСИФІКАЦІЇ СТАНУ ШГНУ ТА ОЦІНКА ЙОГО ЕФЕКТИВНОСТІ

3.1 Оцінка надійності моделей на тестовій вибірці

Для проведення експерименту було використано розширений набір згенерованих даних (srp_data_engineered_10new_12.csv), який включає 100 точок нормалізованої динамограми та 12 сформованих метаознак (геометричних та статистичних) для кожного згенерованого циклу (описано в розділі 2). Оцінка ефективності алгоритмів проводилася на відкладеній тестовій вибірці (20% від загального масиву), а для підтвердження стійкості результатів додатково застосовувалася 5-кратна перехресна перевірка.

Зведені результати класифікації за метриками загальної точності (Accuracy) та зваженої F1-міри наведено в табл. 3.1.

Таблиця 3.1 — Порівняльна ефективність алгоритмів класифікації станів ШГНУ

Алгоритм машинного навчання	Accuracy (Точність)	F1-Score (Тест)	F1-Score (5-fold cross validation)
Багатошаровий перцептрон (MLP)	0.8675	0.8672	0.8660
Градiєнтний бустинг (XGBoost)	0.8610	0.8602	0.8610
Метод опорних векторів (SVM RBF)	0.8405	0.8393	0.8420
Random Forest	0.8285	0.8264	0.8340
Згорткова нейромережа (1D-CNN)	0.8035	0.8032	-
К-найближчих сусідів (KNN)	0.7875	0.7840	0.8040
Логістична регресія (Logistic Reg.)	0.7400	0.7373	0.7380

Аналіз наведених в табл. 3.1 метрик дозволяє зробити наступні висновки щодо застосовності різних класів алгоритмів для діагностики ШГНУ в умовах інтенсивних стохастичних завад:

1) неспроможність лінійних методів: логістична регресія показала найнижчий результат (Accuracy = 74.00%). Це підтверджує гіпотезу про те, що лінійні розділяючі гіперплощини не здатні впоратися зі складними просторовими викривленнями. Фазовий зсув, імітований у наборі даних, "змішує" ознаки між колонками, а температурний дрейф нуля зміщує абсолютні значення, роблячи лінійні алгоритми непридатними для промислових IoT-систем;

2) ефективність глибокого навчання: найкращу ефективність продемонструвала нейронна мережа MLP (86.75%), з мінімальним відривом випередивши XGBoost (86.10%). Успіх MLP пояснюється її здатністю виявляти глибокі нелінійні взаємозв'язки між "сирими" точками динамограми та розрахованими фізичними метаознаками (такими як спектральна ентропія чи перекіс навантаження). Збіг результатів на тестовій вибірці ($F1 = 0.8672$) та під час крос-валідації ($CV\ F1 = 0.8660$) свідчить про високу стійкість моделі до перенавчання;

3) роль згорткових нейромереж (1D-CNN): розроблена архітектура 1D-CNN на базі PyTorch показала задовільний результат (80.35%), значно випередивши лінійні методи. Така поведінка пояснюється її здатністю автоматично виділяти просторові ознаки за допомогою згорткових ядер (Conv1d). Однак, деяке відставання від ансамблевих методів (XGBoost) є типовим для задач аналізу структурованих табличних даних обсягом до 10 000 екземплярів, де алгоритми на основі дерев рішень переважають завдяки швидшій збіжності.

Програмний код модуля порівняння ефективності алгоритмів класифікації станів ШГНУ наведено в додатку В, а результат виводу показано нижче:

SRP_Diagnosis_ML\compare_models_advanced.py

					КРБ.СІ - 21.00.00.000 ПЗ	39
Змн.	Арк.	№ докум.	Підпис	Дата		

Завантаження даних з srp_data_engineered_10new_12.csv...
Генерація t-SNE...

Model Name	Accuracy	F1-Score
Logistic Regression	0.7400	0.7373 (CV F1: 0.738)
SVM (RBF)	0.8405	0.8393 (CV F1: 0.842)
KNN	0.7875	0.7840 (CV F1: 0.804)
Random Forest	0.8285	0.8264 (CV F1: 0.834)
MLP (Neural Net)	0.8675	0.8672 (CV F1: 0.866)
XGBoost	0.8610	0.8602 (CV F1: 0.861)
1D-CNN (PyTorch)	0.8035	0.8032

Process finished with exit code 0

3.2 Аналіз ефективності моделей з використанням алгоритму t-SNE та аналізу матриць помилок

Для розуміння складності задачі класифікації в умовах інтенсивних стохастичних завад було проведено зниження розмірності простору ознак за допомогою алгоритму t-SNE (t-Distributed Stochastic Neighbor Embedding). Результат проєктування 112-вимірному простору (100 сирих точок + 12 метазнак) у двовимірну площину наведено на рис. 3.1.

З рис. 3.1 видно, що клас FluidPound (фіолетові маркери) формує відносно відокремлений кластер у правій частині графіку. Це фізично обґрунтовано: різке падіння навантаження при ударі об рідину генерує настільки ознаку (висока асиметрія та зсув), що навіть сильний шум не здатний її замаскувати. Окрім того очевидна лінійна нероздільність інших класів: класи Normal (норма), GasLock (вплив газу) та ValveLeak (витік клапана) утворюють щільне скупчення ("хмару") з високим ступенем взаємного перекриття в лівій частині. Додані в датасет аномалії (фазовий зсув, лінійний тренд та амплітудний дрейф) штучно розмивають межі між цими станами, роблячи їх лінійно нероздільними.

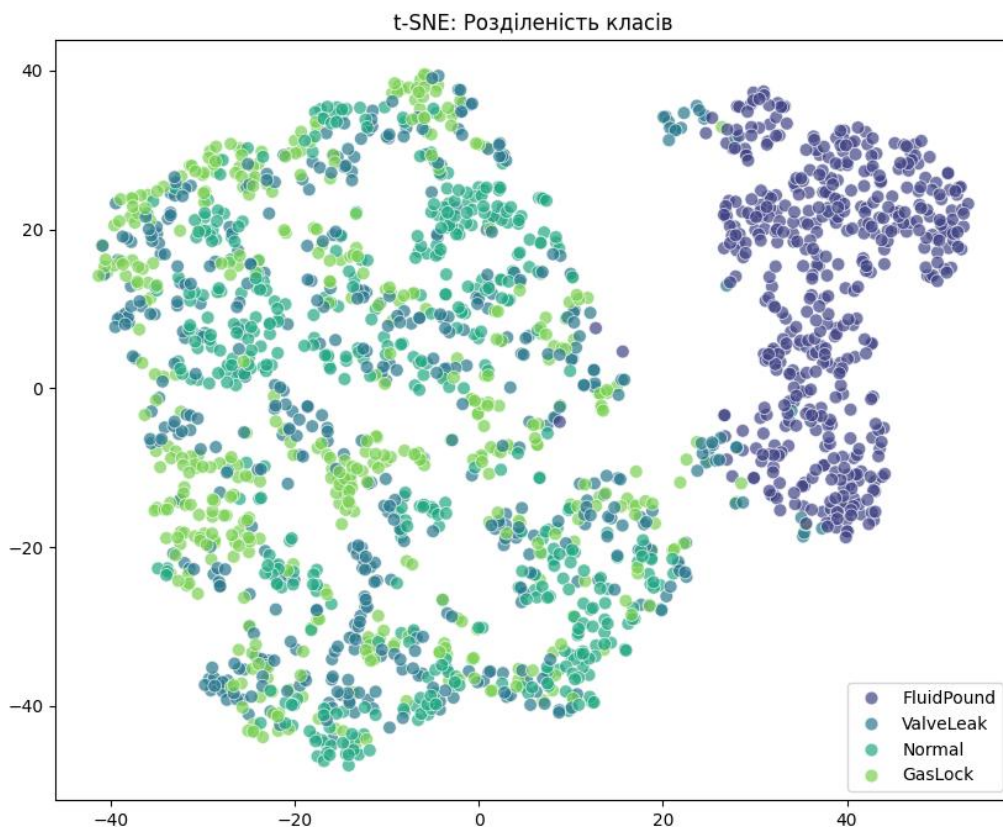


Рисунок 3.1 — Візуалізація розділеності класів станів ШГНУ методом t-SNE у двовимірному просторі

Для розуміння як різні архітектури машинного навчання справляються з "хмарою" перекриття (рис. 3.1) було побудовано матриці помилок для кожної моделі на тестовій вибірці (рис. 3.2-3.3).

