

Міністерство освіти і науки України  
Львівський національний університет природокористування  
Факультет механіки, енергетики та інформаційних технологій  
Кафедра інформаційних технологій

# КВАЛІФІКАЦІЙНА РОБОТА

другого (магістерського) рівня вищої освіти

на тему:

«Використання методів машинного навчання для оцінювання  
транспортних засобів агрофірми»

Виконав: здобувач групи Іт-61  
спеціальності 126  
«Інформаційні системи та технології»

Стецишин О.М.

Керівник:

Запорожцев С.Ю.

Рецензент:

Семерак В.М.

ЛЬВІВ-2024

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ЛЬВІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ПРИРОДОКОРИСТУВАННЯ  
ФАКУЛЬТЕТ МЕХАНІКИ, ЕНЕРГЕТИКИ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ  
КАФЕДРА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Другий (магістерський) рівень вищої освіти  
Спеціальність 126 – „Інформаційні системи та технології”

“ЗАТВЕРДЖУЮ”

Завідувач кафедри \_\_\_\_\_  
д.т.н., проф. А.М. Тригуба  
“ \_\_\_\_\_ ” \_\_\_\_\_ 202\_ р.

## ***ЗАВДАННЯ***

на кваліфікаційну роботу здобувачу

\_\_\_\_\_ Стецишин Олег Михайлович \_\_\_\_\_

1. Тема роботи: Використання методів машинного навчання для оцінювання транспортних засобів агрофірми

Керівник роботи Запорожцев Сергій Юрійович, к.т.н., доцент.

Затверджені наказом по університету від 12 вересня 2024 року № 616/к-с

2. Строк подання здобувачем роботи 05.12.2024 р.

3. Вихідні дані до роботи: Загальні відомості про методи оцінювання транспортних засобів, варіанти підходів до машинного навчання та інструментарію для вирішення задачі

4. Зміст розрахунково-пояснювальної записки:

### Вступ

1. Аналіз стану питання та постановка завдання. Загальні підходи до оцінки транспортних засобів. Розрахунок вартості машин на основі майнового підходу. Розрахунок собівартості обладнання на основі дохідного підходу. Розрахунок вартості машин на основі порівняльного підходу. Постановка задачі дослідження.

2. Обґрунтування та вибір інструментарію вирішення задачі. Алгоритми машинного навчання для оцінювання транспорту. Оптимізація параметрів ML-моделей для оцінки транспорту. Вибір програмних засобів реалізації системи.

3. Результати вирішення задачі. Підключення до API сервісу оголошень. Завантаження та попередня обробка вхідних даних. Реалізація та оптимізація моделі машинного навчання. Розробка елементів інтерфейсу користувача.

4. Охорона праці та безпека у надзвичайних ситуаціях

5. Визначення ефективності інформаційної системи.

Висновки та пропозиції.

Список використаних джерел.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень): \_\_\_\_\_  
Мета роботи. Підходи до оцінки машин та обладнання. Методи машинного навчання.  
Інструментарій вирішення задачі. Отримання даних через API. Попередня обробка  
даних. Розвідувальний аналіз та підготовка даних. Реалізація та навчання моделі.  
Оптимізація і результати моделювання. Елементи графічного інтерфейсу. Висновки.

6. Консультанти з розділів:

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1, 2, 3, 5	<i>Запорожцев С.Ю., доцент кафедри інформаційних технологій</i>		
4	<i>Городецький І.М., доцент кафедри управління проектами та безпеки виробництва</i>		

7. Дата видачі завдання 12 вересня 2024 р.

### **КАЛЕНДАРНИЙ ПЛАН**

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	<i>Написання першого розділу</i>	12.09 - 25.09.24	
2	<i>Виконання другого розділу та аркушів ілюстраційного матеріалу до нього</i>	26.09 - 10.09.24	
3.	<i>Виконання третього розділу та аркушів ілюстраційного матеріалу до нього</i>	11.10 - 25.10.24	
4.	<i>Написання розділу «Охорона праці та безпека у надзвичайних ситуаціях»</i>	26.10 - 5.11.24	
5.	<i>Написання розділу «Визначення ефективності...»</i>	6.11 - 15.11.24	
6.	<i>Завершення оформлення розрахунково-пояснювальної записки та аркушів ілюстраційного матеріалу</i>	16.11 - 29.11.24	
7.	<i>Завершення роботи в цілому</i>	30.11 - 5.12.24	

Студент \_\_\_\_\_ Стецишин О.М.  
 (підпис)

Керівник роботи \_\_\_\_\_ Запорожцев С.Ю.  
 (підпис)

## РЕФЕРАТ

УДК 004.8 : 336.66

Використання методів машинного навчання для оцінювання транспортних засобів агрофірми

Стецишин О.М. Кафедра ІТ – Дубляни, Львівський НУП, 2024.

Кваліфікаційна робота: 70 с. текст. част., 27 рис., 1 табл., 12 арк. ілюстраційного матеріалу, 30 джерел.

Об'єкт дослідження – є процес вибору та оцінки машин та обладнання.

Мета роботи – підвищення ефективності процесу оцінки транспортних засобів за рахунок розробки сучасної моделі із застосуванням елементів штучного інтелекту.

Проведений аналіз предметної області, особливостей оцінки машин та обладнання, проаналізовані підходи до оцінки та обрано найбільш відповідний для задачі даної роботи. Проаналізовані сучасні підходи до побудови інтелектуальних моделей, типи задач машинного навчання та найпоширені методи їх рішення в залежності від потреб задачі. Обрано метод рішення – регресійна модель. Зроблено вибір інструментарію, реалізована модель, здійснено її навчання, проведена оптимізація та перевірка отриманих результатів. Запропоновані елементи графічного інтерфейсу користувача. Розглянуті питання охорони праці та визначення економічної ефективності отриманих рішень.

**Ключові слова:** методи оцінки транспортних засобів, алгоритми машинного навчання, регресійна модель, графічний інтерфейс користувача.

## ABSTARCT

UDC 004.8 : 336.66

Using Machine Learning Methods to Evaluate Agricultural Firms' Vehicles  
Stetsishin O.M. Department of IT – Dublyany, Lviv NEU, 2024.

Qualification work: 70 p. text, 27 pict., 1 tabl., 12 p. illustrative material, 30 sources.

The object of research is the process of selecting and evaluating machinery and equipment.

The purpose of the work is to increase the efficiency of the vehicle appraisal process by developing a modern model using elements of artificial intelligence.

The analysis of the subject area, features of the assessment of machines and equipment is carried out, approaches to assessment are analyzed and the most appropriate for the task of this work is chosen. Modern approaches to building intelligent models, types of machine learning problems and the most common methods of solving them depending on the needs of the problem are analyzed. The solution method is chosen - the regression model. The choice of tools was made, the model was implemented, its training was carried out, optimization and verification of the results obtained. Proposed graphical user interface elements. The issues of labor protection and determination of the economic efficiency of the solutions obtained are considered.

**Keywords:** vehicle valuation methods, machine learning algorithms, regression model, graphical user interface.

## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

- ML – Machine Learning, машинне навчання;
- NN – Neural Networks, нейронні мережі;
- SVM – Support Vector Machine, метод опорних векторів;
- RBF - Radial Basis Function, радіальна базисна функція;
- PCA - principal component analysis, метод головних компонент;
- IoT - Internet of Things, Інтернет речей;
- GPS – Global Positioning System, глобальна система позиціонування;
- GPU - Graphics Processing Unit, графічний процесор;
- TPU - Tensor Processing Unit, тензорний блок обробки;
- API - Application Programming Interface, прикладний програмний інтерфейс;
- GUI - Graphical User Interface, графічний інтерфейс користувача.

## ЗМІСТ

ВСТУП .....	8
1. АНАЛІЗ СТАНУ ПИТАННЯ ТА ПОСТАНОВКА ЗАВДАННЯ .....	10
1.1. Загальні підходи до оцінки транспортних засобів .....	10
1.2. Розрахунок вартості машин на основі майнового підходу .....	11
1.3. Розрахунок собівартості обладнання на основі дохідного підходу .....	16
1.4. Розрахунок вартості машин на основі порівняльного підходу .....	19
Постановка задачі дослідження .....	24
2. ОБҐРУНТУВАННЯ ТА ВИБІР ІНСТРУМЕНТАРІЮ ВИРІШЕННЯ ЗАДАЧІ .....	25
2.1. Алгоритми машинного навчання для оцінювання транспорту .....	25
2.2. Оптимізація параметрів ML-моделей для оцінки транспорту .....	30
2.3. Вибір програмних засобів реалізації системи .....	35
Висновки до розділу .....	39
3. РЕЗУЛЬТАТИ ВИРІШЕННЯ ЗАДАЧІ .....	40
3.1. Підключення до API сервісу оголошень .....	40
3.2. Завантаження та попередня обробка вхідних даних .....	43
3.3. Реалізація та оптимізація моделі машинного навчання .....	51
3.4. Розробка елементів інтерфейсу користувача .....	55
Висновки до розділу .....	60
4. ОХОРОНА ПРАЦІ ТА БЕЗПЕКА У НАДЗВИЧАЙНИХ СИТУАЦІЯХ .....	61
5. ВИЗНАЧЕННЯ ЕФЕКТИВНОСТІ ІНФОРМАЦІЙНОЇ СИСТЕМИ .....	64
ВИСНОВКИ ТА ПРОПОЗИЦІЇ .....	67
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ .....	68

## ВСТУП

Сучасний розвиток аграрної галузі вимагає ефективного використання ресурсів, серед яких особливе місце займають транспортні засоби агрофірм. Їх стан та експлуатаційні характеристики безпосередньо впливають на продуктивність підприємства, витрати на обслуговування та загальну економічну ефективність.

У цьому контексті виникає потреба в розробці інструментів, які дозволяють точно оцінювати технічний стан, вартість та прогнозувати майбутні витрати на обслуговування транспортних засобів. Використання методів машинного навчання є перспективним підходом для вирішення таких завдань завдяки їх здатності аналізувати великі обсяги даних та знаходити закономірності, які важко виявити традиційними методами.

Актуальність роботи зумовлена зростаючою складністю транспортних засобів та значним обсягом даних, пов'язаних з їх експлуатацією. Методи машинного навчання дозволяють автоматизувати процеси оцінювання, знижуючи ризик суб'єктивних помилок, та надають можливість створювати адаптивні моделі, які враховують зміни в експлуатаційних умовах і специфіці використання техніки.

Це особливо важливо для агрофірм, де ефективне управління транспортними засобами сприяє оптимізації логістики, мінімізації витрат на паливо, обслуговування та ремонт.

Розробка методів оцінювання транспортних засобів із використанням машинного навчання має не лише економічне, а й екологічне значення. Точна оцінка технічного стану дозволяє своєчасно виявляти несправності, що сприяє зниженню негативного впливу на довкілля. Крім того, це сприяє зменшенню надмірного використання ресурсів, таких як паливо та запчастини.

У роботі розглядаються сучасні методи аналізу даних та машинного навчання для оцінювання транспортних засобів, аналізуються особливості їх застосування, а також пропонується практичне вирішення поставленої задачі.



Отримані результати можуть бути використані для впровадження автоматизованих систем оцінювання в агрофірмах, що підвищить рівень їх управління транспортною технікою та зменшить експлуатаційні витрати.

## РОЗДІЛ 1

### АНАЛІЗ СТАНУ ПИТАННЯ ТА ПОСТАНОВКА ЗАВДАННЯ

#### 1.1. Загальні підходи до оцінки транспортних засобів

Машини, обладнання та транспортні засоби є найважливішими елементами бізнесу, тобто процесу, що дозволяє економічному суб'єкту отримувати прибуток (дохід).

Оцінка будь-якої складової бізнесу ґрунтується на врахуванні кількості, якості та тривалості вигоди, яку ці об'єкти будуть виробляти протягом певного часу. Оцінювач, вивчивши ринкову інформацію, перераховує ці вигоди в єдину суму, яка і називається поточною вартістю об'єкта. Оскільки процедура оцінки бізнесу багато в чому суб'єктивна (це визнають всі практикуючі оцінювачі), то оцінка будь-якого елемента бізнесу також ґрунтується на суб'єктивному підході. [1-4]

При оцінці машин, обладнання та транспортних засобів використовуються відомі і добре розроблені підходи до оцінки: майновий (собівартісний або витратний), порівняльний (ринковий) і дохідний, а також методи і технологія оцінки. Візуальний огляд цих об'єктів дозволяє досягти досить високого ступеня об'єктивності в оцінці їх фізичного стану; вартісна оцінка будь-якого об'єкта пов'язана з кваліфікацією і посадою оцінювача, а також з поставленим перед оцінювачем завданням.

При майновому (витратному) підході до оцінки об'єкта [5] в першу чергу враховуються витрати на його створення і продаж. Такий підхід часто використовується для визначення ринкової вартості машин, обладнання, транспортних засобів, технічних пристроїв спеціального призначення або унікальних об'єктів. Всі методи, які використовуються в витратному (майновому) підході до оцінки об'єктів, засновані на калькуляції витрат на виробництво, тобто витрат на виготовлення технічного пристрою.

При порівняльному підході до оцінки технічного пристрою враховуються сформовані на ринку ціни купівлі-продажу аналогічних об'єктів, а основним принципом, який використовується оцінювачем в даній ситуації, є порівняння об'єктів. [6] Якщо на ринку є точний аналог об'єкта оцінки, то це полегшує завдання оцінювачу. Якщо точного аналога немає, то підбирається приблизний аналог, який продається на вторинному ринку. Якщо вторинного ринку для цієї продукції немає зовсім, то для порівняння вибирається новий аналогічний об'єкт, технічні характеристики якого підлягають відповідним коригуванням на предмет технічної порівнянності та умов продажу. Очевидно, що такий підхід до оцінки вимагає великих обсягів ринкової інформації та використання адекватних методів порівняння об'єктів. Іншими словами, оцінювач повинен мати велику і постійно оновлювану базу даних по багатьом видам основних виробничих фондів.

Дохідний підхід до оцінки машин, обладнання та транспортних засобів базується на методології оцінки бізнесу. [7] Суть цієї методики полягає в наступному. Спочатку розраховується чистий дохід від експлуатації майнового комплексу (тобто підприємства), потім на його основі або визначається вартість майнового комплексу в цілому, або з нього виділяється вартість окремих елементів (нерухомість, машини, обладнання, транспортні засоби тощо); або визначити дохід від функціонування майнового комплексу (підприємства) і витягнути з нього дохід, отриманий за допомогою окремих елементів цієї виробничої системи, а вже потім (на основі цих часток доходу) визначити вартість нерухомого майна, машин, обладнання, транспортних засобів тощо.

Розглянемо дані підходи дещо детальніше для порівняння та вибору, яким саме будемо користуватись в даній роботі.

## **1.2. Розрахунок вартості машин на основі майнового підходу**

Як зазначалося вище, майновий (вартісний) підхід - це підхід до визначення ринкової вартості об'єкта на основі його відновлювальної вартості з

урахуванням фізичної, функціональної (моральної) та зовнішньої (економічної) амортизації. [8] Відомо також, що відновна вартість оцінюваного об'єкта - це вартість відтворення цього об'єкта в сучасних умовах. Поняття «вартість відтворення» містить ряд умовностей. По-перше, ніхто не збирається створювати оцінювані об'єкти заново (тобто оцінка абстрактна). По-друге, якби ці об'єкти створювалися заново, то при їх створенні використовувалися б нові сучасні матеріали і технології. По-третє, чим старіше об'єкт, що оцінюється, тим більше умовностей і припущень.

Практика розрахунку вартості на відтворювальній основі добре відома, особливо в нашій країні, де до ринкової реформи всі ціни на основні виробничі фонди встановлювалися тільки методами, заснованими на витратному підході. В основу розрахунку лягли витрати на виробництво, тобто витрати на створення об'єкта оцінки.