Розподіл значень в матрицях, наведених на рис. 3.2-3.3, демонструє наступне:

— стійкість до FluidPound: усі алгоритми успішно ідентифікують удар плунжера (точність >99%, або 512-516 правильних відповідей з 517). Це підтверджує висновки, зроблені на основі t-SNE (рис.3.1);

— найпростіша модель логістичної регресії часто плутає стани. Наприклад, з 526 реальних випадків GasLock вона правильно вгадала лише 249, помилково віднісши 186 випадків до витоку клапана (ValveLeak), а 48 — до

норми. Це доводить, що за наявності фазового зсуву та зміни базової лінії лінійні рівняння втрачають здатність розпізнавати форму динамограми;

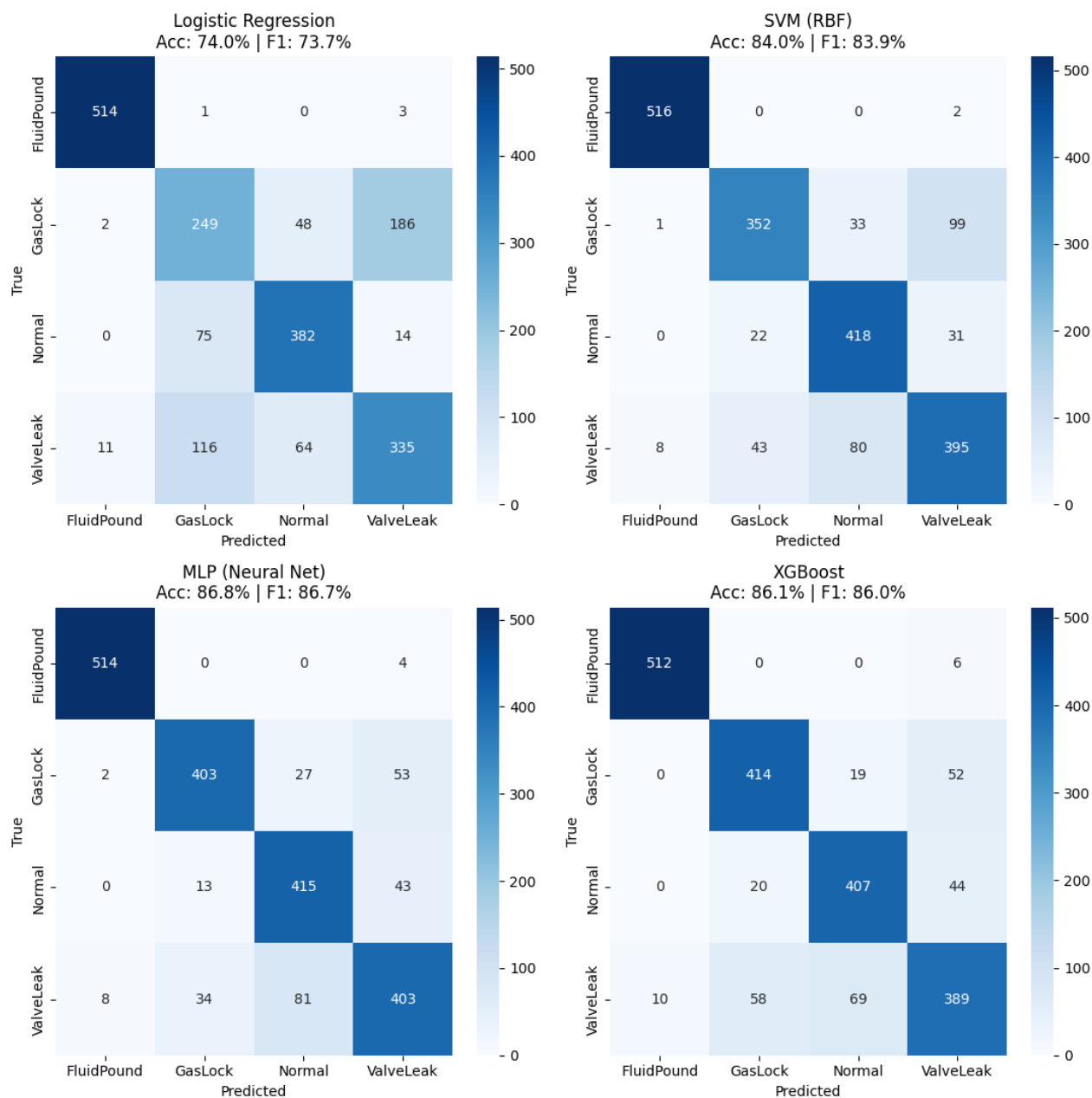


Рисунок 3.2 — Матриці помилок алгоритмів класифікації Logistic Reg., SVM, MLP, XGBoost на тестовій вибірці

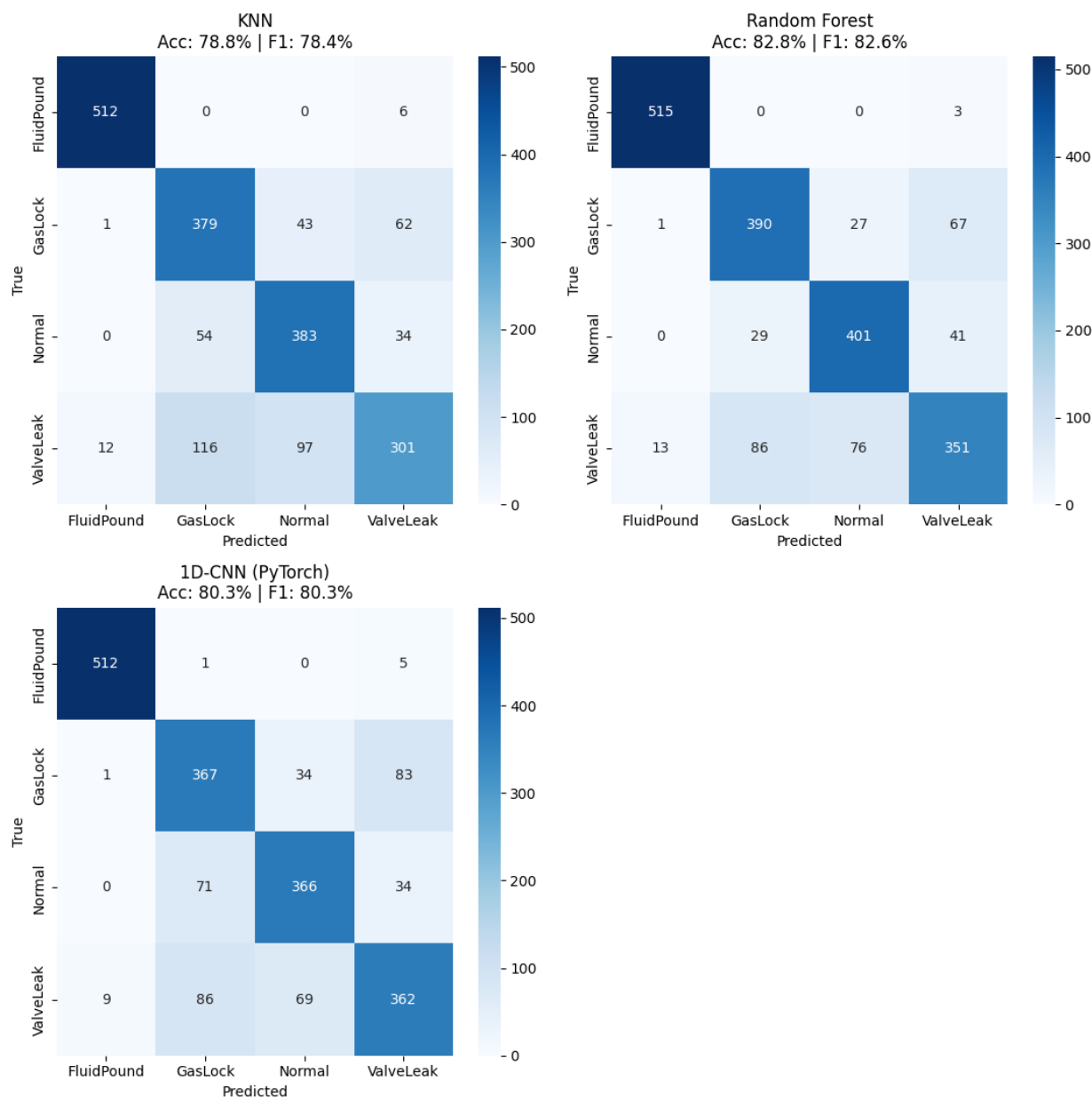


Рисунок 3.3 — Матриці помилок алгоритмів класифікації KNN, Random Forest, 1D-CNN на тестовій вибірці

— перевага багатоварового перцептрона (MLP) та XGBoost: вказані моделі ефективно розплутують нелінійні зв'язки. Нейромережа MLP підняла точність розпізнавання найскладнішого класу GasLock з 249 (у регресії) до 403 правильних прогнозів;

— межа чутливості: навіть у найкращих моделей (MLP, XGBoost) зберігається певний відсоток взаємного переплутування між Normal та ValveLeak (близько 30-40 хибних спрацювань). Фізично це пояснюється

наявністю у вибірці "мікро-витоків" (менше 10-15%), графічний прояв яких математично губиться в рівні накладеного гауссового шуму ($\sigma = 3,0$ кН).

Зведена гістограма продуктивності розробленого ансамблю моделей за ключовою метрикою зваженої F1-міри представлена на рис. 3.4.

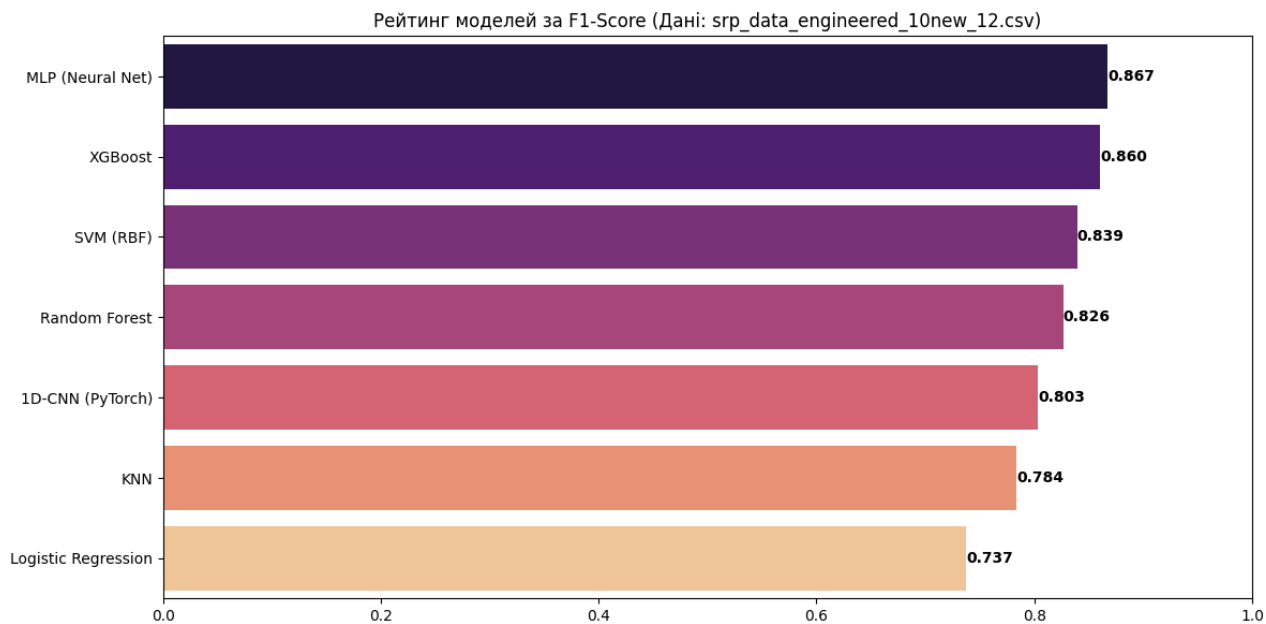


Рисунок 3.4 — Рейтинг моделей машинного навчання за зваженою метрикою F1-Score

Гістограма наочно демонструє зміну ефективності алгоритмів від простих до складних. Використання виключно розробленого комплексу інженерних ознак та сирих даних дозволило архітектурі глибокого навчання (MLP) досягти показника 0.867, що є значним результатом для настільки сильно спотвореного набору даних. Градієнтний бустинг (XGBoost) показав статистично ідентичний результат (0.860), що робить його оптимальним вибором для впровадження у промислові мікроконтролери (Edge Computing) завдяки меншим вимогам до обчислювальних ресурсів порівняно з нейромережами.

3.3 Дослідження інформативності складових вектора ознак

Для забезпечення прозорості прийняття рішень моделями машинного навчання та підтвердження ефективності розробленого алгоритму формування вектору ознак було проведено аналіз важливості кожної змінної (рис. 3.5-3.6).