Економічна категорія "собівартість" відображає поточні витрати на виробництво і реалізацію основних виробничих фондів, виражені в грошовій формі. Алгоритм розрахунку вартості об'єктів, що оцінюються з позицій майнового (вартісного) підходу, може бути виражений таким чином:

- розрахунок прямих витрат на створення об'єкта, тобто оцінка вартості витрачених матеріалів, сировини, палива та енергії, інструментів, заробітної плати виробничого персоналу і т.д.;
- розрахунок цільових витрат на збут і управління, непрямих витрат на оплату праці співробітників;
- розрахунок прибутку творця об'єкта, що оцінюється;
- розрахунок податків, акцизів;
- розрахунок транспортних витрат на доставку до місця монтажу об'єкта;
- розрахунок витрат на такелажні роботи, витрат заводу-виробника на установку, монтаж, підключення і запуск обладнання та інших технічних пристроїв;
- розрахунок непрямих витрат на підбір і покупку обладнання, ліцензійних зборів і податків, зборів за розробку плану монтажу і т.д.;

- розрахунок втрат у вартості об'єкта, що оцінюється в результаті фізичного, функціонального (морального) і зовнішнього (економічного) зносу.

Очевидно, що найбільш складним і трудомістким етапом визначення вартості оцінюваного об'єкта з позицій витратного підходу є визначення вартості його створення. В основному існує два методи визначення собівартості: метод прямого розрахунку і блок методів цілісної оцінки (параметричні методи - кореляційне моделювання, конкретні економічні показники і експертні оцінки). Суть методу прямого розрахунку полягає в тому, що аналізуються всі нормативи економічних показників по кожній конкретній галузі народного господарства і порівнюються з фактичними витратами на виробництво оцінюваного продукту. Цей метод вимагає аналізу великого обсягу інформації, але дає максимально точні результати. Суть параметричних методів полягає у встановленні взаємозв'язку між вартістю об'єкта оцінки, його технічними параметрами і характеристиками. Ці методи, як правило, громіздкі в застосуванні і не відрізняються особливою точністю.

Після визначення вартості виготовлення оцінюваного об'єкта або його аналога будь-яким методом можна розрахувати відновлювальну вартість об'єкта. Залежно від наявної у оцінювача інформації для визначення відновної вартості машин, обладнання, транспортних засобів або інших технічних пристроїв можуть використовуватися такі методи:

- методом розрахунку відновної вартості об'єкта за ціною об'єкта-аналога;
- поелементним методом розрахунку відновлювальної вартості;
- методом, заснованим на аналізі та індексації витрат;
- методом розрахунку агрегованих норм собівартості.

При розрахунку відновлювальної вартості машин та обладнання, транспортних засобів, інших технічних пристроїв за ціною об'єкта-аналога, використовуючи ринкову інформацію, оцінювач вибирає об'єкт, аналогічний об'єкту оцінки за конструктивною конструкцією, матеріалами, що використовуються при його виробництві, технологією виготовлення тощо. Слід зазначити, що є деяка специфіка в підборі об'єкта-аналог. Справа в тому, що

аналоговий об'єкт може мати зовсім інше призначення, ніж об'єкт, що оцінюється; аналогічний об'єкт може використовуватися і в інших галузях народного господарства, ніж об'єкт, що оцінюється; об'єкт повинен бути затребуваний на ринку і мати відому оцінювачу ринкову ціну. За таких умов вибору аналогічного об'єкта професійні оцінювачі вважають за можливе припустити, що вартість аналогічного об'єкта приблизно дорівнює вартості створення об'єкта, що оцінюється. Якщо оцінюється предмет, який не користується попитом на відповідному ринку, то вартість об'єкта приймається рівною собівартості, тобто вартість дорівнює витратам на його виготовлення.

Поелементний метод розрахунку відновної вартості технічних засобів використовується в тому випадку, коли оцінюваний об'єкт складається з декількох одиниць і ринкові ціни на них вже відомі, збірка технічних вузлів проста і може бути виконана самим споживачем. Як приклад можна привести оцінку вартості комп'ютерів по їх зібраним деталям, оцінка вартості виробничої лінії, яка складається з декількох одиниць універсального обладнання тощо. Технологія використання агрегатного методу полягає в наступному. Спочатку складається перелік основних частин оцінюваного об'єкта (прилади, агрегати, агрегати і т.д.), які можна придбати окремо. Далі оцінювач збирає інформацію про ціни по кожній частині об'єкта оцінки, тобто ринкові ціни складових одиниць. Зібрана інформація про ціни окремих частин об'єкта використовується для розрахунку повної вартості об'єкта оцінки в цілому.

В оціночній практиці досить широко використовується метод індексації витрат на їх створення для розрахунку відновної вартості машин, обладнання, транспортних засобів та інших технічних пристроїв. Суть методу полягає в тому, що балансова, початкова або будь-яка інша вартість оцінюваного об'єкта приводиться до поточного рівня цін за допомогою коригувальних індексів (коефіцієнтів). Цей метод використовують, наприклад, при переоцінці основних засобів. Особливість цього методу полягає в тому, що індексується не собівартість (ціна) об'єкта в цілому, а витрати, що складають його вартість. Як правило, інформація про цінові індекси ресурсів, з яких виготовляється об'єкт

оцінки, зазвичай більш доступна, ніж інформація про основні індекси готової продукції, і це полегшує вирішення проблеми. Далі оцінювач підсумовує проіндексовані витрати і таким чином визначає вартість об'єкта на момент оцінки.

Якщо на об'єкт оцінки є конструкторська документація або якщо оцінювач може отримати відповідну інформацію про продукцію від виробника, то з'являється можливість розрахувати вартість виготовлення технічного пристрою та розрахувати його відновну вартість за укрупненими нормативами. Такі прийоми використовуються в науково-дослідних інститутах і конструкторських бюро для техніко-економічних обґрунтувань створення нових виробів. Ці методи можуть бути застосовані і в оціночній практиці.

Таким чином, для застосування методики розрахунку вартості машин, обладнання, транспортних засобів та інших технічних засобів за допомогою укрупнених нормативів необхідно накопичувати і систематично оновлювати дані з області виробництва, розробляти і систематично оновлювати нормативи собівартості.

Розглянуті вище методи розрахунку відновної вартості машин, обладнання та інших технічних пристроїв з позицій вартісного підходу до їх оцінки мають як переваги, так і недоліки.

В якості переваг можна відмітити наступне:

- універсальність методів, тобто вони можуть бути застосовані до будь-яких видів технічних пристроїв;
- можливість більш точно врахувати вплив всіх видів амортизації оцінюваного об'єкта, визначити страхове відшкодування, провести оцінку майна при його поділі;
- можливість повноцінно використовувати фінансові та бухгалтерські документи для вирішення завдань, пов'язаних з оцінкою, тобто отримані результати можуть бути легко обґрунтовані.

Як недоліки можна вказати такі:

- вартість матеріальних і трудових витрат при виробництві об'єктів оцінки, як правило, завищена;
- розрахунки занадто детальні і тому трудомісткі;
- витрати можуть бути навмисно спотворені на виробничих підприємствах і можуть істотно відрізнятись від середніх витрат по галузі.

### **1.3. Розрахунок собівартості обладнання на основі дохідного підходу**

З розвитком ринкових відносин зростає роль дохідного підходу в оцінці вартості виробничих потужностей. [9] Дохідний підхід до оцінки виробничих потужностей базується на методології оцінки бізнесу, суть якої полягає в наступному. Спочатку розраховується чистий дохід від функціонування всієї виробничої системи (наприклад, підприємства в цілому), а потім на його основі або визначається величина і відокремлюється від неї вартість машин і обладнання, інших об'єктів оцінки. Або визначається дохід від всієї системи і з неї виділяється дохід, одержуваний за допомогою виробничих машин і обладнання, інших об'єктів оцінки, і тільки потім, виходячи з цієї частини доходу, визначається вартість виробничих машин і обладнання, інших об'єктів оцінки.

Перш ніж перейти до характеристики методів оцінки вартості технічних пристроїв з позицій дохідного підходу, нагадаємо кілька загальних положень, яких дотримуються оцінювачі при вирішенні цього завдання. По-перше, оцінювачі прийшли до єдиної думки, що вартість машин, обладнання, транспортних засобів та іншої продукції виробничого призначення визначається розміром, якістю і тривалістю періоду отримання вигоди, яку ці об'єкти (як очікується) принесуть в майбутньому. Оцінювач перетворює ці вигоди в єдину грошову суму, яка називається поточною вартістю об'єкта. По-друге, перед тим як обчислювати ці вигоди від володіння, користування або розпорядження об'єктом оцінки в грошовій формі, оцінювач повинен визначити:



- максимально точно визначити суму доходу, яку власник може отримати за допомогою даного об'єкта;
- час, протягом якого власник буде отримувати дохід від використання даного об'єкта;
- відсоток ризику, тобто ймовірність того, що буде отриманий дохід від використання даного об'єкта протягом прогнозованого періоду часу;
- норма доходу, тобто процентне співвідношення між чистим доходом та інвестованим капіталом або між доходом, отриманим об'єктом оцінки, та вартістю цього об'єкта.

Дохідний підхід до оцінки машин та обладнання поєднує в собі три методи:

- метод капіталізації прибутку;
- метод дисконтованого грошового потоку;
- метод рівноефективного аналога.

Всі три методи засновані на прогнозуванні майбутніх доходів підприємства. Ці майбутні доходи створюються підприємством в цілому, включаючи машини та обладнання, будівлі, споруди, оборотні кошти, нематеріальні активи, тобто весь виробничий комплекс. Тому для того, щоб використовувати дохідний підхід до оцінки будь-якої частини виробничого комплексу, будь то машини та обладнання, будівлі, споруди тощо, оцінювачі (використовуючи будь-який метод) використовують технологію поетапного вирішення поставленого завдання.

Метод капіталізації прибутку - це процес перетворення майбутнього доходу в єдину суму теперішньої вартості. При цьому необхідно враховувати розмір майбутнього доходу, коли дохід повинен бути отриманий і тривалість цього доходу. В основі методу лежить принцип складних відсотків. Цей метод в оцінці бізнесу отримав широке застосування, хоча в його використанні є труднощі. Перш за все, складно спрогнозувати прибуток (особливо в російських умовах), а також складно підібрати величину коефіцієнта капіталізації.

Метод дисконтування чистого доходу. Цей метод має ряд переваг. По-перше, він дозволяє визначити прогнозовану (майбутню) рентабельність підприємства. Це важливо для інвестора, так як при визначенні прогнозованої рентабельності враховуються ризик вкладення інвестицій, рівень інфляції в країні, зміна ринкової кон'юнктури, економічне старіння підприємства і т.д.

Метод рівноефективного аналога, як і два попередніх методи, заснований на дохідному підході до оцінки конкретного об'єкта. Його суть полягає в наступному:

- по-перше, вибирається аналог (базовий об'єкт), який виконує аналогічні функції, але може відрізнятися від оцінюваного об'єкта за експлуатаційними характеристиками, терміном служби, якістю продукції, що виготовляється з його допомогою, та іншими показниками;

- по-друге, визначається дохід від об'єкта, що оцінюється, але не в повному обсязі, а тільки в тій його частині, на яку дохід оцінюваного об'єкта відрізняється від доходу аналогічного (базового) об'єкта.

Вартість об'єкта визначається з ціни базового об'єкта і аналога за умови однакової їх рентабельності. В основі методу лежить так звана теорія ефективності техніки. Математична модель, яка використовується в цьому процесі, виводиться з формули для розрахунку порівняльного економічного ефекту.

Всі три методи дохідного підходу до оцінки технічних пристроїв використовуються в наступних випадках:

- по-перше, коли є можливість розрахувати чистий дохід від системи, в якій більша частина доходу надходить від машин і обладнання, в цих випадках дохід розрахувати не складно і може бути застосований метод капіталізації прибутку або метод дисконтованого чистого прибутку;

- по-друге, широко використовується метод капіталізації доходу, коли є можливість точно оцінити вартість земельної ділянки і вартість будівлі;

- по-третє, слід зазначити, що всі методи дохідного підходу мають свої переваги і недоліки.

Переваги: оптимальний спосіб вимірювання сумарної амортизації всіх активів підприємства, можливість враховувати вплив економічних факторів на вартісну оцінку об'єктів, логічне обґрунтування прийнятих рішень.

Недоліки: складність виділення окремих активів підприємства для їх індивідуальної оцінки, суб'єктивність прогнозів рентабельності і вибору норм доходу.

В цілому слід мати на увазі, що дохідний підхід все ж дає достовірні результати, коли є можливість впевнено прогнозувати такі показники, як грошові потоки, витрати, процентні ставки доходів і т.д. В умовах нестабільної економіки важко бути впевненим у таких прогнозах.

#### **1.4. Розрахунок вартості машин на основі порівняльного підходу**

Ринковий підхід до оцінки вартості об'єктів найбільш прийнятний для тих видів технічних пристроїв, які мають розвинений вторинний ринок (автомобілі, багато видів верстатів, кораблі, літаки, інше нестандартне обладнання). Метод оцінки таких об'єктів ґрунтується на визначенні ринкових цін, які адекватно відображають «вартість» одиниці технічного пристрою в його поточному стані. Основним принципом ринкового підходу є порівняння об'єктів оцінки з аналогічним або новим товаром, який реалізується на вторинному ринку. Очевидно, що такий підхід до оцінки вимагає великих обсягів ринкової інформації та використання адекватних методів порівняння об'єктів. [10-12]

Спосіб організації інформаційного забезпечення процесу оцінки визначається самим оцінювачем. Як правило, бази даних, що складають основу даної інформаційної системи, представлені в алфавітному порядку виробниками, описують методи кодування даних і основні класифікаційні ознаки, інші принципи збору та аналізу даних, а також можливості їх об'єднання. Такі бази даних містять інформацію про реальні ринкові ціни на нові та вживані технічні пристрої; на ціни в прайс-листах виробників (як правило, нове обладнання); про дилерські ціни; про ціни пропозицій (оферт),

які надходять у результаті письмових або усних запитів оцінювачів; Фіксуються індекси цін. Для практикуючих оцінювачів цінність такої інформації очевидна, методи обробки та аналізу відомі, а пошук джерел і технологія збору необхідної інформації представляють певні труднощі.

Величина цін реальних угод полягає в тому, що вони досить точно відображають рівень ринкової ціни в конкретний момент часу і дають можливість отримати інформацію про деталі угоди. Цей фактор підвищує достовірність результатів оцінки будь-якого об'єкта. Джерелами такої інформації є особисті зв'язки оцінювачів, преса та спеціальні видання. Ціни прайс-листів і каталогів, тобто офіційно опублікована інформація, точно відображають тип об'єкта, обладнання та інші технічні параметри. При цьому вони не враховують можливі знижки, яких можна досягти в процесі купівлі-продажу об'єкта. Тому цінові характеристики, отримані з цих джерел, можуть використовуватися тільки в якості базових цін, які повинні бути скориговані, і тільки після цього вони можуть бути використані для визначення вартості того чи іншого об'єкта.

На ринку є ще один вид цінової інформації - це ціни пропозицій (пропозицій на продаж або купівлю). Цінність такої інформації полягає в тому, що вона повністю відображає технічні характеристики об'єкта і умови його продажу, але містить, як правило, завищену або занижену ціну. Крім того, отримати його досить легко (за запитом, особистими зв'язками і т.д.), але при цьому оцінювач повинен дотримуватися певну тактику поведінки.

В основі ринкового підходу лежить принцип порівняння, тобто порівнюються оцінювані і схожі об'єкти. Цей підхід реалізований в методі прямого порівняння продажів. Технологія використання цього методу полягає в наступному:

- спочатку вибирається декілька (не менш 3-5) схожих об'єктів, порівнюються їх техніко-економічні параметри і фіксуються відмінності;
- по-друге, зафіксовані різниці оцінюються у вартісному вираженні і включаються в ціну продажу аналога;

- по-третє, визначається відновлювальна вартість технічного пристрою або групи об'єктів.

При підборі аналога перевага віддається тим об'єктам, які, як і оцінюваний об'єкт, були виготовлені одним і тим же виробником і в одній країні, мають однакове функціональне призначення і кваліфікаційну схожість, а також мають хоча б часткову технологічну схожість.