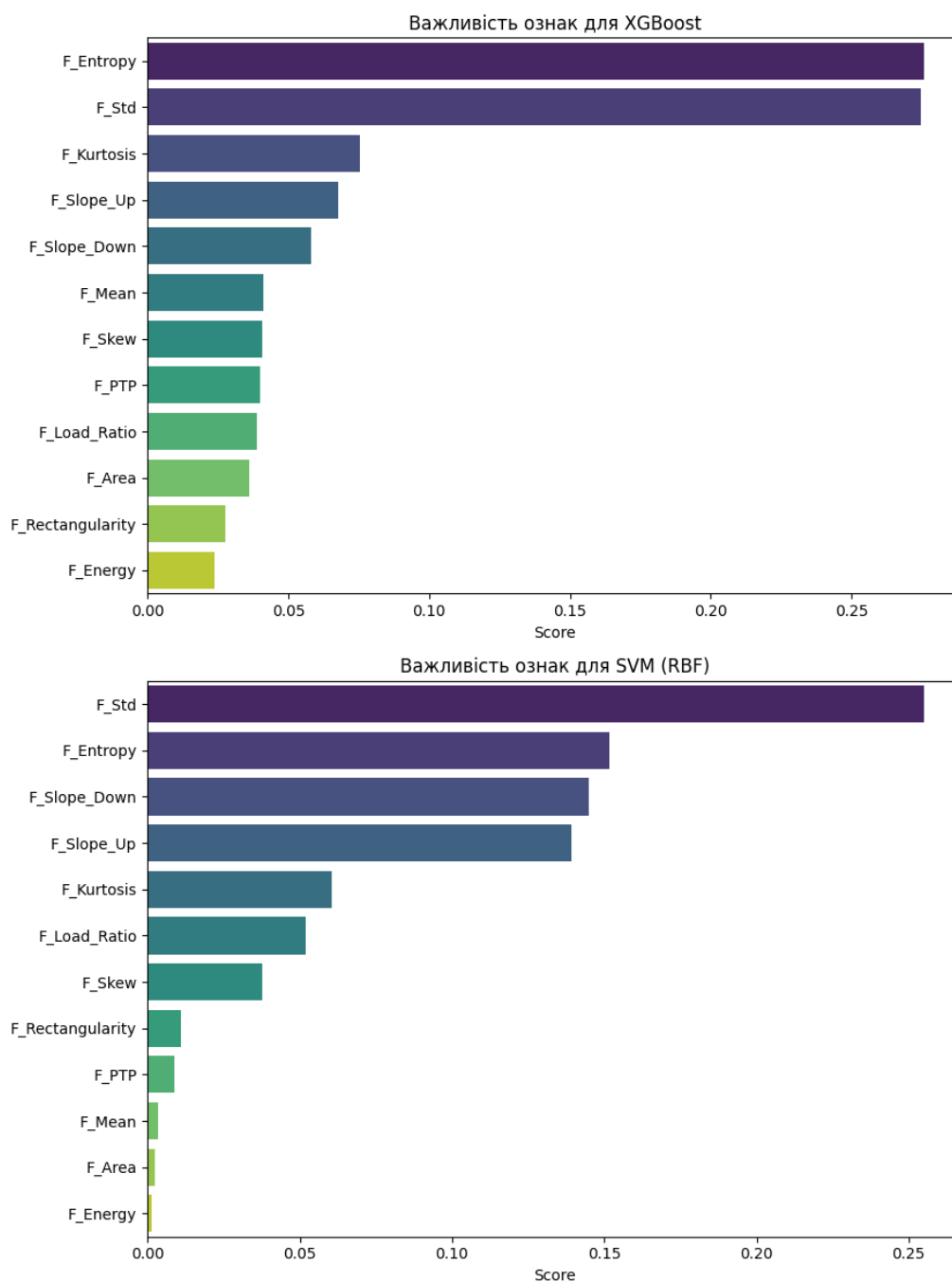


Рисунок 3.5 — Рейтинг важливості складових вектора ознак для XGBoost та SVM

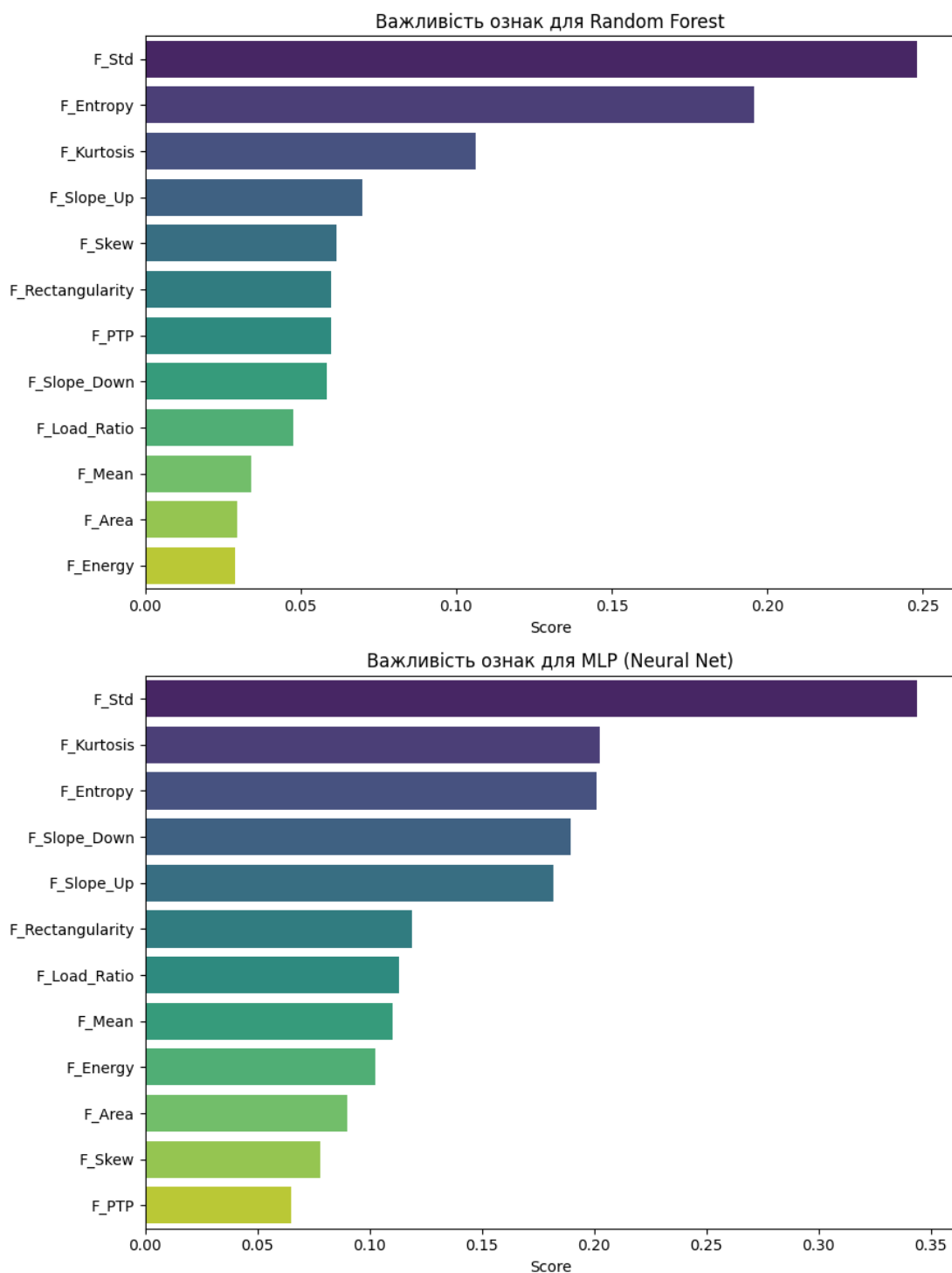


Рисунок 3.6 — Рейтинг важливості складових вектора ознак для Random Forest та MLP

Для ансамблевих моделей на базі дерев (XGBoost, Random Forest) використовувався вбудований критерій приросту інформації. Для нелінійних моделей (SVM, MLP) застосовано метод перестановочної важливості, який

оцінює падіння точності моделі при випадковому перемішуванні значень конкретної ознаки. Код модуля аналізу важливості складових вектору ознак наведено в додатку Г, а результат його роботи показано нижче:

```
SRP_Diagnosis_ML\analyze_features.py
Завантаження даних з srp_data_engineered_10new_12.csv...
```

Model	Top Feature	Importance

Аналіз моделі: XGBoost...		
XGBoost	F_Entropy	0.2757
Аналіз моделі: Random Forest...		
Random Forest	F_Std	0.2483
Аналіз моделі: SVM (RBF)...		
SVM (RBF)	F_Std	0.2550
Аналіз моделі: MLP (Neural Net)...		
MLP (Neural Net)	F_Std	0.3439

Process finished with exit code 0

Аналіз інформативності ознак дозволяє сформулювати такі висновки:

— найвагомішими ознаками для всіх чотирьох архітектур стали стандартне відхилення (F_Std) та спектральна ентропія (F_Entropy). Стандартне відхилення вловлює загальну амплітуду коливань навантаження, яка суттєво змінюється при різних режимах заповнення насоса. Спектральна ентропія математично описує рівень "хаосу" в сигналі. Вплив газу (Gas Lock) перетворює чіткий паралелограм динамограми на заокруглену фігуру з високим рівнем нелінійності, що фіксується ентропією, дозволяючи моделям (особливо XGBoost) відокремлювати цей клас від завад;

— детектування ударних навантажень: ексцес (Kurtosis), який визначає "гостроту" піків та наявність хвостів у розподілі, входить до перших трьох ознак (зокрема для MLP та Random Forest). Фізично ця ознака реагує на стан Fluid Round (удар плунжера), оскільки різке падіння зусилля під час ходу вниз генерує аномалію (сходінку) в часовому ряді, яку ексцес виділяє навіть на тлі білого шуму;

— додані ознаки кутів нахилу ліній регресії на фазах всмоктування та нагнітання (F_Slope_Up, F_Slope_Down) є важливими для роботи нелінійних моделей (для SVM та MLP вони займають 3-тє та 4-тє місця за важливістю). Саме

ці ознаки дозволили алгоритмам виявляти негерметичність клапанів (Valve Leak), оскільки витік фізично проявляється як зміна кута нахилу відповідних гілок динамограми. Лінійна регресія, проведена лише на цільових відрізках (точки 10-40 та 60-90), виявилася стійкою до фазового зсуву та дрейфу нуля.

Отже, здатність моделей (зокрема XGBoost та MLP) досягати точності понад 86% в умовах критичного зашумлення пояснюється вдалою комбінацією глобальних енергетичних (F_Entropy) та локальних геометричних (F_Slope) метаознак. Ці математичні абстракції діють як цифровий фільтр, що відсікає стохастичні завади і залишає алгоритмам машинного навчання чисту фізичну суть процесів у свердловині.

На основі отриманих експериментальних результатів та аналізу ефективності алгоритмів машинного навчання, можна сформулювати рекомендації щодо практичного використання розробленого модуля класифікації стану ШГНУ за виглядом динамограм:

1) зважаючи на те, що локальні контролери телемеханіки, встановлені безпосередньо біля гирла свердловини, мають обмежені обчислювальні ресурси, розгортання важких нейромережових архітектур (MLP, 1D-CNN) на них є недоцільним. Рекомендується використовувати модель градієнтного бустингу XGBoost. Вона продемонструвала точність (86.10%), статистично співставну з нейромережами (86.75%), проте у скомпільованому вигляді займає значно менше пам'яті та забезпечує мілісекундну швидкодію прогнозу, що є критичним для систем реального часу;

2) умовою успішного впровадження є відтворення розробленого алгоритму попереднього опрацювання даних на промисловому контролері. Програмний модуль реалізовувати таку послідовність:

- збір "сирого" масиву точок від датчиків зусилля та положення за один повний цикл роботи верстата-качалки;
- інтерполяція (стиснення або розтягнення) циклу до фіксованого розміру у 100 точок ;

- математичне обчислення 12 розроблених інженерних метаознак (спектральна ентропія, стандарте відхилення, ексцес, кути нахилу регресії). Подача у предиктивну модель лише сирих точок без цих ознак призведе до різкого падіння точності через наявність шумів;

3) для забезпечення масштабованості системи рекомендується інтеграція з хмарними платформами (наприклад, AWS SageMaker):

- периферійний контролер повинен передавати в SCADA-систему та хмару не весь масив точок, а лише розрахований діагноз та стиснений вектор метаознак, що зменшить навантаження на радіоканали або GSM-мережі підприємства;

- оскільки первинна модель навчена на синтетичних (згенерованих) даних, після встановлення системи на реальну свердловину необхідно застосувати метод трансферного навчання (Transfer Learning). Реальні динамограми, розмічені оператором-технологом як еталонні для конкретної свердловини, повинні періодично використовуватись для донавчання моделі у хмарі з подальшим оновленням ваг у локальному контролері;

4) результати класифікації не повинні одразу призводити до аварійної зупинки насоса (за винятком критичних станів, таких як обрив штанг або сильний удар плунжера – Fluid Pound). Для таких станів як вплив газу (Gas Lock) або витік клапана (Valve Leak), система повинна працювати в режимі системи підтримки в прийнятті рішень, накопичуючи статистику діагнозів за кілька десятків циклів поспіль. Якщо аномальний стан підтверджується стабільно - система генерує тривожне сповіщення для диспетчера та планує виїзд ремонтної бригади, реалізуючи концепцію предиктивного обслуговування.

Висновки до розділу 3

У третьому розділі проведено експериментальне дослідження розробленого програмного модуля автоматизованої класифікації станів ШГНУ на базі згенерованого та зашумленого набору даних. За результатами навчання,

					КРБ.СІ - 21.00.00.000 ПЗ	49
Змн.	Арк.	№ докум.	Підпис	Дата		

валідації та аналізу моделей машинного навчання доведено ефективність нелінійних та ансамблевих алгоритмів; обґрунтовано непридатність класичних лінійних методів для промислових IoT-систем; візуалізовано топологічну складність простору ознак; експериментально підтверджено високу інформативність розроблених інженерних метаознак, сформовано рекомендації щодо промислового впровадження. Розроблений конвеєр обробки даних (12 інженерних ознак та класифікатор XGBoost або MLP) показав високу стійкість до завад. Зважаючи на досягнуту точність та помірні вимоги до обчислювальних ресурсів, алгоритм XGBoost рекомендовано як базове математичне ядро для інтеграції в периферійні контролери нафтовидобувних свердловин з метою забезпечення предиктивного обслуговування ШГНУ.

					КРБ.СІ - 21.00.00.000 ПЗ	
Змн.	Арк.	№ докум.	Підпис	Дата		50

ВИСНОВКИ

У кваліфікаційній роботі розв'язано актуальну науково-прикладну задачу — розроблення алгоритмічного та програмного забезпечення для автоматизованої діагностики технічного стану штангових глибинно-насосних установок шляхом інтелектуального аналізу динамограм методами машинного навчання. За результатами проведених теоретичних та експериментальних досліджень зроблено наступні висновки.

Встановлено, що традиційні методи візуальної та ручної інтерпретації динамограм є суб'єктивними і не відповідають сучасним вимогам цифровізації нафтогазової галузі. Обґрунтовано необхідність переходу до автоматизованих систем на базі машинного навчання (ML), здатних працювати з "зашумленими" даними від периферійних IoT-датчиків у режимі реального часу.

Запропоновано шляхи розв'язання проблеми відсутності збалансованих промислових баз даних аварійних режимів. На основі чисельного розв'язку одновимірного хвильового рівняння Гіббса (з використанням бібліотеки `rusodun`) розроблено алгоритм синтезу еталонних динамограм для нормального режиму та трьох критичних несправностей (удар плунжера, вплив газу, витік клапана).

Для наближення синтетичних даних до реальних умов експлуатації запропоновано комплексну математичну модель деградації сигналу. Доведено, що накладання на еталонні динамограми фазового зсуву, температурного дрейфу нуля, імпульсних викидів та мультиплікативного шуму перетворює класи станів на лінійно нероздільні в багатовимірному просторі ознак (що підтверджено аналізом t-SNE).

Розроблено та програмно реалізовано алгоритм формування 12 ознак із часових рядів. Використання статистично-енергетичних параметрів (спектральна ентропія, ексцес) та локальних геометричних паттернів (кути нахилу фаз всмоктування/нагнітання) дозволило створити цифровий фільтр, що

відокремлює фізичну суть роботи насоса від стохастичних спотворень телеметрії.

Проведено експериментальне порівняння семи алгоритмів машинного навчання (Logistic Regression, SVM, KNN, Random Forest, MLP, XGBoost, 1D-CNN). Експериментально доведено, що класичні лінійні методи не здатні впоратися із значно зашумленими даними (точність 74.00%). Натомість архітектури глибокого навчання (MLP) та градієнтного бустингу (XGBoost) забезпечили високу точність діагностики — 86.75% та 86.10% відповідно, продемонструвавши високу стійкість до перенавчання (за результатами 5-кратної крос-валідації).

За допомогою аналізу важливості ознак підтверджено фізичну адекватність побудованих моделей. Виявлено, що спектральна ентропія є ключовим параметром для ідентифікації газових блокувань, ексцес ефективно фіксує удари плунжера, а регресійний нахил дозволяє виявляти втрату герметичності клапанного вузла.

Алгоритм XGBoost рекомендовано як обчислювальне ядро, тоді як хмарне середовище (наприклад, AWS SageMaker) пропонується використовувати для накопичення еталонних даних та періодичного донавчання алгоритмів методом трансферного навчання (Transfer Learning). Розроблений модуль може бути інтегрованим в системи предиктивного обслуговування нафтодобувних підприємств.

					КРБ.СІ - 21.00.00.000 ПЗ	
Змн.	Арк.	№ докум.	Підпис	Дата		52

ПЕРЕЛІК ПОСИЛАНЬ НА ДЖЕРЕЛА

1. Зікратий В.С., Штаєр Л.О. Сучасні Проблеми Керування Штанговими Глибинно-Насосними Установками. *II International Scientific and Practical Conference «The latest technologies implemented and teaching in universities»*: Proceedings of conference. Sofia, Bulgaria, 2025. P. 32–35. URL: <https://isg-konf.com/the-latest-technologies-implemented-and-teaching-in-universities/>.
2. Білецький В.С. Мала Гірнична Енциклопедія. Д.: Східний видавничий дім, 2013. Vol. 3. 644 p.
3. Зікратий В.С., Штаєр Л.О. Огляд Динамометричних Методів Контролю Технічного Стану Штангових Глибинно-Насосних Установок. *METHODS AND DEVICES OF QUALITY CONTROL*. 2025. 2(55). P. 13–21. DOI: 10.31471/1993-9981-2025-2(55)-13-21.
4. Зікратий В.С., Штаєр Л.О. Динамокарта Як Джерело Інформації Для Оцінки Технічного Стану Штангових Глибинно-Насосних Установок. *Всеукраїнська науково-практична конференція молодих учених і студентів «Інформаційні технології в освіті, техніці та промисловості»*: збірник доповідей. Івано-Франківськ: ІФНТУНГ, 2025. P. 65–69. URL: <https://drive.google.com/file/d/1sim7SKa62RzaS5XZrTGt248h5VAFf1WQ/view>.
5. Gibbs S.G. Diagnostic Analysis of Dynamometer Cards by Means of the Wave Equation. *Journal of Petroleum Technology*. 1967. Vol. 19, No. 02. P. 60–69.
6. Takács G. Sucker-Rod Pumping Handbook: Production Engineering Fundamentals and Long-Stroke Rod Pumping. Elsevier Science, 2015.
7. Заміховський Л.М., Ровінський В.А., Євчук О.В. Діагностика Технічного Стану Штангових Глибинно-Насосних Установок. Івано-Франківськ: Симфонія форте, 2006. 308 p.

					КРБ.СІ - 21.00.00.000 ПЗ	
Змн.	Арк.	№ докум.	Підпис	Дата		53

8. Заміховський Л.М., Ровінський В.А., Васьків О.В. Вдосконалення Тензометричного Давача Автоматизованої Системи Діагностування Штангових Глибинно-Насосних Установок. *Методи та прилади контролю якості*. 2002. No. 8. P. 19–21.