Іноді використовуються поправки на технічну порівнянність. Тут є поправки на стандартні розміри порівнюваних об'єктів (потужність, вантажопідйомність, продуктивність і т.д.), їх конфігурацію (наявність додаткових приладів і пристроїв), вік і якість виробу, ступінь фізичного і функціонального зносу, розташування об'єкта при продажу (в місці його використання, на складі дилера тощо).

Вибір технічних характеристик, які мають найбільший вплив на ціну товару, як правило, вирішує фахівець. Зазвичай їх кількість не повинна перевищувати кількість оцінювачів, оскільки, як показує практика, велика кількість обраних параметрів не підвищує точність аналізу (найчастіше використовується від трьох до семи параметрів). Технічні характеристики, які використовуються в математичній моделі, повинні бути незалежні один від одного, хоча на практиці це дуже складно реалізувати, так як всі одиниці у виробі функціонально взаємозалежні. Якщо технічні параметри сильно залежать один від одного, то їх бажано виключити зовсім.

Для побудови параметричних моделей цін найбільш широко використовується метод кореляційно-регресійного аналізу. [13-16] Його суть полягає в наступному: для групи аналогічних об'єктів встановлюється форма зв'язку між показником собівартості (наприклад, ціною, собівартістю і т.д.) і технічним параметром об'єкта (наприклад, потужністю, вагою, обсягом і т.д.). Отримані точки наносяться на графік.

Потім оцінювач на свій розсуд вибирає тип кривої (прямої) лінії на графіку і потім за допомогою математичної функції визначає плавну лінію регресії (в конкретному випадку регресія може бути і прямою лінією). Лінія регресії

показує загальну тенденцію досліджуваної залежності і згладжує випадкові викиди від впливу побічних факторів (рис. 1.1).

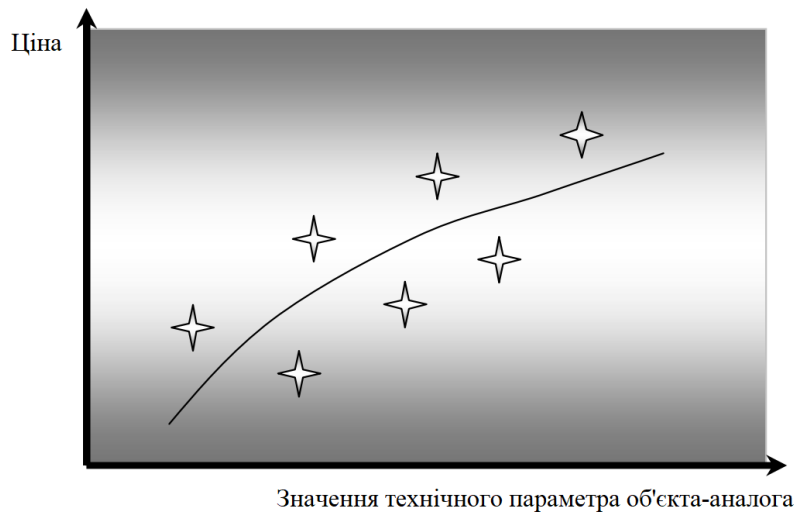


Рисунок 1.1 - Тенденція взаємозалежності ціни об'єкта від його технічного параметра

Цей метод дозволяє визначити очікувані витрати на виробництво нової продукції вже на етапі розробки дизайну. Для цих цілей використовуються залежності витрат на виробництво від планових технічних параметрів об'єкта. Особливо це важливо при виробництві складної дорогої техніки (кораблі, літаки, прилади великої потужності і т.д.). Кореляційно-регресійний аналіз використовується багатьма фірмами для обробки інформації про ціни і параметри продукції (машин і обладнання) з метою визначення конкурентоспроможності власної продукції, встановлення середнього ринкового рівня цін і т.д. Методика побудови економічних моделей, в тому числі і цінових, добре відома (в статистиці), але конкретні дані, отримані за допомогою цих моделей, є комерційною таємницею.

Така аналітична робота з оцінки вартості машин і обладнання проводиться у вітчизняній практиці, як правило, за допомогою технологій комп'ютерної оцінки. Велике значення тут має процес формування бази даних. При формуванні бази даних, необхідної для розрахунку собівартості продукції,

враховуються два важливих фактори - однорідність і репрезентативність зібраної інформації. Однорідність інформації забезпечується при наявності невеликої розкиду зібраних даних у часі, однакових комерційних умов продажу, а також незначних відмінностей в конструкції порівнюваних технічних пристроїв і комплектації цих виробів. І хоча абсолютно ідентичних товарів не існує, тим не менш, можна привести отриману з різних джерел інформацію в єдиний момент часу, однакові терміни поставки і оплати, використовуючи індекси цін, курси валют за відповідний період часу, з урахуванням відмінностей в транспортуванні, облік витрат при переході з одних умов поставки на інші і т.д. Якщо для включення в математичну модель вибирається інформація про ціни і параметри основних компаній, що випускають даний вид технічних пристроїв.

Оскільки в практичній роботі оцінювач стикається з необхідністю накопичення та обробки інформації про різні види машин та обладнання, їх класифікацію, споживчі властивості та якісні показники, про нормативний термін служби та фізичний стан обладнання, про ринкові ціни та індекси цін для галузей та груп товарів, про курси валют, норми амортизації тощо, то в даній ситуації неможливо вирішити на сучасному рівні такі проблеми проблема оцінки вартості засобів праці. В даний час за допомогою комп'ютерних технологій здійснюється пошук аналогової продукції, оцінка конкретних об'єктів, розробка математичних моделей, що встановлюють функціональну залежність між цінами на продукцію і її технічними параметрами.

Окремо треба вказати, що саме ринковий підхід дозволяє в повній мірі використовувати машинне навчання для оцінки транспортних засобів. Одна з основних проблем, які можуть виникнути на цьому шляху - недостатня кількість об'єктів-аналогів (або нерозвиненість даного сегменту ринку). В такому випадку втілюють різноманітні додаткові моделі, поправки та коефіцієнти, які можуть покращити якість моделі машинного навчання.

## Постановка задачі дослідження

Підсумовуючи описане вище, можна говорити, що для моделей машинного навчання треба обирати ринковий (або порівняльний) підхід з урахуванням того, що оцінювач (або розробник) мусить забезпечити таку модель достатньою кількістю вхідної інформації. Враховуючи, що сучасні транспортні засоби налічують десятки різноманітних технічних параметрів навіть загального характеру - то база даних об'єктів-аналогів мусить мати, як мінімум, сотні аналогічних або близьких ринкових пропозицій. Тільки в такому випадку можна казати про те, що модель машинного навчання буде адекватно оцінювати транспортний засіб користувача.

В якості задач дослідження обрані наступні:

- провести дослідження предметної області, проаналізувати існуючі методи розв'язання задач оцінки та обрати найефективніші з них;
- визначити відповідний математичний інструментарій для розв'язання поставленої задачі;
- розробити модель машинного навчання, підготувати вхідні дані, здійснити навчання моделі, провести за потреби її оптимізацію, отримати результати та сформулювати висновки.



## РОЗДІЛ 2

### ОБҐРУНТУВАННЯ ТА ВИБІР ІНСТРУМЕНТАРІЮ ВИРІШЕННЯ ЗАДАЧІ

#### 2.1. Алгоритми машинного навчання для оцінювання транспорту

Машинне навчання (ML) є ключовим інструментом у сучасному аналізі транспортних засобів, оскільки дозволяє автоматизувати процес оцінки їх технічного стану, продуктивності та економічної ефективності. Алгоритми ML забезпечують обробку великих обсягів даних і знаходження закономірностей, які важко ідентифікувати традиційними методами. [17-19] У цій статті розглянемо найпопулярніші алгоритми, їх застосування для оцінки транспорту, переваги та обмеження.

Одним із найпростіших алгоритмів є лінійна регресія [20], яка використовується для прогнозування безперервних показників, таких як витрати на пальне, час обслуговування чи пробіг. Формально модель лінійної регресії можна записати як:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_n x_n + \epsilon,$$

де  $y$  – прогнозоване значення,  $x$  – вхідні ознаки,  $\beta_i$  – коефіцієнти, що визначають вплив кожної ознаки, а  $\epsilon$  – похибка. Лінійна регресія має обмеження у випадках, коли зв'язок між ознаками і цільовою змінною нелінійний. Хоча є методи, які дозволяють обійти це обмеження, наприклад, генерація поліноміальних або періодичних ознак,. Інший шлях - використання загальних регресійних алгоритмів, які не обмежені лінійними рішеннями.

Для роботи з класифікаційними задачами, такими як визначення класу технічного стану (наприклад, "відмінний", "задовільний", "критичний"), часто застосовується логістична регресія. [21] Вона прогнозує ймовірність належності об'єкта до певного класу за допомогою сигмоїдної функції:

$$P(y = 1 | x) = \frac{1}{1 + e^{-(\beta_0 + \sum_{i=1}^n \beta_i x_i)}}.$$

Логістична регресія підходить для бінарної класифікації, проте її можна розширити для багатокласових задач (методом "один проти всіх").

Одними з найбільш універсальних алгоритмів є дерева рішень та їх ансамблеві методи, такі як випадковий ліс (Random Forest). [22] Дерева рішень розділяють дані на підгрупи на основі їхніх властивостей, створюючи структуру, що нагадує дерево. Однак, вони схильні до перенавчання, що компенсується ансамблевими підходами. Random Forest використовує кілька дерев, кожне з яких створюється на основі випадкової вибірки даних і підмножини ознак. Результати прогнозу визначаються шляхом голосування дерев для класифікації або усереднення у випадку регресії.

Іншим потужним ансамблевим методом є градієнтний бустинг [23], представлений алгоритмами, такими як XGBoost, LightGBM та CatBoost. Градієнтний бустинг комбінує послідовність слабких моделей (наприклад, дерев рішень), кожна з яких виправляє помилки попередньої. Це дає змогу досягати високої точності, особливо для задач прогнозування витрат чи ефективності транспорту.

Методи кластеризації [24], такі як K-means або DBSCAN, використовуються для групування транспортних засобів за схожими характеристиками, наприклад, за рівнем витрат на обслуговування чи продуктивності. K-means визначає центроїди кластерів і мінімізує відстань між точками в кожному кластері та їх центром. DBSCAN, своєю чергою, ідентифікує щільні зони у даних, що корисно для виявлення аномалій, таких як транспортні засоби з незвичайними витратами.

Для роботи з великими обсягами даних, включаючи телеметричні записи, застосовуються нейронні мережі. [25,26] Простий штучний нейрон має ваги та функцію активації, що визначає його вихід. У глибоких нейронних мережах

(Deep Learning) використовується багат шарова структура, яка дозволяє моделювати складні зв'язки між входами і виходами. Наприклад, рекурентні нейронні мережі (RNN) та їх модифікації, такі як LSTM (Long Short-Term Memory), ефективно працюють із часовими рядами, прогножуючи технічний стан транспорту на основі історичних даних.

Ще одним важливим методом є підтримуючий векторний апарат (Support Vector Machine, SVM), який добре працює в задачах класифікації та регресії. SVM намагається знайти гіперплощину, що максимально розділяє дані різних класів. У випадку нелінійних даних використовується ядрова функція, наприклад, RBF (радіальна базисна функція).

Важливим аспектом є попередня обробка даних, яка впливає на якість моделей. Техніки нормалізації, відбору ознак, балансування класів та зменшення розмірності (наприклад, за допомогою методу головних компонент, PCA) дозволяють підвищити ефективність алгоритмів.

У сучасній практиці вибір алгоритму залежить від типу задачі, доступних даних і вимог до продуктивності. Наприклад, для швидкої реалізації із середньою точністю часто використовують дерева рішень чи Random Forest. У задачах з високими вимогами до точності, але великими обчислювальними ресурсами, перевага віддається градієнтному бустингу чи нейронним мережам.

Для оцінки транспортних засобів кожен алгоритм має свої сильні та слабкі сторони, і при прийнятті рішень обов'язково треба враховувати, що алгоритми машинного навчання мають низку обмежень при їх використанні в транспортній аналітиці. Однією з основних проблем є якість даних, адже моделі потребують точних і достовірних входів. У транспортних системах часто трапляються пропуски у даних, наявність шуму або помилкових значень, наприклад, через збої датчиків. Це може призводити до зниження точності моделі або її некоректної роботи.

Додатковою складністю є перенавчання моделей, яке виникає, коли алгоритми занадто точно підлаштовуються під тренувальні дані. Наприклад, дерева рішень добре працюють із навчальними даними, але можуть

демонструвати погані результати на нових вибірках. У таких випадках потрібні методи регуляризації [27] або використання ансамблевих підходів, що зменшують ризик перенавчання.

Складність інтерпретації є ще одним важливим фактором. Наприклад, нейронні мережі або градієнтний бустинг працюють як "чорні ящики", що ускладнює пояснення результатів їх роботи користувачам. У практичній експлуатації транспорту інженерам і керівникам необхідно розуміти, чому модель рекомендує певні рішення. Ця проблема частково вирішується інструментами інтерпретації, такими як методи SHAP або LIME.

Ще одним обмеженням є потреба в обчислювальних ресурсах. Більшість сучасних алгоритмів, особливо глибокі нейронні мережі, вимагають потужного апаратного забезпечення для навчання та виконання прогнозів. У багатьох агрофірм немає доступу до кластерних систем або хмарних технологій, що може обмежувати застосування таких моделей.

Залежність результатів від правильно підібраних параметрів є ще однією проблемою. Наприклад, у K-means потрібно точно визначити кількість кластерів, а в Random Forest – оптимальну кількість дерев. Невдалий вибір параметрів може призвести до зниження ефективності моделей, тому для їх налаштування часто застосовуються автоматизовані методи оптимізації.

Застосування Інтернету речей (IoT) у транспортних системах значно розширює можливості машинного навчання. Завдяки підключеним пристроям у реальному часі збираються великі обсяги телеметричних даних, які включають GPS-координати, інформацію від датчиків стану транспорту та навіть відеозаписи. Ці дані створюють основу для більш точного і своєчасного аналізу.

Алгоритми машинного навчання можуть використовувати ці дані для прогнозування несправностей, аналізуючи показники двигуна, температури та інших параметрів. Це дозволяє завчасно виявляти зношення окремих компонентів і мінімізувати ризик несподіваних поломок. Використання GPS-

даних і записів руху сприяє оптимізації маршрутів, зменшуючи витрати пального та підвищуючи ефективність логістики.

Моделі машинного навчання також ефективно виявляють аномалії в даних. Наприклад, якщо транспортний засіб демонструє незвично високі витрати пального або значні коливання температури двигуна, це може свідчити про приховані технічні проблеми. Такі підходи автоматизують процеси моніторингу та технічного обслуговування, значно знижуючи витрати на експлуатацію транспорту.

Таблиця 2.1 містить основні критерії порівняння популярних методів машинного навчання.

Таблиця 2.1 - Порівняння методів машинного навчання

Метод	Переваги	Недоліки	Приклади застосування
Лінійна регресія	Проста реалізація, інтерпретованість	Неефективність для нелінійних залежностей	Прогнозування витрат на пальне
Логістична регресія	Підходить для класифікації, висока швидкість обчислень	Погана масштабованість для складних задач	Визначення класу технічного стану
Дерева рішень	Зрозумілість структури, здатність до нелінійного поділу	Схильність до перенавчання	Розподіл транспорту за групами
Випадковий ліс	Стійкість до перенавчання, висока точність	Високі вимоги до пам'яті	Аналіз продуктивності транспортного парку
Гradientний бустинг	Висока точність, здатність до роботи з неякісними даними	Часо- та ресурсоемність	Прогнозування зносу техніки
Нейронні мережі	Ефективність для великих даних, складних залежностей	Неінтерпретованість, необхідність великої кількості даних	Аналіз телеметрії та прогнозування стану
K-means	Простота реалізації, швидкість роботи	Необхідність визначення кількості кластерів	Групування техніки за рівнем витрат
SVM	Висока ефективність у класифікації малих наборів даних	Часо- та ресурсоемність на великих даних	Виявлення дефектних транспортних засобів

Таким чином, застосування машинного навчання для оцінки транспортних засобів агрофірми дозволяє не лише автоматизувати цей процес, а й забезпечити високий рівень точності прийняття рішень. Ефективність цих методів зростає із розвитком обчислювальних технологій і доступом до великих обсягів даних, що відкриває нові можливості для транспортної аналітики.

## **2.2. Оптимізація параметрів ML-моделей для оцінки транспорту**

Одним із важливих аспектів побудови ефективних моделей машинного навчання є правильна оптимізація їх параметрів, яка безпосередньо впливає на точність прогнозів, швидкість роботи та здатність до узагальнення. У транспортній галузі, де точність моделі може визначати успіх чи провал технічного обслуговування, маршрутизації чи прогнозування зносу, значення гіперпараметрів є критично важливим.

Оптимізація параметрів моделей передбачає тонке налаштування зовнішніх характеристик алгоритму, які не оновлюються під час навчання, але визначають ефективність цього процесу. Для задач у сфері транспорту це може означати, наприклад, вибір оптимального рівня глибини дерева рішень для виявлення несправностей транспортного засобу або налаштування швидкості навчання для прогнозування витрат пального.

Роль налаштування параметрів у транспортних задачах підкреслюється ще й тим, що ці задачі зазвичай характеризуються великими обсягами даних, високою залежністю від зовнішніх факторів (наприклад, погодних умов, стану доріг) і складними взаємозв'язками між змінними. У такому контексті недостатньо просто використати "дефолтні" налаштування алгоритмів: для отримання якісного результату потрібно враховувати особливості конкретних даних, обчислювальні можливості та практичні потреби користувача.

Оптимізація параметрів моделей машинного навчання є процесом вибору таких гіперпараметрів, які максимізують ефективність алгоритму.

Гіперпараметри — це зовнішні параметри моделі, що встановлюються до початку тренування. Вони впливають на структуру, навчання та продуктивність моделі. У контексті машинного навчання їх необхідно відрізнити від внутрішніх параметрів (weights), які модель автоматично оновлює під час тренування.

Наприклад, у лінійній регресії внутрішніми параметрами є коефіцієнти, які визначають нахил і зсув прямої. У нейронних мережах це ваги та зміщення, які модель адаптує під час процесу навчання. Натомість гіперпараметри включають такі аспекти, як швидкість навчання (learning rate), кількість нейронів у прихованому шарі чи кількість дерев у алгоритмах ансамблів.

Основна мета оптимізації гіперпараметрів полягає у досягненні найкращого балансу між перенавчанням (overfitting) і недонавчанням (underfitting). Перенавчання виникає, коли модель надто точно відтворює навчальні дані, включаючи їхній шум, але втрачає здатність до узагальнення. Недонавчання, навпаки, означає, що модель не встигає навчитися коректно відображати навіть основні закономірності в даних.

Щоб уникнути цих проблем, вибір гіперпараметрів має враховувати специфіку даних, наприклад, розподіл змінних, наявність пропусків чи аномалій, а також цільову метрику, за якою оцінюється продуктивність моделі. Для класифікаційних задач важливими метриками є precision, recall, F1-score, тоді як у регресійних задачах використовуються MSE, RMSE або MAE.

Важливим є також розуміння складності взаємозв'язків між параметрами. Наприклад, у градієнтному бустингу зменшення швидкості навчання (learning rate) зазвичай потребує збільшення кількості дерев (n\_estimators) для досягнення аналогічної точності, але це збільшує час тренування.

Отже, гіперпараметри можна умовно поділити на дві категорії:

- структурні параметри, які визначають архітектуру моделі (глибина дерева, кількість нейронів);
- параметри навчання, які впливають на процес оптимізації (learning rate, batch size).

Оптимізація параметрів може виконуватися різними методами, які відрізняються за ступенем автоматизації, складністю реалізації та ефективністю. Основні підходи включають ручну настройку, систематичний перебір (Grid Search), випадковий пошук (Random Search) і більш сучасні методи, такі як Байєсівська оптимізація, генетичні алгоритми чи методи гіперпараметричного пошуку, засновані на глибокому навчанні.

Ручна настройка є найпростішим підходом і часто використовується на початкових етапах розробки моделей. Дослідник вибирає кілька значень параметрів, перевіряє їх ефективність і зупиняється на оптимальному варіанті. Проте цей метод має серйозні недоліки: він займає багато часу, залежить від досвіду аналітика та ефективний лише для моделей з обмеженою кількістю параметрів.

Grid Search є систематичним підходом, який перебирає всі можливі комбінації параметрів із попередньо визначених діапазонів. Наприклад, якщо потрібно вибрати оптимальну глибину дерева рішень та мінімальну кількість зразків для поділу вузла, Grid Search протестує всі можливі пари цих значень. Хоча цей підхід гарантує знаходження оптимального варіанту, його обчислювальна вартість зростає експоненційно зі збільшенням кількості параметрів. Для задач оцінки транспорту, де моделі можуть мати десятки гіперпараметрів, Grid Search стає малопридатним.

Random Search надає більш ефективну альтернативу. Він генерує випадкові комбінації параметрів і оцінює їхню продуктивність. Дослідження показали, що Random Search у багатьох випадках досягає таких самих результатів, як і Grid Search, але значно швидше, оскільки не тестує всі можливі варіанти. Особливо цей метод корисний, коли не всі параметри рівною мірою впливають на ефективність моделі.

Байєсівська оптимізація використовує статистичні моделі, щоб прогнозувати ефективність параметрів до їх тестування. Наприклад, цей метод будує ймовірнісний розподіл функції мети (objective function), яка оцінює продуктивність моделі. Використовуючи ці прогнози, Байєсівська оптимізація



спрямовує пошук у найбільш перспективні області параметрів. Цей підхід особливо ефективний для моделей, що працюють з великими наборами даних і складними функціями мети.

Інші сучасні методи включають:

- генетичні алгоритми, які імітують природний відбір, генеруючи "популяції" параметрів та поступово покращуючи їх через мутацію й кросовер;
- Nuregband, що ефективно управляє обчислювальними ресурсами, скорочуючи навчання для моделей із низькою продуктивністю на ранніх етапах;
- методи на основі глибокого навчання, які створюють метамоделі для прогнозування ефективності гіперпараметрів.

Вибір конкретного методу оптимізації залежить від задачі, доступних обчислювальних ресурсів та складності моделі. Для задач оцінки транспорту, де потрібно враховувати реальні обмеження часу і ресурсів, випадковий пошук чи Байєсівська оптимізація можуть бути найбільш придатними.

Для ефективного вирішення задач оцінювання транспортних засобів найчастіше використовуються такі моделі машинного навчання, як лінійна регресія, дерева рішень, градієнтний бустинг і нейронні мережі. Кожна з цих моделей має специфічні гіперпараметри, які впливають на їхню продуктивність.

У випадку лінійної регресії основним параметром є коефіцієнт регуляризації. Він допомагає уникати переобладнання, додаючи до функції втрат штраф за занадто великі значення ваг. Наприклад, у Lasso-регресії функція втрат визначається як:

$$L = \sum_{i=1}^n (y_i - \hat{y}_i)^2 + \lambda \sum_{j=1}^p |\beta_j|,$$

де  $\lambda$  — коефіцієнт регуляризації, а  $\beta_j$  — ваги моделі. Налаштування  $\lambda$  проводиться через перехресну валідацію, щоб знайти баланс між точністю моделі та її здатністю до узагальнення.

Для дерев рішень ключовими параметрами є глибина дерева (`max_depth`), мінімальна кількість зразків для поділу вузла (`min_samples_split`) і мінімальна кількість зразків у листі (`min_samples_leaf`). Наприклад, занадто велика глибина дерева може призвести до переобладнання, тому рекомендується тестувати діапазон значень (наприклад, від 3 до 10). Мінімальна кількість зразків для поділу вузла також впливає на складність моделі: більші значення сприяють зниженню переобладнання, але можуть зменшити точність.

Для градієнтного бустингу (наприклад, XGBoost, LightGBM) основними гіперпараметрами є:

- `learning_rate` — швидкість навчання, яка зазвичай налаштовується в діапазоні 0,01–0,3;
- `n_estimators` — кількість дерев, що корелює з `learning_rate` (менші значення `learning_rate` вимагають більшої кількості дерев);
- `max_depth` — максимальна глибина дерева, яка контролює здатність моделі до узагальнення.

Наприклад, для задач оцінки витрат пального використання невеликих значень `learning_rate` (0.05–0.1) у поєднанні з більшою кількістю дерев (500–1000) часто дає найкращі результати.

Для нейронних мереж найважливіші параметри включають:

- кількість шарів і кількість нейронів у кожному шарі;
- `learning_rate`, який визначає, як швидко модель адаптує свої ваги;
- `batch_size` — розмір партії даних, що використовується для одного оновлення ваг.

Наприклад, для задач прогнозування технічного стану транспорту оптимальним може бути використання 2–3 прихованих шарів із 50–100 нейронами в кожному, а `learning_rate` налаштовується в діапазоні 0,001–0,01. Використання регуляризації (`dropout`) також може бути корисним для запобігання переобладнанню.

Для задач оцінки транспорту рекомендується використовувати перехресну валідацію або спеціалізовані методи, такі як Bayesian Optimization, для підбору

оптимальних параметрів. Це дозволяє забезпечити баланс між точністю та швидкістю навчання, враховуючи обмеження ресурсів.

Для задач оцінювання транспортних засобів ефективність оптимізації також залежить від специфічних вимог до моделі. Наприклад, для прогнозування витрат пального можуть бути важливі точність і швидкість роботи, тоді як для оцінки технічного стану транспорту — чутливість до рідкісних подій, таких як несправності.

Таким чином, оцінка ефективності оптимізації повинна базуватися на комбінації якісних метрик, стабільності моделі, витрат на обчислення та специфічних вимог до задачі. Це дозволяє не тільки знайти найкращу конфігурацію параметрів, а й забезпечити практичну цінність моделі в реальних умовах.

### **2.3. Вибір програмних засобів реалізації системи**

Програмні засоби відіграють ключову роль у реалізації ML-рішень. Вони забезпечують не лише можливість будувати та налаштовувати моделі, а й інтегрувати їх у реальні виробничі системи. Для аграрного сектору програмні інструменти також стають містком між традиційним веденням господарства та сучасними технологіями Інтернету речей (IoT), що дозволяє збирати та аналізувати дані в реальному часі.

Реалізація машинного навчання у аграрному секторі потребує потужних програмних платформ і бібліотек, які забезпечують зручність побудови, навчання та тестування моделей. Існує кілька широко визнаних інструментів, що можуть бути ефективно використані для задач, пов'язаних із аналізом даних і прогнозуванням у сільському господарстві.

Однією з найпопулярніших платформ є TensorFlow, яка пропонує широкі можливості для створення складних моделей машинного навчання, включаючи нейронні мережі. TensorFlow дозволяє працювати як із класичними ML-алгоритмами, так і з методами глибокого навчання. Для аграрних задач

TensorFlow може бути корисним при розробці моделей для аналізу зображень, наприклад, для виявлення хвороб рослин за фотографіями.

Інша потужна платформа — PyTorch, яка відома своєю гнучкістю та зручністю в роботі. PyTorch підтримує динамічну обчислювальну графіку, що дозволяє легко налаштовувати моделі та експериментувати з ними. У контексті аграрного сектору PyTorch часто застосовується для задач прогнозування, таких як передбачення врожайності на основі кліматичних даних.

Для задач, які потребують меншої обчислювальної потужності, але високої точності, можна використовувати Scikit-learn. Ця бібліотека ідеально підходить для класичних ML-алгоритмів, таких як лінійна регресія, дерева рішень, метод опорних векторів. Scikit-learn відома простотою використання та великою кількістю вбудованих функцій для попередньої обробки даних, що є ключовим етапом для роботи з аграрними наборами даних.

Для роботи з великими обсягами даних або складними завданнями оптимізації активно використовуються XGBoost та LightGBM. Ці інструменти базуються на методі бустингу та відомі своєю швидкістю та ефективністю. Наприклад, вони можуть бути використані для прогнозування споживання ресурсів, таких як вода чи паливо, з урахуванням багатьох змінних.

Крім того, варто звернути увагу на спеціалізовані бібліотеки, такі як GeoPandas, які дозволяють працювати з просторовими даними, та Statsmodels для статистичного аналізу. Їх застосування є особливо актуальним у випадках, коли необхідно враховувати географічні особливості або виконувати глибокий аналіз даних.

Як бачимо, широкий вибір платформ і бібліотек дозволяє адаптувати програмне забезпечення під конкретні потреби аграрного сектору. Важливим фактором вибору є баланс між обчислювальною ефективністю, простотою використання та специфічними вимогами до аналізу даних.

Для реалізації методів машинного навчання вибір алгоритмічної мови програмування є одним із найважливіших етапів. Від цього залежить ефективність розробки, продуктивність моделей та можливість інтеграції

рішень у реальні системи. У сфері машинного навчання переважно використовуються такі мови: Python, R та Julia.

Python є найбільш популярною мовою для роботи з ML-алгоритмами завдяки своїй простоті, універсальності та величезній екосистемі бібліотек. Серед основних переваг Python можна назвати наступні.

Багатий набір бібліотек, а саме TensorFlow, PyTorch, Scikit-learn, pandas, NumPy тощо. Вони дозволяють швидко створювати та тестувати моделі, працювати з даними будь-якої складності.

Python легко інтегрується з іншими мовами, базами даних та хмарними платформами. Це робить його ідеальним для розгортання ML-рішень у виробничих системах.

Велика кількість документації, навчальних матеріалів і активна підтримка розробників значно спрощують процес вирішення технічних проблем.

R — мова програмування, яка історично використовувалася для статистичного аналізу. Вона є потужним інструментом для обробки даних та візуалізації результатів. Головною перевагою R є наявність великої кількості спеціалізованих пакетів, таких як caret, mlr, randomForest. Однак у контексті ML, особливо для складних моделей, R поступається Python через меншу продуктивність і меншу кількість фреймворків для роботи з нейронними мережами.

Julia — порівняно нова мова програмування, яка поєднує високу продуктивність із простотою використання. Julia була створена з акцентом на наукові обчислення та машинне навчання. Її основною перевагою є швидкість виконання, що особливо важливо при роботі з великими наборами даних або складними моделями. Проте Julia має менш розвинену екосистему у порівнянні з Python, що може обмежувати її використання в аграрному секторі.

Для задач аграрного сектору найкращим вибором є Python. Ця мова забезпечує не лише потужність і гнучкість, але й має велику кількість спеціалізованих бібліотек, які можна застосовувати для обробки геоданих, аналізу IoT-даних та побудови моделей прогнозування. Наприклад, бібліотека

GeoPandas дозволяє працювати з картографічними даними, що важливо для планування сільськогосподарських робіт.

Тому зупиняємось на алгоритмічній мові Python, що є оптимальним вибором для реалізації методів машинного навчання в аграрному секторі завдяки своїй продуктивності, універсальності та підтримці широкого спектра інструментів для роботи з даними.

Ефективна розробка ML-рішень у аграрному секторі залежить також від правильно обраного середовища розробки та інструментів автоматизації. Ці компоненти забезпечують зручність написання, налагодження, тестування коду та спрощують процес розгортання моделей у виробничих умовах.

Найбільш популярними середовищами розробки для машинного навчання є Jupyter Notebook, PyCharm, VS Code та Google Colab.

Jupyter Notebook є універсальним інструментом для інтерактивної роботи з кодом, що дозволяє писати, виконувати та документувати код у реальному часі. Його основними перевагами є:

- інтеграція з Python і бібліотеками для ML, такими як TensorFlow та Scikit-learn;
- зручність у візуалізації даних завдяки підтримці таких бібліотек, як Matplotlib і Seaborn;
- можливість створення нотаток із кодом і графіками, що спрощує документування процесу розробки.