9. Fakher S., Khlaifat A., Hossain M.E., Nameer H. A comprehensive review of sucker rod pumps' components, diagnostics, mathematical models, and common failures and mitigations. *Journal of Petroleum Exploration and Production Technology*. 2021. Vol. 11, No. 10. P. 3815–3839. DOI: 10.1007/s13202-021-01270-7.

10. Peng Y. Artificial Intelligence Applied in Sucker Rod Pumping Wells: Intelligent Dynamometer Card Generation, Diagnosis, and Failure Detection Using Deep Neural Networks. *SPE Annual Technical Conference and Exhibition: SPE Annual Technical Conference and Exhibition*. Calgary, Alberta, Canada: SPE, 2019. P. D021S020R007. DOI: 10.2118/196159-MS.

11. Копей В.Б. Імітаційна Модель Свердловинної Штангової Насосної Установки На Основі Абстрактних Автоматів. *Розвідка та розробка нафтових і газових родовищ*. 2017. 3(64). P. 40–49. URL: <http://elar.nung.edu.ua/handle/123456789/6668>.

12. WelWellPilot® Controllers for Rod Pump Optimization. 2017. URL: <https://www.weatherford.com/documents/brochure/products-and-services/production-optimization/wellpilot-controllers-for-rod-pump-optimization/>.

13. XSPOC Production Optimization Software. *ChampionX*. 2023. URL: <https://www.championx.com>.

14. Westermo Effective Oilfield Automation Using Mobile Routers. URL: <https://www.westermo.com/us/industries/oil-and-gas/success-stories/lufkin>.

15. Yin C., Zhang K., Zhang L., Wang Z., Liu P., Zhang H., Yang Y., Yao J. Imbalanced Working States Recognition of Sucker Rod Well Dynamometer Cards

Based on Data Generation and Diversity Augmentation. *SPE Journal*. 2023. Vol. 28, No. 04. P. 1925–1944. DOI: 10.2118/214661-PA.

16. Копей В.Б. Програмна Бібліотека Для Розрахунку Та Проектування Насосного Обладнання (PU). URL: <https://github.com/vkopey/PU>.

					КРБ.СІ - 21.00.00.000 ПЗ	
Змн.	Арк.	№ докум.	Підпис	Дата		55

Додаток А Програмний код синтезу динамограм з шумом

```
import pandas as pd
import numpy as np
import math
import random
from scipy.interpolate import interp1d

try:
    from rycodyn import SpringDamper, Mass, Force, System, dt
except ImportError:
    print("ПОМИЛКА: Не знайдено файл rycodyn.py або в ньому помилки синтаксису.")
    exit()

# --- КОНСТАНТИ МОДЕЛІ (з файлу main2s.py) ---
FS_BASE = -34687.0 # Вага колони (Section weight)
FR_BASE = -18499.0 # Вага рідини (Liquid weight)
# Налаштування компонентів (Spring, Mass)
C_ROD = 44650.0
D_ROD = 2120.7
M_ROD = 3961.0

# --- ЛОГІКА НЕСПРАВНОСТЕЙ (Fault Injection) ---
class PumpSimulator:

    def __init__(self, fault_type='Normal'):
        self.fault_type = fault_type

        # Рандомізація фізичних параметрів
        self.stroke_amp = (2.1 / 2) * random.uniform(0.95, 1.05)
        self.freq_n = (6.4 / 60) * random.uniform(0.9, 1.1)
        self.pound_severity = random.uniform(0.85, 0.99)
        self.leak_severity = random.uniform(0.10, 0.40)

        # Ініціалізація компонентів rycodyn
        self.s1 = SpringDamper(name='s1', c=C_ROD, d=D_ROD)
        self.m1 = Mass(name='m1', m=M_ROD)
        self.f1 = Force(name='f1')

        # Збірка системи (Topology)
        peqs = self.s1.pinEqs(1, [self.m1.pins[0]])
        peqs += self.m1.pinEqs(1, [self.f1.pins[0]])
        self.sys = System(els=[self.s1, self.m1, self.f1], eqs=peqs)

        self.ics = {
            self.m1.x: 0.0, self.m1.v: 0.0, self.m1.a: 0.0,
            self.m1.xp: 0.0, self.m1.vp: 0.0,
            self.m1.f1: 0.0, self.m1.f2: 0.0,
            self.s1.x1: 0.0, self.s1.x1p: 0.0, self.s1.x2: 0.0, self.s1.x2p: 0.0,
            self.s1.vrel: 0.0, self.s1.f1: 0.0, self.s1.f2: 0.0,
            self.f1.f: FS_BASE + FR_BASE, self.f1.x: 0.0, self.f1.xp: 0.0
        }
        self.sys.solve(self.ics)

    def motion(self, t):
        """Кінематика точки підвісу (верхня точка)"""
        return self.stroke_amp * math.sin(2 * math.pi * self.freq_n * t)

    def force_logic(self, v, t_cycle_normalized):
        # Базова сила (вага штанг)
```

```

F = FS_BASE

if self.fault_type == 'Normal':
    if v > 0: F += FR_BASE

elif self.fault_type == 'FluidPound':
    if v > 0:
        F += FR_BASE
    else:
        if abs(math.sin(t_cycle_normalized * 2 * math.pi)) > self.pound_severity:
            pass
        else:
            F += FR_BASE * 0.8

elif self.fault_type == 'GasLock':
    if v > 0:
        factor = math.tanh(v / 0.5)
        F += FR_BASE * factor
    else:
        factor = math.tanh(abs(v) / 0.5)

elif self.fault_type == 'ValveLeak':
    if v > 0:
        F += FR_BASE * (1.0 - self.leak_severity)
    else:
        F += FR_BASE * (0.1 * self.leak_severity)

return F * math.tanh(abs(v) / 0.01)

def run(self):
    # Функція граничних умов для Solver
    def fnBC(d, t):
        # Визначаємо, де ми в циклі (0..1)
        period = 1.0 / self.freq_n
        t_cycle = (t % period) / period

        val = self.motion(t, self.force_logic(d[self.m1.v], t_cycle)
        return dict(zip([self.sys.elsd['s1'].x1, self.sys.elsd['f1'].f], val))

    # Налаштування змінних для fnBC (pseudocode specific magic)
    fnBC.vrs = self.sys.elsd['s1'].x1, self.sys.elsd['f1'].f

    # Запуск динамічного розрахунку (2 повних цикли)
    period = 1.0 / self.freq_n
    sim_time = 2.5 * period

    T_raw, R_raw = self.sys.solveDyn(self.ics, timeEnd=sim_time, fnBC=fnBC)

    start_cut = 1.5 * period

    s1_x1_sym = self.sys.elsd['s1'].x1
    s1_f1_sym = self.sys.elsd['s1'].f1

    pos_data = []
    load_data = []

    for t, res in zip(T_raw, R_raw):
        if t > start_cut:
            pos_data.append(float(res[s1_x1_sym]))
            # Сила в пускодун може бути від'ємною або в Ньютонах.

```

```

        # Ділимо на 1000 для кН (як в main2s.py)
        load_data.append(float(res[s1_f1_sym]) / 1000.0)

    return pos_data, load_data

def resample_data(x, y, target_length=100):
    """
    Інтерполяція даних до фіксованої довжини (100 точок),
    щоб це можна було зберегти в таблицю CSV.
    """
    if len(x) < 2: return np.zeros(target_length), np.zeros(target_length)

    indices = np.linspace(0, 1, len(x))
    target_indices = np.linspace(0, 1, target_length)

    f_x = interp1d(indices, x, kind='linear')
    f_y = interp1d(indices, y, kind='linear')

    return f_x(target_indices), f_y(target_indices)

if __name__ == "__main__":
    NUM_SAMPLES = 10000
    dataset = []
    labels = []
    classes = ['Normal', 'FluidPound', 'GasLock', 'ValveLeak']

    print(f"Починаємо генерацію {NUM_SAMPLES} прикладів (Final Scientific Mode)...")

    for i in range(NUM_SAMPLES):
        fault = random.choice(classes)
        try:
            sim = PumpSimulator(fault_type=fault)
            pos_raw, load_raw = sim.run()

            # 2. Нормалізація
            _, load_100 = resample_data(pos_raw, load_raw, target_length=100)

            # Перетворюємо в нумпу для зручності
            cycle = np.array(load_100)

            # --- 3. НАКЛАДАННЯ ЗАВАД НА ГОТОВИЙ ВЕКТОР ---

            shift = random.randint(-20, 20)
            cycle = np.roll(cycle, shift)

            if random.random() < 0.5:
                num_spikes = random.randint(1, 3)
                for _ in range(num_spikes):
                    idx = random.randint(0, 99)
                    cycle[idx] += random.choice([-1, 1]) * random.uniform(15, 30)

            trend = np.linspace(random.uniform(-5, 5), random.uniform(-5, 5), 100)
            cycle += trend

            cycle += random.uniform(-10, 10)

            cycle *= random.uniform(0.9, 1.1) # Масштаб +/- 10%
            cycle += np.random.normal(0, 2.0, 100) # Фоновий шум

            # -----

```

```

dataset.append(cycle)
labels.append(fault)

if i % 200 == 0: print(f"Згенеровано {i}/{NUM_SAMPLES}...")

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

# Збереження
cols = [f'L{i}' for i in range(100)]
df = pd.DataFrame(dataset, columns=cols)
df['Label'] = labels

csv_name = 'srp_simulated_data_complex_noise_10.csv'
df.to_csv(csv_name, index=False)
print(f"УСПІХ! Дані збережено в {csv_name}")

```

Додаток Б Програмний код модуля інженерії ознак

```
import pandas as pd
import numpy as np
from scipy.stats import skew, kurtosis, entropy, linregress
from scipy.signal import periodogram

def add_all_features():
    # 1. НАПАШТУВАННЯ
    # Вхідний файл (сирі дані 10к)
    input_file = 'srp_simulated_data_complex_noise_10.csv'
    output_file = 'srp_data_engineered_10new_12.csv'

    print(f"Завантаження даних з {input_file}...")
    try:
        df = pd.read_csv(input_file)
    except FileNotFoundError:
        print(f"Помилка: Файл {input_file} не знайдено!")
        return

    data_cols = [col for col in df.columns if col.startswith('L') and col != 'Label']
    X_raw = df[data_cols].values.astype(float)

    print(f"Розрахунок ПОВНОГО набору ознак для {len(df)} рядків...")

    # --- ІНІЦІАЛІЗАЦІЯ СПИСКІВ ---
    # Група 1: Базові
    f_area, f_mean, f_std, f_skew, f_kurt = [], [], [], [], []
    f_ptp, f_energy, f_entropy = [], [], []

    # Група 2: Геометричні
    f_rect, f_slope_up, f_slope_down, f_load_ratio = [], [], [], []

    for row in X_raw:
        # --- РОЗРАХУНОК ГРУПИ 1 (БАЗОВІ + ЕНТРОПІЯ) ---
        # Площа
        try:
            area = np.trapezoid(row)
        except AttributeError:
            area = np.trapz(row)

        # Статистика
        mean_val = np.mean(row)
        min_val = np.min(row)
        max_val = np.max(row)
        ptp = max_val - min_val

        f_area.append(area)
        f_mean.append(mean_val)
        f_std.append(np.std(row))
        f_skew.append(skew(row))
        f_kurt.append(kurtosis(row))
        f_ptp.append(ptp)
        f_energy.append(np.sum(row ** 2))
```

```

# Ентропія
freqs, psd = periodogram(row, fs=1.0)
psd_norm = psd / (np.sum(psd) + 1e-10)
f_entropy.append(entropy(psd_norm))

# --- РОЗРАХУНОК ГРУПИ 2 ---
# A. Rectangularity (Прямокутність)
rect_area_ref = ptp * 100
if rect_area_ref == 0: rect_area_ref = 1e-6
f_rect.append(area / rect_area_ref)

# B. Slopes (Нахили)
# Хід вгору (точки 10-40)
slope_u, _, _, _ = linregress(np.arange(10, 40), row[10:40])
f_slope_up.append(slope_u)

# Хід вниз (точки 60-90)
slope_d, _, _, _ = linregress(np.arange(60, 90), row[60:90])
f_slope_down.append(slope_d)

# C. Load Ratio (Переки́с навантаження)
mean_start = np.mean(row[:30])
mean_end = np.mean(row[70:])
if mean_end == 0: mean_end = 1e-6
f_load_ratio.append(mean_start / mean_end)

features_df = pd.DataFrame({
    # Старі (попередні)
    'F_Area': f_area,
    'F_Mean': f_mean,
    'F_Std': f_std,
    'F_Skew': f_skew,
    'F_Kurtosis': f_kurt,
    'F_PTP': f_ptp,
    'F_Energy': f_energy,
    'F_Entropy': f_entropy,
    # Нові
    'F_Rectangularity': f_rect,
    'F_Slope_Up': f_slope_up,
    'F_Slope_Down': f_slope_down,
    'F_Load_Ratio': f_load_ratio
})

df_final = pd.concat([features_df, df["Label"]], axis=1)

df_final.to_csv(output_file, index=False)
print("-" * 50)
print(f"ГОТОВО! Файл збережено: {output_file}")
print(f"Всього ознак (Feature Engineering): {features_df.shape[1]}")
print(f"Загальна кількість колонок: {df_final.shape[1]}")
print("-" * 50)

if __name__ == "__main__":
    add_all_features()

```

Додаток В Програмний код модуля порівняння ефективності алгоритмів класифікації станів ШГНУ

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
from sklearn.model_selection import train_test_split, cross_val_score
from sklearn.preprocessing import LabelEncoder, StandardScaler
from sklearn.metrics import accuracy_score, confusion_matrix, f1_score
from sklearn.manifold import TSNE
from sklearn.base import BaseEstimator, ClassifierMixin

# Імпорт моделей
from sklearn.linear_model import LogisticRegression
from sklearn.svm import SVC
from sklearn.ensemble import RandomForestClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.neural_network import MLPClassifier
from xgboost import XGBClassifier

# --- БЛОК PYTORCH (CNN) ---
import torch
import torch.nn as nn
import torch.optim as optim

# 1. Архітектура CNN
class SimpleCNN(nn.Module):
    def __init__(self, num_classes=4, input_len=100):
        super(SimpleCNN, self).__init__()
        self.layer1 = nn.Sequential(
            nn.Conv1d(1, 16, kernel_size=5, padding=2),
            nn.BatchNorm1d(16),
            nn.ReLU(),
            nn.MaxPool1d(kernel_size=2, stride=2)
        )
        self.layer2 = nn.Sequential(
            nn.Conv1d(16, 32, kernel_size=5, padding=2),
            nn.BatchNorm1d(32),
            nn.ReLU(),
            nn.MaxPool1d(2)
        )
        self.drop = nn.Dropout(0.3)

    with torch.no_grad():
        dummy = torch.zeros(1, 1, input_len)
```

```

        dummy = self.layer1(dummy)
        dummy = self.layer2(dummy)
        flat_size = dummy.view(1, -1).size(1)

    self.fc = nn.Linear(flat_size, num_classes)

def forward(self, x):
    out = self.layer1(x)
    out = self.layer2(out)
    out = out.reshape(out.size(0), -1)
    out = self.drop(out)
    out = self.fc(out)
    return out

# 2. Обработка PyTorch
class PyTorchClassifier(BaseEstimator, ClassifierMixin):
    def __init__(self, epochs=50, lr=0.001):
        self.epochs = epochs
        self.lr = lr
        self.model = None
        self.device = torch.device('cuda' if torch.cuda.is_available() else 'cpu')

    def fit(self, X, y):
        self.classes_ = np.unique(y)
        input_len = X.shape[1]

        X_tensor = torch.FloatTensor(X).unsqueeze(1).to(self.device)
        y_tensor = torch.LongTensor(y).to(self.device)

        self.model = SimpleCNN(num_classes=len(self.classes_), input_len=input_len).to(self.device)
        criterion = nn.CrossEntropyLoss()
        optimizer = optim.Adam(self.model.parameters(), lr=self.lr)

        self.model.train()
        for epoch in range(self.epochs):
            optimizer.zero_grad()
            outputs = self.model(X_tensor)
            loss = criterion(outputs, y_tensor)
            loss.backward()
            optimizer.step()
        return self

    def predict(self, X):
        self.model.eval()
        X_tensor = torch.FloatTensor(X).unsqueeze(1).to(self.device)
        with torch.no_grad():
            outputs = self.model(X_tensor)
            _, predicted = torch.max(outputs.data, 1)

```