Для великих проєктів, де потрібно забезпечити чітку структуру та інтеграцію з іншими інструментами, рекомендовано використовувати PyCharm або VS Code. PyCharm пропонує зручний редактор коду, автоматичне завершення, підтримку віртуальних середовищ (virtual environments) і вбудовані інструменти для тестування. VS Code, своєю чергою, вирізняється широким набором розширень, які дозволяють працювати з бібліотеками, базами даних та платформами хмарного обчислення.

Якщо доступ до потужних обчислювальних ресурсів обмежений, варто скористатися Google Colab. [28] Ця платформа забезпечує безкоштовний доступ

до GPU та TPU, дозволяючи запускати складні моделі без локальної апаратної підтримки. Colab також легко інтегрується з Google Drive, що спрощує зберігання та обмін результатами.

Автоматизація процесів розробки та розгортання моделей є ще одним важливим аспектом. У цьому контексті часто використовуються інструменти, такі як Docker, Kubernetes, MLflow та Airflow.

Docker дозволяє створювати ізольовані контейнери, що включають усе необхідне для роботи моделі — бібліотеки, середовища виконання, налаштування. Це спрощує розгортання моделей у різних середовищах.

MLflow забезпечує відстеження експериментів, управління моделями та автоматизацію їхнього розгортання. Це особливо корисно для роботи з великими командами або при розробці кількох моделей одночасно.

Apache Airflow використовується для організації автоматичних робочих процесів, наприклад, регулярного оновлення моделей на основі нових даних. Цей інструмент дозволяє налаштовувати складні пайплайни, які включають етапи збору, очищення та аналізу даних.

Таким чином, вибір середовища розробки та інструментів автоматизації залежить від специфіки завдань і доступних ресурсів. Для невеликих проєктів таких як дана система, цілком достатньо Jupyter Notebook або Google Colab.

## **Висновки до розділу**

В результаті виконання даного розділу вирішені такі питання:

- розглянуті методи машинного навчання, їх особливості, переваги та недоліки;
- описані підходи та методи оптимізації моделей машинного навчання для задачі оцінки транспортних засобів;
- зроблено огляд програмних засобів реалізації системи та обраний відповідний інструментарій.

## РОЗДІЛ 3

### РЕЗУЛЬТАТИ ВИРІШЕННЯ ЗАДАЧІ

#### 3.1. Підключення до API сервісу оголошень

В якості джерела даних було обрано популярну дошку оголошень по продажу автомобілів - [auto.gia.com](http://auto.gia.com). Цей сайт - один з декількох дошок платформи [gia.com](http://gia.com), яка надає розробникам можливість взаємодіяти з базами даних через програмний інтерфейс (API). Сайт цього сервісу має назву [developers.gia.com](http://developers.gia.com). Опишемо його деякі можливості та порядок реєстрації.

API сайту [developers.gia.com](http://developers.gia.com) [29] надає широкий спектр можливостей для розробників, які прагнуть інтегрувати сервіси популярної платформи RIA у свої проекти. Цей інтерфейс прикладного програмування забезпечує доступ до великої бази даних автомобілів, нерухомості та інших категорій оголошень, пропонуючи інструменти для ефективного використання цих даних у веб-додатках, мобільних застосунках або корпоративних системах.

Однією з головних переваг API є його багатофункціональність. Розробники отримують можливість здійснювати пошук оголошень, фільтрувати дані за різними параметрами, переглядати детальну інформацію про об'єкти, отримувати фотографії, а також працювати з історією зміни цін або статусу оголошень. Це дозволяє створювати продукти, які можуть не тільки відображати актуальну інформацію, а й надавати аналітику, прогнози та персоналізовані рекомендації.

API розроблений з урахуванням сучасних стандартів безпеки та продуктивності. Користувачі можуть бути впевнені в тому, що доступ до даних є захищеним, а функціонал залишається стабільним навіть під час обробки великого обсягу запитів. Завдяки оптимізованій структурі запитів API забезпечує високу швидкість роботи, що є критично важливим для систем, які залежать від реального часу, наприклад, платформ моніторингу або торгових ботів.



Однією з ключових переваг API developers.ria.com є гнучкість у налаштуванні запитів. Сервіс дозволяє розробникам самостійно обирати необхідні параметри, що знижує обсяг даних, які передаються між сервером і клієнтом, тим самим зменшуючи навантаження на систему. Це особливо важливо для мобільних додатків, де економія трафіку та ресурсів пристрою грає важливу роль.

API також пропонує можливість інтеграції сторонніх сервісів та платформ, що розширює його застосування у бізнес-процесах. Наприклад, доступ до бази даних автомобілів може бути корисним для платформ страхування, оцінки вартості транспортних засобів чи сервісів підбору автозапчастин. Водночас функціонал, пов'язаний із нерухомістю, відкриває нові горизонти для агентств, які прагнуть автоматизувати свою роботу, надаючи клієнтам зручні інструменти для пошуку житла.

Розробники мають доступ до детальної документації, що значно полегшує процес інтеграції. Вона включає описи методів API, параметри запитів, приклади використання та можливі варіанти обробки помилок. Це знижує бар'єр входу для новачків і дає змогу швидко адаптуватися до роботи навіть без великого досвіду роботи з API.

Іншою перевагою є постійне оновлення функціоналу API, яке забезпечує актуальність і відповідність сучасним потребам. Платформа підтримує зворотний зв'язок із користувачами, що дозволяє враховувати їхні побажання під час впровадження нових можливостей. Такий підхід гарантує, що API залишається конкурентоспроможним і затребуваним серед розробників різного рівня.

Таким чином, API developers.ria.com є потужним інструментом для створення додатків, які використовують дані платформи RIA. Його можливості дозволяють розробникам легко отримувати доступ до великого масиву даних, забезпечуючи їхню інтеграцію у різні бізнес-процеси, сервіси та програми. Завдяки своїй гнучкості, швидкодії та зручності використання цей інтерфейс відкриває широкі перспективи для реалізації інноваційних ідей у сфері IT.

Щоб почати роботу з API, необхідно пройти простий процес реєстрації, який забезпечить доступ до функціоналу та дозволить використовувати всі переваги інтерфейсу.

Перш за все, потрібно створити обліковий запис на офіційному сайті платформи. Для цього необхідно перейти на сторінку [developers.gia.com](https://developers.gia.com) і натиснути кнопку реєстрації. Користувачу буде запропоновано заповнити стандартну форму, вказавши ім'я, електронну пошту та пароль. Після успішного заповнення форми та підтвердження електронної пошти буде створено обліковий запис.

Далі слід авторизуватись в акаунті, щоб отримати доступ до персонального кабінету розробника. У кабінеті необхідно створити проєкт, який використовуватиме API. Під час створення проєкту вказуються основні дані, зокрема назва, опис і сфера застосування. На цьому етапі система генерує унікальний API-ключ, який слугуватиме ідентифікатором проєкту та забезпечуватиме доступ до функціоналу API.

API-ключ — це важливий елемент безпеки, тому його слід зберігати в конфіденційності. Цей ключ використовується в кожному запиті до API для авторизації. Щоб протестувати ключ і переконатись у його працездатності, можна скористатися інструментами тестування запитів, доступними в документації платформи. Там же наведено приклади використання ключа в різних сценаріях.

Якщо у вас виникають питання або труднощі на будь-якому етапі реєстрації чи налаштування, служба підтримки платформи готова надати допомогу. Вона доступна через контактну форму на сайті або електронну пошту. Крім того, у розділі документації міститься розгорнутий опис поширених проблем і способів їх вирішення, що значно полегшує процес підключення.

Автор виконав всі вимоги по реєстрації та отримав API-ключ та доступ до особистого кабінету. Відповідний скрін-шот наведено на рис.3.1.

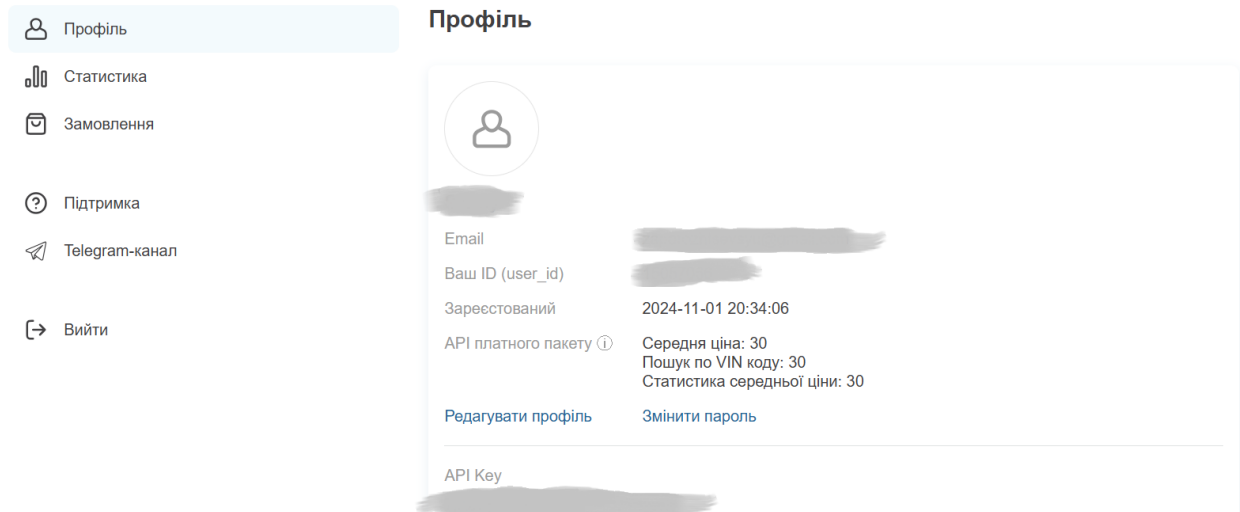


Рисунок 3.1 - Особистий кабінет розробника developers.ria.com

Таким чином, зареєструвавшись на платформі та отримавши API-ключ, ми можемо почати інтеграцію даних RIA у свій проєкт, використовуючи всі переваги API. Цей процес займає мінімум часу, що дозволяє розробникам швидко перейти до практичного застосування інструментів платформи для створення ефективних і сучасних рішень.

### 3.2. Завантаження та попередня обробка вхідних даних

Почнемо створювати програму. Спочатку треба завантажити основні бібліотеки для роботи з даними та файлами. Сюди входять як стандартні `pandas`, `numpy` та деякі з `sklearn`, так і бібліотеки для роботи з форматами `json`, обміну по запитам `requests`, а також всілякі додаткові (рис.3.2).

Надаємо змінній `api_key` значення API-ключа з особистого кабінету (рис.3.3). Це позбавить нас необхідності кожного разу копіювати цю послідовність символів для створення запитів. Кожного разу, коли треба звернутись до API - використовуємо цю змінну.

```

import os, pandas as pd, numpy as np
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
import seaborn as sns
import json
import requests
from google.colab import files
import time
import random
import matplotlib.pyplot as plt
import warnings
warnings.filterwarnings('ignore')

```

Рисунок 3.2 - Імпорт основних бібліотек

```

api_key = [REDACTED] # Register and get your API key from https://developers.ria.com/

```

Рисунок 3.3 - Зафіксоване значення API-ключа

В процесі перевірки роботи програмного інтерфейсу отримано результат виконання запиту у форматі JSON (рис.3.4). Це дозволяє зручно аналізувати дані та перевіряти коректність відповіді API, забезпечуючи інтеграцію з іншими системами.

```

url = 'https://developers.ria.com/auto/categories/?api_key=' + api_key
j_categories = json.loads(requests.get(url).text)
print(j_categories)

```

```

[{'name': 'Легкові', 'value': 1}, {'name': 'Мото', 'value': 2}, {'name': 'Водний транспорт', 'value': 3}, {'na

```

Рисунок 3.4 - Перевірка роботи API

Бачимо список категорій транспортних засобів, з якого можемо обрати необхідну нам. Аналогічно можна викликати каталоги різних ознак - наприклад, марки, моделі, тип кузова, тип пального, тип коробки передач і т.д. Це показано на рис. 3.5. В реальній роботі додатка вибір відповідних номерів з каталогу користувач буде робити за допомогою спливаючих меню-списків, використовуючи віконний інтерфейс користувача.

```

# Вибір параметрів автомобіля
# Категорія
url = 'https://developers.ria.com/auto/categories/?api_key=' + api_key
j_categories = json.loads(requests.get(url).text)
print(j_categories)
category = str(1) # simulation selection 'Легкові'
# Марка
url = 'https://developers.ria.com/auto/categories/' + category + '/marks?api_key=' + api_key
j_mark = json.loads(requests.get(url).text)
print(j_mark)
mark = str(84) # simulation selection 'Volkswagen'
# Модель
url = 'https://developers.ria.com/auto/categories/' + category + '/marks/' + mark + '/models?api_key=' + api_key
j_model = json.loads(requests.get(url).text)
print(j_model)
model = str(38063) # simulation selection 'Passat B7'
# Тип кузова
url = 'https://developers.ria.com/auto/categories/' + category + '/bodystyles?api_key=' + api_key
j_bodystyles = json.loads(requests.get(url).text)
print(j_bodystyles)
bodystyle = str(3) # simulation selection 'Седан'

# Додаткові параметри, які не приймають участь в пошуку та вибірці, але потрібні для моделювання
# Тип пального
url = 'https://developers.ria.com/auto/type?api_key=' + api_key
j_type = json.loads(requests.get(url).text)
print(j_type)
type = str(1) # simulation selection 'Бензин'
# Тип коробки передач
url = 'https://developers.ria.com/auto/categories/' + category + '/gearboxes?api_key=' + api_key
j_gearboxes = json.loads(requests.get(url).text)
print(j_gearboxes)
gearboxe = str(1) # simulation selection 'Ручна / Механіка'

[{'name': 'Легкові', 'value': 1}, {'name': 'Мото', 'value': 2}, {'name': 'Водний транспорт', 'value': 3}, {'name':
[{'name': 'Abarth', 'value': 5166}, {'name': 'Acura', 'value': 98}, {'name': 'Adler', 'value': 2396}, {'name': 'A
[{'name': 'Amarok', 'value': 31575}, {'name': 'Apollo', 'value': 59688}, {'name': 'Arteon', 'value': 51838}, {'na
[{'name': 'Седан', 'value': 3}, {'name': 'Кросовер', 'value': 5}, {'name': 'Мінівен', 'value': 8}, {'name': 'Варі
[{'name': 'Бензин', 'value': 1}, {'name': 'Газ', 'value': 3}, {'name': 'Газ пропан-бутан / Бензин', 'value': 4},
[{'name': 'Ручна / Механіка', 'value': 1}, {'name': 'Автомат', 'value': 2}, {'name': 'Типтронік', 'value': 3}, {'

```

Рисунок 3.5 - Каталоги бази даних та вибір параметрів автомобілів

Створюємо запит до бази даних, щоб отримати перелік активних оголошень про продаж автомобілів з параметрами, що відповідають зазначеним критеріям.

Приклад API-запиту може виглядати так, як показано на рис. 3.6.