```

return predicted.cpu().numpy()

# --- ГОЛОВНА ФУНКЦІЯ ---
def analyze_and_visualize():
    filename = 'srp_data_engineered_10new_12.csv'
    print(f"Завантаження даних з {filename}...")

    try:
        df = pd.read_csv(filename)
    except FileNotFoundError:
        print("Файл не знайдено! Спочатку запустіть add_all_features.py")
        return

    # Підготовка
    X = df.drop('Label', axis=1).values
    y_text = df['Label'].values

    le = LabelEncoder()
    y = le.fit_transform(y_text)
    class_names = le.classes_

    scaler = StandardScaler()
    X_scaled = scaler.fit_transform(X)

    X_train, X_test, y_train, y_test = train_test_split(X_scaled, y, test_size=0.2, random_state=42)

    # --- 1. t-SNE ---
    print("Генерація t-SNE...")
    limit = min(2000, len(X_scaled))
    indices = np.random.choice(len(X_scaled), limit, replace=False)

    tsne = TSNE(n_components=2, random_state=42, perplexity=30, init='pca', learning_rate='auto')
    X_embedded = tsne.fit_transform(X_scaled[indices])

    plt.figure(figsize=(10, 8))
    sns.scatterplot(x=X_embedded[:, 0], y=X_embedded[:, 1], hue=y_text[indices], palette='viridis', s=60,
alpha=0.7)
    plt.title('t-SNE: Розділеність класів')
    plt.show()

    # --- 2. НАВЧАННЯ ТА МАТРИЦІ ---
    models = {
        "Logistic Regression": LogisticRegression(max_iter=1000),
        "SVM (RBF)": SVC(),
        "KNN": KNeighborsClassifier(),
        "Random Forest": RandomForestClassifier(n_estimators=100),
        "MLP (Neural Net)": MLPClassifier(hidden_layer_sizes=(100, 50), max_iter=500),
        "XGBoost": XGBClassifier(use_label_encoder=False, eval_metric='mlogloss'),
    }

```

```

    "1D-CNN (PyTorch)": PyTorchClassifier(epochs=100, lr=0.001)
}

fig, axes = plt.subplots(2, 4, figsize=(22, 11))
axes = axes.flatten()

results_acc = {}
results_f1 = {}

print("-" * 65)
print(f"{'Model Name':<25} | {'Accuracy':<10} | {'F1-Score':<10}")
print("-" * 65)

for i, (name, model) in enumerate(models.items()):
    # 1. Тренування на спліті
    model.fit(X_train, y_train)
    preds = model.predict(X_test)

    # Метрики
    acc = accuracy_score(y_test, preds)
    f1 = f1_score(y_test, preds, average='weighted')

    results_acc[name] = acc
    results_f1[name] = f1

    # 2. Cross-Validation
    if name != "1D-CNN (PyTorch)":
        cv_scores = cross_val_score(model, X_scaled, y, cv=5, scoring='f1_weighted')
        cv_str = f"(CV F1: {cv_scores.mean():.3f})"
    else:
        cv_str = ""

    print(f"{'name':<25} | {'acc':.4f} | {'f1':.4f} {cv_str}")

    # Матриця
    cm = confusion_matrix(y_test, preds)
    sns.heatmap(cm, annot=True, fmt='d', cmap='Blues', ax=axes[i],
                xticklabels=class_names, yticklabels=class_names)

    axes[i].set_title(f"{name}\nAcc: {acc:.1%} | F1: {f1:.1%}")
    axes[i].set_xlabel('Predicted')
    axes[i].set_ylabel('True')

for j in range(i + 1, len(axes)):
    fig.delaxes(axes[j])

plt.tight_layout()
plt.show()

```

```

# --- 3. ФІНАЛЬНИЙ ГРАФІК (F1 SCORE) ---
plt.figure(figsize=(12, 6))
sorted_res = dict(sorted(results_f1.items(), key=lambda item: item[1], reverse=True))

sns.barplot(
    x=list(sorted_res.values()),
    y=list(sorted_res.keys()),
    hue=list(sorted_res.keys()),
    legend=False,
    palette='magma' # Змінив палітру для різноманіття
)
plt.xlim(0, 1.0)
plt.title(f'Рейтинг моделей за F1-Score (Дані: {filename})')

for index, value in enumerate(sorted_res.values()):
    plt.text(value, index, f'{value:.3f}', va='center', fontweight='bold')

plt.tight_layout()
plt.show()

if __name__ == "__main__":
    analyze_and_visualize()

```

Додаток Г Програмний код модуля аналізу важливості складових вектору ознак

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder, StandardScaler
from sklearn.inspection import permutation_importance

# Моделі
from sklearn.ensemble import RandomForestClassifier
from sklearn.neural_network import MLPClassifier
from sklearn.svm import SVC
from xgboost import XGBClassifier

def analyze_feature_importance():
    # 1. ЗАВАНТАЖЕННЯ
    filename = 'srp_data_engineered_10new_12.csv'
    print(f"Завантаження даних з {filename}...")

    try:
        df = pd.read_csv(filename)
    except FileNotFoundError:
        print("Файл не знайдено!")
        return

    # Підготовка
    X = df.drop('Label', axis=1)
    y = LabelEncoder().fit_transform(df['Label'])
    feature_names = X.columns.tolist()

    # Масштабування (важливо для SVM/MLP)
    X_scaled = StandardScaler().fit_transform(X)

    # Спліт
    X_train, X_test, y_train, y_test = train_test_split(X_scaled, y, test_size=0.2, random_state=42)

    # Список моделей для аналізу
    models = {
        "XGBoost": XGBClassifier(use_label_encoder=False, eval_metric='mlogloss', random_state=42),
        "Random Forest": RandomForestClassifier(n_estimators=100, random_state=42),
        "SVM (RBF)": SVC(random_state=42),
        "MLP (Neural Net)": MLPClassifier(hidden_layer_sizes=(100, 50), max_iter=500, random_state=42)
    }

    fig, axes = plt.subplots(2, 2, figsize=(18, 12))
    axes = axes.flatten()

    print("-" * 50)
    print(f'{"Model":<20} | {"Top Feature":<20} | {"Importance"}')
```

```

print("-" * 50)

for i, (name, model) in enumerate(models.items()):
    print(f"Аналіз моделі: {name}...")
    model.fit(X_train, y_train)

    # Розрахунок важливості
    if name in ["XGBoost", "Random Forest"]:
        # Для дерев використовуємо вбудовану важливість
        importances = model.feature_importances_
    else:
        # Для SVM та MLP використовуємо Permutation Importance
        perm_result = permutation_importance(model, X_test, y_test, n_repeats=10, random_state=42)
        importances = perm_result.importances_mean

    # Створення DataFrame для сортування
    feat_imp = pd.DataFrame({
        'Feature': feature_names,
        'Importance': importances
    }).sort_values(by='Importance', ascending=False)

    top_f = feat_imp.iloc[0]
    print(f"{name:<20} | {top_f['Feature']:<20} | {top_f['Importance']:.4f}")

    sns.barplot(
        x='Importance',
        y='Feature',
        data=feat_imp.head(15),
        ax=axes[i],
        palette='viridis',
        hue='Feature',
        legend=False
    )
    axes[i].set_title(f"Важливість ознак для {name}")
    axes[i].set_xlabel("Score")
    axes[i].set_ylabel("")

plt.tight_layout()
plt.show()

if __name__ == "__main__":
    analyze_feature_importance()

```

БІБЛІОГРАФІЧНА ДОВІДКА

Тема кваліфікаційної роботи бакалавра – «Розроблення програмного модуля автоматизованої класифікації стану штангових глибинно-насосних установок на основі інтелектуального аналізу динамограм»

Обсяг пояснювальної записки в аркушах – 69.

Перелік креслень графічної частини:

КРБ.СІ-21.00.00.000 С1 – Узагальнена структура системи керування ШГНУ. Схема структурна (1 аркуш)

КРБ.СІ-21.00.00.000 – Синтетичні динамограми. Робочі вікна (1 аркуш)

КРБ.СІ-21.00.00.000 – Матриці помилок алгоритмів класифікації на тестовій вибірці. Робочі вікна (1 аркуш)

КРБ.СІ-21.00.00.000 – Аналіз важливості складових вектора ознак. Робочі вікна (1 аркуш)

Дата закінчення бакалаврської роботи « 15 » червня 2026 р.

Студент _____ Кобльовський С.Ю.