Бачимо, що в результаті роботи запиту маємо список номерів активних оголошень, по яким вже можна отримувати безпосередню інформацію про самі оголошення.

```

▶ url = 'https://developers.ria.com/auto/search?api_key=' + api_key + (
    '&category_id=' + category +
    '&marka_id=' + mark +
    '&model_id=' + model +
    '&countpage=100&page=0'
)
r = requests.get(url)
r.text

⇌ '{"additional_params":{"lang_id":2,"page":"0","view_type_id":0,"target":"search'
9032951","29033500","29187454","29146966","29125409","29001774","29190403","2894
4","29160787","28657138","28588598","29004757","29181519","29181001","29180640".
12845","29104999","29034461","28911178","29154782","25658763","29020477","28818
2","29163431","28264986","29001609","29181794","29171728","29165106","29117574".
71765","29150038","29116349","29099674","28812635","29040389","26841911","29010:

```

Рисунок 3.6 - Запит на список номерів активних оголошень

Приклад коду для зчитування оголошень по АРІ наведено на рис. 3.7. Код враховує обмеження на кількість запитів (на безкоштовному акаунті RIA обмеження складає 1000 запитів).

```

▶ json2 = json.loads(r.text)
print(json2['result']['search_result']['count'])
ids = json2['result']['search_result']['ids']
print (len(json2['result']['search_result']['ids']))
if json2['result']['search_result']['count'] > 100 :
    for i in range(1, json2['result']['search_result']['count'] // 100 + 1) :
        url = 'https://developers.ria.com/auto/search?api_key=' + api_key + (
            '&category_id=' + category +
            '&marka_id=' + mark +
            '&model_id=' + model +
            '&countpage=100&page=' + str(i)
        )
        r = requests.get(url)
        json2 = json.loads(r.text)
        print (len(json2['result']['search_result']['ids']))
        ids += json2['result']['search_result']['ids']
len(ids)

⇌ 1871

```

Рисунок 3.7 - Блочне зчитування номерів оголошень по АРІ

Бачимо, що кількість оголошень, які відповідають нашим фільтрам, складає 1871, що набагато більше, ніж існуючі обмеження. Тому обмежимося кількістю трохи меншою, ніж 1000. Виберемо з цілого списку випадковим чином 990 оголошень (код наведено на рис.3.8).

```

v = 990
if len(ids) > v :
    ids_temp = ids.copy()
    ids_model = []
    while v > 0 :
        rnd_choice = random.choice(ids_temp)
        ids_temp.remove(rnd_choice)
        ids_model.append(rnd_choice)
        v -= 1
ids_model

```

```

['28754729',
'29078992',
'26865829',
'20101201']

```

Рисунок 3.8 - Випадковий вибір оголошень для виконання лімітів

На наступному етапі здійснюється зчитування даних і їх конвертація у потрібний формат. Оскільки частина інформації в оголошеннях може бути відсутня, для кожного поля використовується оператор try. У разі відсутності значення в полі записується 'undefined'. Код показаний на рис.3.9.

```

list_for_df = []
for id_ob in range(len(ids_model)) :
    url = 'https://developers.ria.com/auto/info?api_key=' + api_key + '&auto_id=' + ids_model[id_ob]
    ob = requests.get(url)
    json_object = json.loads(ob.text)
    new_dict = {}
    try:
        new_dict.update({'active':json_object['autoData']['active']})
    except:
        new_dict.update({'active':'undefined'})
    try:
        new_dict.update({'bodyId':str(json_object['autoData']['bodyId'])}) # Тип кузова
    except:
        new_dict.update({'bodyId':'undefined'})
    try:
        new_dict.update({'fuelId':str(json_object['autoData']['fuelId'])}) # Тип пального
    except:
        new_dict.update({'fuelId':'undefined'})
    try:
        new_dict.update({'gearBoxId':str(json_object['autoData']['gearBoxId'])}) # Тип трансмісії
    except:
        new_dict.update({'gearBoxId':'undefined'})
    try:
        new_dict.update({'raceInt':json_object['autoData']['raceInt']}) # Пробіг в тис.км
    except:
        new_dict.update({'raceInt':'undefined'})
    try:
        new_dict.update({'age':int(time.time()[-4:])-json_object['autoData']['year']}) # Вік (різниця років)
    except:
        new_dict.update({'age':'undefined'})
    try:
        new_dict.update(json_object['autoInfoBar'])
    except:
        new_dict.update({'custom':'undefined', 'abroad':'undefined', 'onRepairParts':'undefined',
        'damage':'undefined', 'underCredit':'undefined', 'confiscatedCar':'undefined'})
    try:
        new_dict.update({'USD': json_object['USD']})
    except:
        new_dict.update({'USD':'undefined'})
    list_for_df.append(new_dict)
df_model = pd.DataFrame(list_for_df)

```

Рисунок 3.9 - Зчитування і конвертація даних

Якщо дані не конвертувати, ми отримаємо їх у вигляді, подібному до наведеного прикладу на рис.3.10.

```

▶ j['autoData']
↔ {'active': True,
  'autoId': 28755676,
  'bodyId': 3,
  'categoryId': 1,
  'categoryNameEng': 'legkovi',
  'custom': 0,
  'description': 'Продаю власне авто Vw Jetta 2015 року (рестайлінг), привозив під себе лютий 2020 року, авто у відмінному робочому стані, за не повний рік проїхав 22 тисяч км, нещодавно провів ТО, у комплекті нова зимова гума BARUM, новий акумулятор Bosh, Причина продажу: потрібні гроші, деталі по телефону',
  'driveId': 2,
  'driveName': 'Передній',
  'fromArchive': False,
  'fuelId': 2,
  'fuelName': 'Дизель, 2 л.',
  'fuelNameEng': 'dizel',
  'gearBoxId': 2,
  'gearboxName': 'Автомат',
  'isSold': False,
  'mainCurrency': 'USD',
  'onModeration': False,
  'race': '78 тис. км',
  'raceInt': 78,
  'statusId': 0,
  'subCategoryNameEng': 'sedan',
  'version': '',
  'withVideo': False,
  'withVideoMessages': False,
  'year': 2015}

```

Рисунок 3.10 - Приклад даних випадкового оголошення

Вміст датасета після конвертації наведено на рис.3.11.

	active	bodyId	fuelId	gearBoxId	raceInt	age	custom	abroad	onRepairParts	damage	underCredit	confiscatedCar	USD
0	True	3	1	2	273	9	False	False	False	False	False	False	10200
1	True	2	2	2	350	10	False	False	False	False	False	False	10300
2	True	3	1	2	135	8	False	False	False	False	False	False	10700
3	True	3	4	2	175	9	False	False	False	False	False	False	10500
4	True	3	1	2	132	8	False	False	False	False	False	False	12300
...	...	...	...	...	...	...	...	...	...	...	...	...	...
985	True	3	1	2	182	7	False	False	False	True	False	False	11500
986	True	3	1	2	98	7	False	False	False	False	False	False	11450
987	True	2	2	2	260	10	False	False	False	False	False	False	10700
988	True	3	0	2	128	8	False	False	False	False	False	False	11200
989	True	3	2	2	154	7	False	False	False	False	False	False	12750

990 rows × 13 columns

Рисунок 3.11 - Датасет для моделювання



Відбираємо лише активні оголошення, видаляємо зайві ознаки, що не використовуватимуться для навчання, та конвертуємо текстові значення у стовпці 'raceInt' (Пробіг) у числовий формат (рис. 3.12). Маємо в результаті 979 записів.

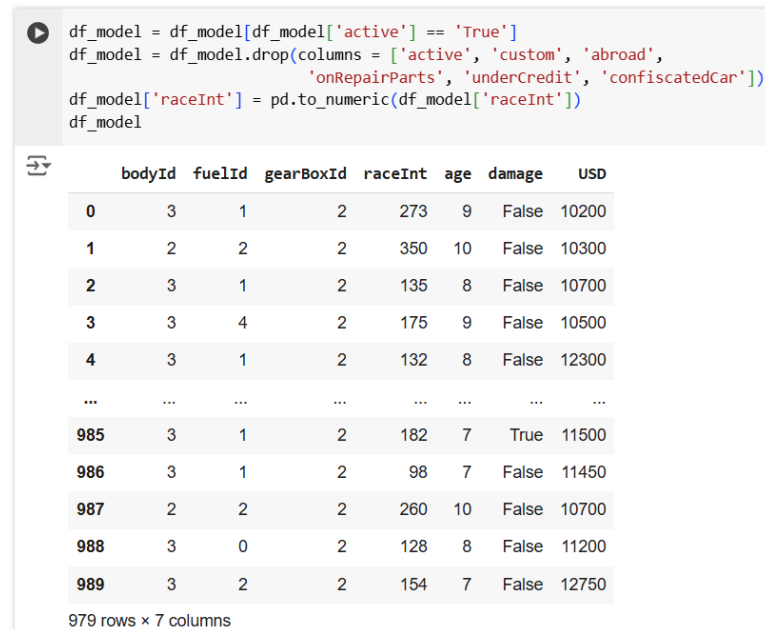


Рисунок 3.12 - Підготовка попереднього набору даних

Скористаємось методом `distplot` бібліотеки `seaborn` та візуально проаналізуємо розподіл цін для обраних автомобілів (рис.3.13).

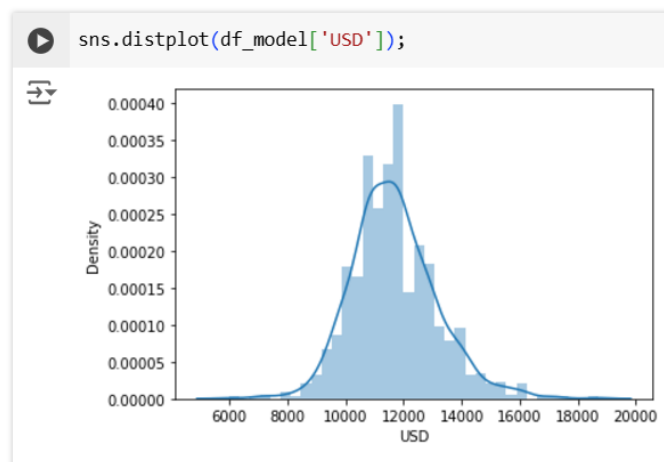


Рисунок 3.13 - Розподіл цін на вибрані автомобілі

Подивимось також, чи є викиди в цінах, щоб прийняти рішення по можливому уточненню датасета. Для цього використовуємо метод `boxplot`. Як бачимо на рис. 3.14 - деякі викиди мають місце.

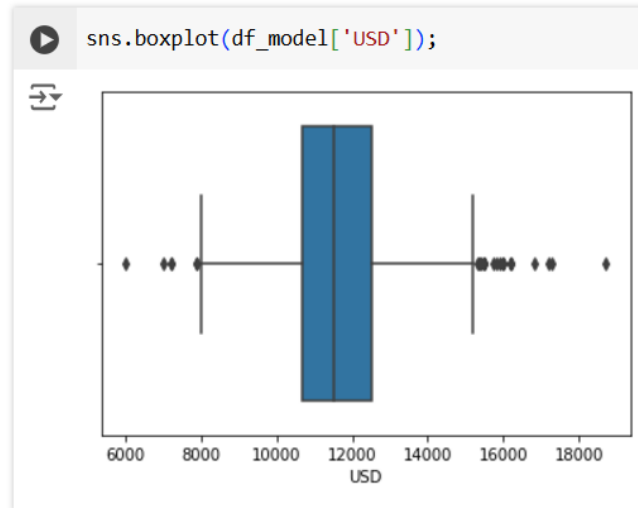


Рисунок 3.14 - Візуалізація викидів

Обираємо межі цін від 7000 до 17000, тим самим позбавивши датасет відвертих викидів (рис. 3.15).

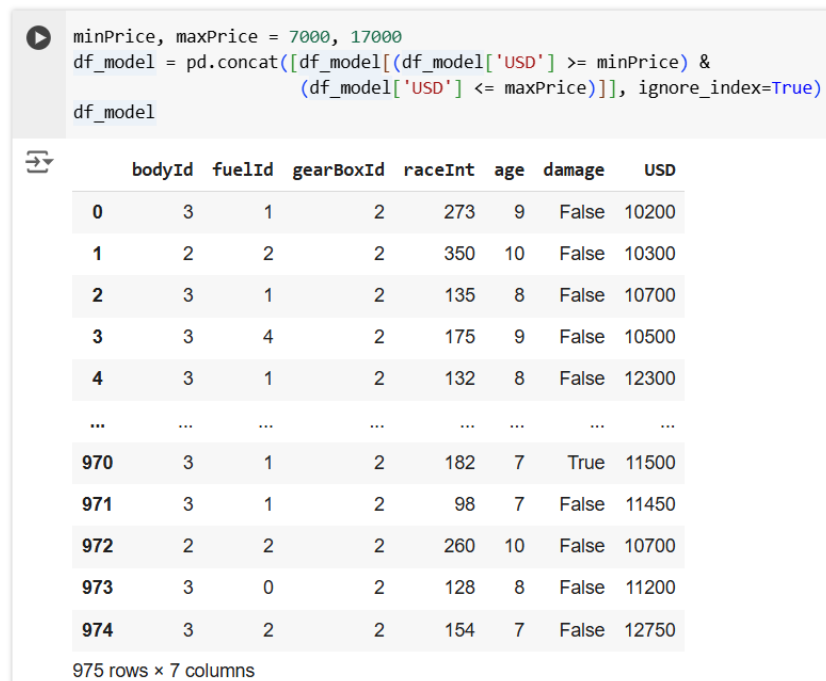


Рисунок 3.15 - Датасет без викидів ціни

Наступний крок - позбавлення категоріальних параметрів. Справа в тому, що в стовпцях `bodyId`, `fuelId`, `gearBoxId` та `damage` ми маємо не числа, а категорії, хоча на вигляд може здаватись навпаки. Використовуємо метод `get_dummies`, результат роботи якого наведений на рис. 3.16.

```
df_dum = pd.get_dummies(df_model)
df_dum = pd.concat([df_dum], ignore_index = True)
df_dum
```

	raceInt	age	USD	bodyId_2	bodyId_3	bodyId_4	bodyId_5	fuelId_0	fuelId_1	fuelId_2	fuelId_4	fuelId_8	gearBoxId_1	gearB
0	273	9	10200	0	1	0	0	0	1	0	0	0	0	
1	350	10	10300	1	0	0	0	0	0	1	0	0	0	
2	135	8	10700	0	1	0	0	0	1	0	0	0	0	
3	175	9	10500	0	1	0	0	0	0	0	1	0	0	
4	132	8	12300	0	1	0	0	0	1	0	0	0	0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
970	182	7	11500	0	1	0	0	0	1	0	0	0	0	
971	98	7	11450	0	1	0	0	0	1	0	0	0	0	
972	260	10	10700	1	0	0	0	0	0	1	0	0	0	
973	128	8	11200	0	1	0	0	1	0	0	0	0	0	
974	154	7	12750	0	1	0	0	0	0	1	0	0	0	

975 rows x 19 columns

Рисунок 3.16 - Остаточний набір даних для навчання

Таким чином, ми отримали остаточний набір даних для навчання, який складається з 975 об'єктів, у кожного по 19 параметрів (18 ознак та 1 цільова змінна). Тепер можна переходити до реалізації та навчання моделі.

### 3.3. Реалізація та оптимізація моделі машинного навчання

Спочатку відокремлюємо цільову змінну від ознак і розподіляємо дані на тренувальну та тестову вибірки (рис. 3.17).

```
# Відділимо цільову змінну
y = df_dum["USD"]
X = df_dum.drop(columns=["USD"])

# Розіб'ємо виборку на тренувальну та тестову
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.15, random_state=10)
```

Рисунок 3.17 - Виділення цільової змінної та розбивка вибірки

Зробимо невеличку візуальну оцінку впливу різних ознак на цільову змінну (рис. 3.18).

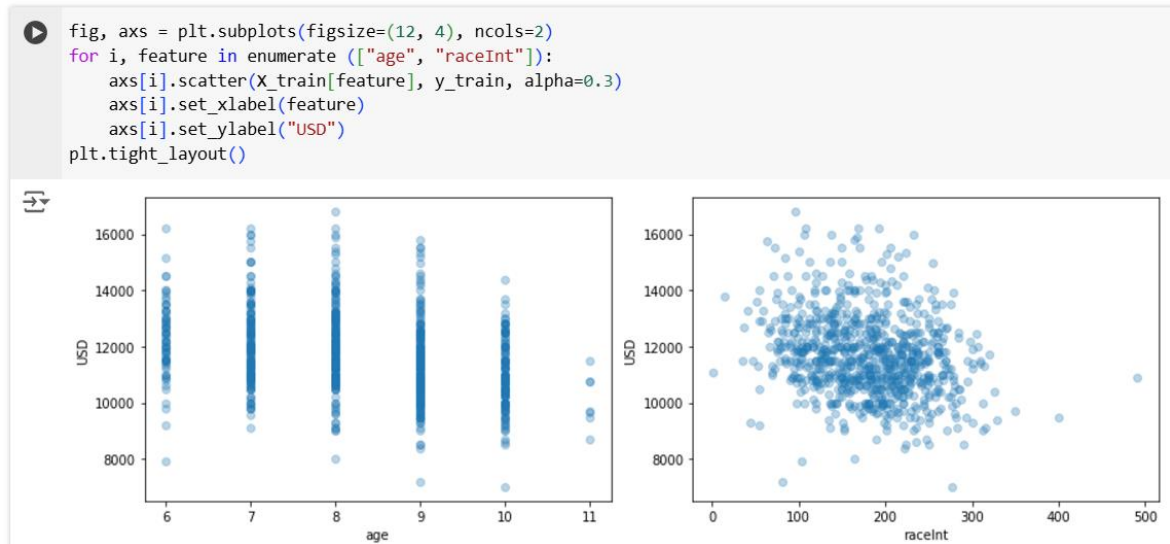


Рисунок 3.18 - Візуальна оцінка впливу

Як можна побачити, зі зростанням віку автомобіля або його пробігу - середня ціна закономірно знижується.

Спершу обчислимо середню ціну та визначимо середні показники помилки, якщо використовувати це значення як прогноз для всіх даних (рис.3.19). Будемо використовувати цей показник для порівняння та встановлення степені якості навчених моделей.

```

# Constant prediction
from sklearn import metrics

mean_y = np.repeat(np.mean(y), y.shape[0])
constant_error = metrics.mean_squared_error(y, mean_y)
constant_error, constant_error**0.5

```

(2012003.59482025, 1418.4511252842835)

Рисунок 3.19 - Середні показники помилки для стаціонарної середньої ціни

Проводимо перше навчання моделі та оцінюємо її результати (рис.3.20).

```

from sklearn.linear_model import Ridge

model = Ridge()
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
y_train_pred = model.predict(X_train)

print("Test RMSE = %.4f" % rmse(y_test, y_pred))
print("Train RMSE = %.4f" % rmse(y_train, y_train_pred))

```

Test RMSE = 1130.6798  
Train RMSE = 1123.0950

Рисунок 3.20 - Перше навчання моделі

Знаходимо оптимальні гіперпараметри через GridSearchCV (рис. 3.21).

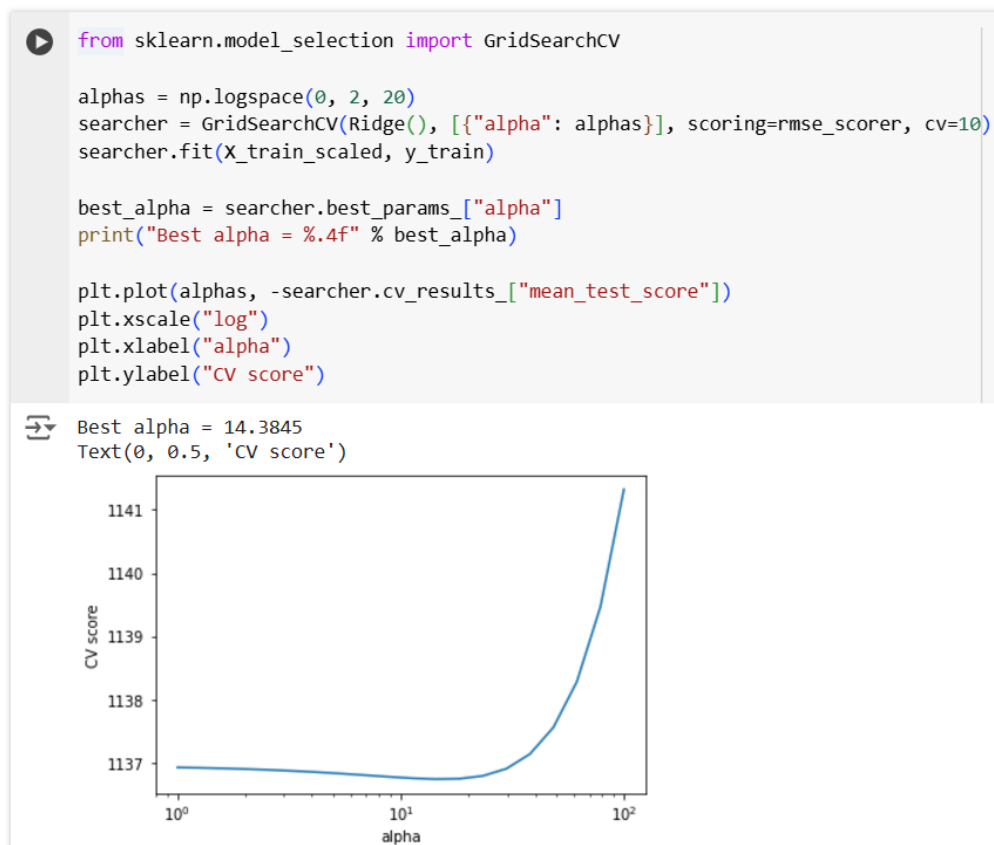


Рисунок 3.21 - Пошук оптимальних гіперпараметрів моделі

Спробуємо навчити модель з оптимальними гіперпараметрами (рис.3.22).

```

▶ from sklearn.pipeline import Pipeline

simple_pipeline = Pipeline([
    ('scaling', StandardScaler()),
    ('regression', Ridge(best_alpha))
])

model = simple_pipeline.fit(X_train, y_train)
y_pred = model.predict(X_test)
print("Test RMSE = %.4f" % rmse(y_test, y_pred))

```

Test RMSE = 1131.5634

Рисунок 3.22 - Навчання з оптимальними гіперпараметрами

І нарешті, спробуємо регресійні моделі з іншими підходами (наприклад, Lasso). Як бачимо (рис.3.23), виявилось, що модель типу Lasso показала найкращий результат.

```

▶ from sklearn.linear_model import (
    Ridge,
    Lasso,
    LinearRegression,
)
from sklearn.metrics import mean_squared_error

model = Lasso()
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
y_train_pred = model.predict(X_train)

print("Test MSE = %.4f" % mean_squared_error(y_test, y_pred))
print("Train MSE = %.4f" % mean_squared_error(y_train, y_train_pred))
np.sqrt(mean_squared_error(y_test, y_pred))

```

Test MSE = 1220398.5185  
Train MSE = 1276468.4479  
1104.7164878528101

Рисунок 3.23 - Найкращий результат регресійних моделей

Якщо брати якість моделювання в порівнянні зі стаціонарною середньою помилкою, то ми змогли зменшити середню помилку більше, ніж на 20%, що є досить непоганим результатом.

### 3.4. Розробка елементів інтерфейсу користувача

Графічний інтерфейс користувача (GUI) є одним із найважливіших елементів будь-якого програмного забезпечення, оскільки забезпечує ефективну та інтуїтивну взаємодію користувача із системою. У сучасному світі інтерфейси є не лише засобом управління додатком, а й важливим фактором, що визначає рівень задоволеності користувача продуктом. Інтерфейси дозволяють абстрагуватися від складності реалізації функціоналу й зосередитися на вирішенні завдань за допомогою простих і зрозумілих візуальних елементів.

При розробці GUI для різних платформ слід враховувати специфіку кожного типу додатків. Наприклад, віконні інтерфейси для настільних комп'ютерів повинні бути оптимізовані для клавіатури та миші, забезпечуючи велику площу екрану для відображення складного функціоналу. Натомість веб-додатки, хоча й можуть працювати на десктопах, часто повинні враховувати можливість доступу з мобільних пристроїв. Це передбачає адаптивний дизайн, інтеграцію сенсорного управління та обмеження у використанні ресурсів.

Розробка віконних додатків на Python пропонує широкий спектр бібліотек. Tkinter, як стандартна бібліотека, чудово підходить для створення базових інтерфейсів, однак має обмеження в дизайні. PyQt або PySide пропонують більший набір інструментів для розробки професійних програм із підтримкою складного функціоналу. Kivy є універсальним вибором для проєктів, орієнтованих на мобільні пристрої, оскільки підтримує мультиплатформеність та роботу з сенсорними екранами.

Веб-додатки, зі свого боку, пропонують гнучкість і доступність завдяки тому, що працюють через браузер. Вони не вимагають встановлення додаткового програмного забезпечення, а їх оновлення відбувається на сервері, що значно спрощує підтримку. Python забезпечує потужний стек інструментів для розробки таких додатків. Django підходить для створення великих і

комплексних проєктів, тоді як Flask дозволяє швидко розробляти легкі веб-рішення.

Основною відмінністю між веб-додатками та настільними програмами є спосіб взаємодії з користувачем. Веб-додатки переважно орієнтовані на мінімалізм, швидкість завантаження та адаптивність, тоді як настільні додатки можуть включати складніші елементи керування, потужний графічний інтерфейс і розширену функціональність. Наприклад, віконний інтерфейс може використовувати багато вкладених меню, багатовіконну структуру та інші елементи, які у веб-додатках можуть виглядати перевантаженими.

Особливості розробки інтерфейсів для веб-додатків також включають потребу інтеграції з різними браузерами та забезпечення коректного відображення на різних екранах. Розробники часто стикаються з необхідністю підтримки швидкої взаємодії між клієнтською та серверною частинами, що вимагає оптимізації передачі даних і використання сучасних технологій, таких як REST API або WebSocket. Натомість у настільних додатках головною проблемою може бути розподіл ресурсів комп'ютера та забезпечення стабільної роботи навіть на старих системах.

Порівнюючи віконні інтерфейси та веб-додатки, можна відзначити, що перші пропонують більшу продуктивність і доступ до апаратних ресурсів, тоді як другі забезпечують зручність доступу і кращу інтеграцію з хмарними сервісами. Для вибору підходу до розробки слід враховувати вимоги до продуктивності, аудиторію продукту, необхідність у багатоплатформеності та специфіку функціоналу.

Python, завдяки своїй універсальності, дозволяє об'єднувати найкращі практики як для веб-додатків, так і для віконних програм. Його багатий вибір бібліотек, потужні засоби інтеграції з іншими мовами та фреймворками, а також велика кількість спільнот розробників роблять Python ідеальним вибором для створення сучасних інтерфейсів користувача.

Python пропонує широкий вибір бібліотек для створення графічних інтерфейсів користувача (GUI), які дозволяють реалізувати як прості, так і



складні програми для різних платформ. Серед найпопулярніших інструментів — Tkinter, PyQt/PySide, Kivy та wxPython, кожна з яких має свої переваги і підходить для певних завдань.

Tkinter є стандартною бібліотекою Python і встановлюється разом із самим інтерпретатором. Це робить її ідеальним вибором для швидкого створення простих додатків без необхідності встановлення додаткових залежностей. Tkinter забезпечує базовий набір елементів керування, таких як кнопки, поля введення, меню та слайдери, але її дизайн виглядає досить простим і застарілим порівняно з сучасними вимогами.

PyQt та PySide є обгортками для Qt, одного з найпотужніших фреймворків для створення кросплатформних додатків. Вони дозволяють створювати складні, професійні інтерфейси, які виглядають сучасно та мають вбудовану підтримку тем оформлення. Окрім базових елементів GUI, Qt пропонує інструменти для роботи з графікою, мультимедіа, мережевими протоколами та багатовіконними структурами. PyQt та PySide часто використовуються для комерційних проєктів, хоча PySide має перевагу у відкритій ліцензії.

Kivy орієнтований на створення мультимедійних і мобільних додатків. Ця бібліотека підтримує мультиплатформеність і чудово підходить для розробки додатків із сенсорним управлінням, таких як програми для Android та iOS. Kivy забезпечує сучасний дизайн і підтримує апаратне прискорення для графіки, що робить її ефективною для ресурсомістких завдань.

WxPython є ще однією популярною бібліотекою для створення GUI на Python. Вона забезпечує рідний вигляд додатків завдяки використанню нативних інструментів операційної системи. WxPython підходить для створення багатофункціональних настільних додатків, але її документація може здатися менш зручною для новачків порівняно з іншими бібліотеками.

Вибір бібліотеки залежить від завдань проєкту: для простих додатків підходить Tkinter, для професійних і кросплатформних — PyQt або PySide, для мобільних — Kivy, а для додатків із нативним виглядом — wxPython. Python

надає розробникам гнучкість та широкий спектр інструментів, що дозволяє створювати інтерфейси будь-якої складності.

Таким чином, розглянувши можливості різних бібліотек по створенню інтерфейсів користувача, було зроблено вибір на користь стандартної Python-бібліотеки Tkinter.

В роботі було розроблено декілька форм для полегшення роботи користувача з додатком.

Форми є одним із ключових елементів GUI, оскільки вони забезпечують взаємодію між користувачем і програмою. Вони можуть бути простими або складними, залежно від функціональності додатка, і часто містять такі елементи, як текстові поля, кнопки, випадаючі списки, прапорці та перемикачі. Основне завдання форми — зібрати та обробити введені дані, тому її дизайн має бути інтуїтивно зрозумілим і зручним для користувача.

Ефективна форма повинна бути лаконічною, мати чітко організовані елементи та уникати зайвих полів, щоб не перевантажувати користувача. Наприклад, використання зрозумілих підказок або текстових міток допомагає орієнтуватися в тому, які дані очікуються. Також важливим є забезпечення зворотного зв'язку, наприклад, сповіщення про помилки у введенні чи підтвердження успішного відправлення форми.

У веб-додатках форми грають ще важливішу роль завдяки інтерактивним можливостям, які надають сучасні фреймворки, такі як React, Vue або Django Forms у Python. Вони дозволяють створювати динамічні форми з валідацією даних на стороні клієнта, а також підтримують адаптивний дизайн для різних пристроїв. Таким чином, форми залишаються універсальним і невід'ємним інструментом створення інтуїтивного та функціонального користувацького досвіду.

Наведемо приклади розроблених форм. Форма для введення даних при конструюванні запиту до відповідного сервісу можна побачити на рис.3.24. Як бачимо, користувачу відразу пропонується обрати джерело даних з декількох

можливих веб-ресурсів, хоча в даній роботі ми користувались тільки одним з них. Так на майбутнє може бути передбачено розширення функціоналу.

Рисунок 3.24 - Форма введення даних для запиту до сервісу

Як відомо, моделі машинного навчання дуже чутливі до якості даних, з якими вони працюють.

Особливу роль також грає кількість об'єктів, на яких навчаються моделі.

Якщо кількості буде недостатньо - це напряму вплине на якість моделювання.

Тому доцільно попереджати користувача в випадках, коли кількість знайдених об'єктів може виявитись недостатньою.

Форма можливого попередження в такому випадку наведена на рис. 3.25.

Варіант форми для вибору режиму навчання моделі наведений на рис.3.26.

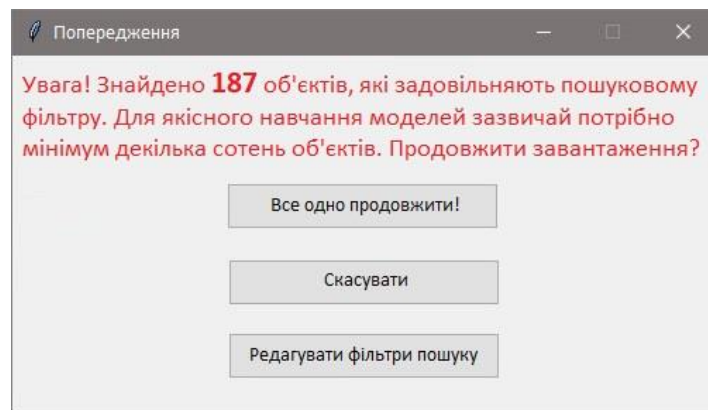


Рисунок 3.25 - Формат попередження про можливі проблеми при навчанні через недостатню кількість об'єктів вибірки

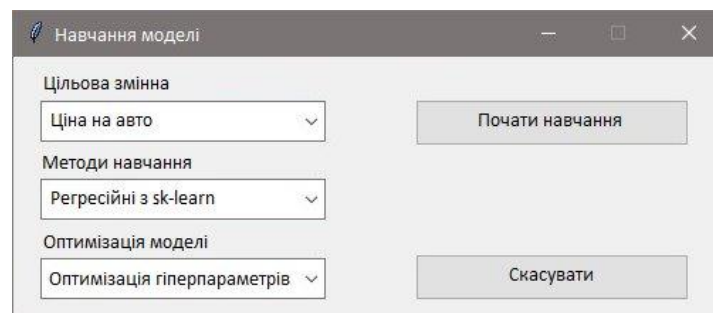


Рисунок 3.26 - Форма вибору режиму навчання моделі

Звісно, всі ці форми наведені тільки як приклади. Точний їх формат буде визначатись в технічному завданні.

### Висновки до розділу

В результаті виконання даного розділу були досягнуті такі результати:

- виконано підключення до API сервісу [gia.com](http://gia.com);
- завантажені та попередньо оброблені вхідні дані;
- проведена реалізація та оптимізація моделі машинного навчання;
- розроблені елементи графічного інтерфейсу користувача.

## РОЗДІЛ 4

### ОХОРОНА ПРАЦІ ТА БЕЗПЕКА У НАДЗВИЧАЙНИХ СИТУАЦІЯХ

Для забезпечення безпеки при використанні системи машинного навчання для оцінювання транспортних засобів агрофірми необхідно враховувати специфіку роботи з такими технологіями та обладнанням. На підприємстві мусять бути розроблені рекомендації щодо охорони праці, техніки безпеки та порядок дій у разі виникнення надзвичайних ситуацій.

При розгортанні системи необхідно дотримуватися загальних правил безпеки при роботі з комп'ютерним обладнанням, периферійними пристроями та спеціалізованими датчиками, які можуть використовуватись для збору даних про стан транспортних засобів.

Робота з електронними системами потребує надійного електрозабезпечення, заземлення обладнання та регулярного технічного обслуговування.

Перед початком роботи слід перевірити стан кабелів, наявність пошкоджень чи порушень ізоляції, а також функціонування програмного забезпечення.

Усі працівники, які працюють із системою, повинні пройти навчання з безпечного використання техніки та ознайомитися з інструкцією експлуатації програмного забезпечення. Це знижує ризик помилок, які можуть призвести до збою системи або аварійної ситуації.

Важливо забезпечити постійний доступ до інструкцій з експлуатації та мати резервні копії ключових даних для уникнення втрат у разі технічних несправностей.

Для захисту працівників від впливу шкідливих факторів, таких як електромагнітне випромінювання, необхідно забезпечити відповідний захист робочого середовища. Робоче місце має бути обладнане засобами ергономіки: зручними столами та кріслами, правильним освітленням, а також мати обладнання для підтримки оптимального мікроклімату. Час безперервної

роботи з системою повинен бути обмежений згідно з нормами, рекомендованими для роботи з комп'ютерними технологіями, наприклад, перерва кожні 60 хвилин.

Додатково слід звернути увагу на забезпечення захисту даних, які обробляються системою. Це включає фізичний доступ до серверів та інших пристроїв лише для уповноважених осіб, а також використання сучасних методів захисту інформації від несанкціонованого доступу або втрат. Забезпечення належного рівня безпеки знижує ризики економічних і технічних втрат.

При виникненні надзвичайних ситуацій, таких як пожежа, аварійна відмова обладнання, ураження працівника електрострумом або хакерська атака на систему, необхідно діяти швидко і відповідно до затверджених інструкцій. У випадку пожежі всі працівники повинні бути евакуйовані з приміщення згідно з планом евакуації, а системи електропостачання слід негайно відключити. Важливо мати в приміщенні вогнегасники та інше протипожежне обладнання, а персонал має бути навчений користуванню цими засобами.

Якщо працівник зазнав ураження електрострумом, необхідно негайно знеструмити відповідну ділянку електромережі та надати першу допомогу. Якщо це можливо, слід викликати медичну допомогу та транспортувати потерпілого до найближчого медичного закладу.

У разі збою роботи обладнання або програмного забезпечення необхідно одразу зупинити використання системи та повідомити відповідальних спеціалістів для діагностики та усунення проблеми. Якщо збій призвів до втрати даних, важливо відновити їх із резервної копії та проаналізувати причини інциденту, щоб запобігти подібним ситуаціям у майбутньому.

У випадку кібератаки необхідно швидко ізолювати систему від зовнішніх мереж, щоб уникнути подальшого розповсюдження загрози. Відповідальні працівники з кібербезпеки повинні провести аналіз загрози, відновити роботу системи та вжити заходів для запобігання повторним атакам.

Важливо, щоб на кожному об'єкті, де використовується система, були розроблені та затверджені локальні інструкції з охорони праці, які враховують специфіку обладнання і програмного забезпечення. Ці інструкції мають бути доведені до кожного працівника під час вступного та повторного інструктажів.

Дотримання всіх наведених заходів гарантує безпечну роботу з системою машинного навчання, мінімізує ризики для здоров'я працівників і зменшує ймовірність виникнення надзвичайних ситуацій.

## РОЗДІЛ 5

### ВИЗНАЧЕННЯ ЕФЕКТИВНОСТІ ІНФОРМАЦІЙНОЇ СИСТЕМИ

Для розрахунку економічної ефективності системи машинного навчання для оцінювання транспортних засобів агрофірми необхідно врахувати витрати на її розробку, впровадження, експлуатацію, а також потенційні економічні переваги від її використання. Економічна ефективність оцінюється за допомогою коефіцієнта економічної доцільності, який відображає співвідношення між витратами та отриманими вигодами.

Складові витрат на впровадження системи включають:

- розробку програмного забезпечення  $C_{\text{розр}}$ , яке забезпечує аналіз і обробку даних, сюди входять створення алгоритмів машинного навчання для аналізу стану транспортних засобів, інтерфейсів для взаємодії з користувачами, модулів для зберігання та обробки даних;
- закупівлю обладнання  $C_{\text{обл}}$ , включаючи сервери, датчики та пристрої для збору даних, комп'ютери для роботи персоналу;
- навчання персоналу  $C_{\text{навч}}$  для роботи із системою та витрати на проведення тренінгів для працівників, які працюватимуть із системою;
- експлуатаційні витрати  $C_{\text{експл}}$ , що охоплюють регулярні оновлення та підтримку програмного забезпечення, поточний ремонт і технічну підтримку обладнання.

Загальні витрати на впровадження системи можна представити формулою:

$$C_{\text{заг}} = C_{\text{розр}} + C_{\text{обл}} + C_{\text{навч}} + C_{\text{експл}} .$$

Економія витрат завдяки системі оцінюється за рахунок таких показників:

- зменшення витрат на ремонт  $E_{\text{ремонт}}$ , оскільки своєчасне виявлення дефектів дозволяє уникнути серйозних пошкоджень техніки;
- оптимізація витрат на паливо  $E_{\text{паливо}}$ , завдяки аналізу ефективності транспортних засобів;
- зниження витрат на простой  $E_{\text{простій}}$ , що пов'язані з меншою кількістю поломок.



Сукупна економія коштів визначається за формулою:

$$E_{\text{заг}} = E_{\text{ремонт}} + E_{\text{паливо}} + E_{\text{простій}} .$$

Для оцінки ефективності системи використовується коефіцієнт рентабельності:

$$R = \frac{E_{\text{заг}}}{C_{\text{заг}}},$$

де  $R > 1$  свідчить про доцільність впровадження системи.

Наведемо приклад розрахунку. Нехай витрати на розробку системи становлять  $C_{\text{розр}} = 500000$  грн, витрати на обладнання —  $C_{\text{обл}} = 300000$  грн, навчання персоналу —  $C_{\text{навч}} = 50000$  грн, а експлуатаційні витрати за рік —  $C_{\text{експл}} = 100000$  грн. Загальні витрати в такому випадку складуть

$$C_{\text{заг}} = 500000 + 300000 + 50000 + 100000 = 950000 \text{ грн.}$$

Нехай економія коштів включає такі показники:

- зменшення витрат на ремонт  $E_{\text{ремонт}} = 400000$  грн;
- оптимізацію витрат на паливо  $E_{\text{паливо}} = 300000$  грн;
- зниження витрат на простої  $E_{\text{простій}} = 200000$  грн.

Сукупна економія становить:

$$E_{\text{заг}} = 400000 + 300000 + 200000 = 900000 \text{ грн.}$$

Розрахуємо коефіцієнт рентабельності:

$$R = \frac{E_{\text{заг}}}{C_{\text{заг}}} = \frac{900000}{950000} \approx 0,95.$$

Оскільки  $R < 1$ , система не є економічно ефективною в короткостроковій перспективі. Однак врахування показників соціально-економічної вигоди, таких як зниження екологічного впливу та підвищення продуктивності агрофірми, може виправдати її впровадження в довгостроковій перспективі. Для цього розраховують загальний коефіцієнт ефективності з урахуванням додаткових вигід:

$$E_{\text{загальний}} = E_{\text{заг}} + E_{\text{соц}} ,$$

де  $E_{\text{соц}}$  — оцінка соціально-економічних переваг.

Якщо, наприклад,  $E_{\text{соц}} = 200000$  грн, то:

$$E_{\text{загальний}} = 900000 + 200000 = 1100000 \text{ грн.}$$

Перерахований коефіцієнт ефективності буде становити:

$$R = \frac{E_{\text{заг}}}{C_{\text{заг}}} = \frac{1100000}{950000} \approx 1,16.$$

Таким чином, з урахуванням показників соціально-економічної вигоди система є економічно вигідною в довгостроковій перспективі. Це свідчить про ефективність і доцільність її впровадження. Система також сприяє сталому розвитку агрофірми, оптимізації логістичних процесів і підвищенню рівня безпеки експлуатації техніки.

## ВИСНОВКИ ТА ПРОПОЗИЦІЇ

В ході виконання кваліфікаційної роботи була розроблена модель машинного навчання для оцінки транспортних засобів. Для цього виконані такі етапи:

1. Проведений аналіз предметної області, особливостей оцінки машин та обладнання, проаналізовані підходи до оцінки та обрано найбільш відповідний для задачі даної роботи.

2. Проаналізовані сучасні підходи до побудови інтелектуальних моделей, типи задач машинного навчання та найпоширені методи їх рішення в залежності від потреб задачі. Обрано метод рішення – регресійна модель.

3. Зроблено вибір інструментарію (мови програмування, середовища та необхідних бібліотек прикладного програмного забезпечення). Реалізована модель, здійснено її навчання, проведена оптимізація та перевірка отриманих результатів.

4. Запропоновані елементи графічного інтерфейсу користувача.

5. Розглянуті питання охорони праці та визначення економічної ефективності отриманих рішень.

6. По результатам роботи опубліковані тези доповіді на студентській науковій конференції [30].

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Особливості оцінки транспортних засобів. URL: <https://estim.com.ua/особливості-оцінки-транспортних-зас/>
2. Сучасні підходи до визначення ринкової вартості транспортних засобів іноземного виробництва. URL: <http://dspace.puet.edu.ua/handle/123456789/4161>
3. Особливості та нюанси оцінювання транспортних засобів. URL: <https://expert-iviv.com/ua/blog/article/11561.htm>
4. Особливості визначення ринкової вартості транспортних засобів з іноземною реєстрацією. URL: <https://naukam.triada.in.ua/index.php/konferentsiji/76-sorok-p-yata-vseukrajinska-praktichno-piznavalna-internet-konferentsiya/1001-osoblivosti-viznachennya-rinkovoji-vartosti-transportnikh-zasobiv-z-inozemnoyu-reestratsieyu>
5. Витратний підхід до оцінки вартості машин та обладнання. URL: <https://pareto.com.ua/ua/blog/vitratnij-pidhid-oczinki-vartosti-mashin-ta-obladnannya/>
6. Порівняльний і витратний підходи до оцінки обладнання. URL: <https://pareto.com.ua/ua/blog/porivnyalniy-i-vitratnij-pidhodi-do-oczinki-obladnannya/>
7. Експертна оцінка ринкової вартості обладнання та машин. URL: <https://www.expert-in.com.ua/ocinka-obladnannya/>
8. Національний стандарт №1 “Загальні засади оцінки майна і майнових прав”. URL: [https://www.aexpert.in.ua/nacstandart\\_no1/](https://www.aexpert.in.ua/nacstandart_no1/)
9. Методичні засади оцінки вартості бізнесу. URL: <https://ekmair.ukma.edu.ua/bitstreams/45f3714c-23ec-4bdd-b3c0-f7956a31396b/download>
10. Оцінка вартості підприємства на основі ринкового підходу. URL: <https://buklib.net/books/26641/>
11. Зубенко А. В. Оцінка бізнесу та об’єктів нерухомості : конспект лекцій для

- студентів денної і заочної форм навчання освітнього рівня «бакалавр» за спец. 071 – Облік і оподаткування / А. В. Зубенко; Харків. нац. ун-т міськ. госп-ва ім. О. М. Бекетова. – Харків : ХНУМГ ім. О. М. Бекетова, 2018. – 88 с.
12. Теоретичні та практичні аспекти оцінки вартості підприємства. URL: [https://ev.nmu.org.ua/docs/2013/1/EV20131\\_080-085.pdf](https://ev.nmu.org.ua/docs/2013/1/EV20131_080-085.pdf)
  13. Основи кореляційного та регресійного аналізу. URL: [https://pchilka-litsei.in.ua/excel-book/basis\\_analysis.html](https://pchilka-litsei.in.ua/excel-book/basis_analysis.html)
  14. Кореляційно-регресійний аналіз. URL: <https://buklib.net/books/30354/>
  15. Використання регресійного аналізу для розуміння складних взаємозв'язків. URL: <https://mindthegraph.com/blog/uk/регресійний-аналіз/>
  16. Регресійний аналіз. URL: [https://uk.wikipedia.org/wiki/Регресійний\\_аналіз](https://uk.wikipedia.org/wiki/Регресійний_аналіз)
  17. Основи машинного навчання : навч. посіб. / В. О. Харченко. – Суми : Сумський державний університет, 2023. – 264 с.
  18. Що таке машинне навчання? Види машинного навчання. URL: <https://beetroot.academy/blog/mashinne-navchannya-abo-naykrashcha-vechirka-v-misti>
  19. Що таке Machine Learning? URL: <https://denovo.ua/resources/what-is-machine-learning>
  20. Що таке лінійна регресія? URL: <https://www.unite.ai/uk/what-is-linear-regression/>
  21. Логістична регресія. URL: [https://uk.wikipedia.org/wiki/Логістична\\_регресія](https://uk.wikipedia.org/wiki/Логістична_регресія)
  22. Random forest. URL: [https://uk.wikipedia.org/wiki/Random\\_forest](https://uk.wikipedia.org/wiki/Random_forest)
  23. Gradient boosting. URL: <https://itwiki.dev/data-science/ml-reference/ml-glossary/gradient-boosting>
  24. Основні поняття кластеризації та постановка задачі. URL: [https://csc.knu.ua/media/study/asp/mod\\_probl\\_inf\\_tech\\_sys\\_analysis\\_ivohin/lecture/lec11.pdf](https://csc.knu.ua/media/study/asp/mod_probl_inf_tech_sys_analysis_ivohin/lecture/lec11.pdf)
  25. Штучна нейронна мережа. URL: [https://uk.wikipedia.org/wiki/Штучна\\_нейронна\\_мережа](https://uk.wikipedia.org/wiki/Штучна_нейронна_мережа)
  26. Як працюють нейронні мережі? URL: <https://aperep.kpi.ua/neural-networks/en>

27. Регуляризація (математика). URL:  
[https://uk.wikipedia.org/wiki/Регуляризація\\_\(математика\)](https://uk.wikipedia.org/wiki/Регуляризація_(математика))
28. Ласкаво просимо до Colab! URL: <https://colab.research.google.com/?hl=uk>
29. RIA.com для розробників. URL: <https://developers.ria.com/>
30. Стецишин О. Огляд методів машинного навчання для оцінювання транспортних засобів. URL:  
[https://lnup.edu.ua/attachments/article/8156/СТУД.ФОРУМ\\_2024.pdf#page=369](https://lnup.edu.ua/attachments/article/8156/СТУД.ФОРУМ_2024.pdf#page=369)