КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ТАРАСА ШЕВЧЕНКА

Факультет інформаційних технологій

Кафедра технологій управління

Спеціальність 122 – Комп'ютерні науки,
освітня програма «Інформаційна аналітика та впливи»

КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА

на тему:

«Інформаційне забезпечення прогнозування цін на нерухомість методами машинного навчання»

Студента 2-го курсу групи ІАВ-21

Бурої Юлії Сергіївни

(прізвище, ім'я, по батькові)

(підпис студента)

Науковий керівник:

доктор технічних наук, доцент

(науковий ступінь, вчене звання)

Хлевна Юлія Леонідівна

(прізвище, ім'я, по батькові)

(дата) (підпис)

Попередній захист:

(Висновок: «До захисту в Екзаменаційній комісії»)

Завідувач кафедри
технологій управління

(підпис) (прізвище, ініціали) (дата)

Київ – 2021

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА
Факультет інформаційних технологій

Кафедра технологій управління
Освітньо-кваліфікаційний рівень Магістр
Спеціальність 122 - Комп'ютерні науки
Освітня програма Інформаційна аналітика та впливи

ЗАТВЕРДЖУЮ
Завідувач кафедри
професор Морозов В.В.

«____» _____20___року

З А В Д А Н Н Я
НА ВИКОНАННЯ КВАЛІФІКАЦІЙНОЇ РОБОТИ

Студент: Бура Юлія Сергіївна

Група: ІАВ-21

1. Тема кваліфікаційної роботи Інформаційне забезпечення прогнозування цін на нерухомість методами машинного навчання
Затверджена наказом по від «09» листопада 2020 р. №4.

2. Строк подання студентом готової роботи – « 7 » травня 2020 р.

3. Цільова установка та вихідні дані до роботи: інформаційне забезпечення прогнозування цін на нерухомість побудована за допомогою мови програмування Python і реалізована із використанням інструменту Amazon Forecast, база даних складається з 2930 спостережень та 81 змінної

4. Зміст роботи Методологія аналізу ринку нерухомості: визначення предметної області дослідження методологія формування впливів на показники цін на нерухомість, аналіз методів прогнозування цін на нерухомість, аналіз динаміки ринку нерухомості України, методологія застосування даних про нерухомість в моделях машинного навчання. Концептуалізація інформаційного забезпечення прогнозування цін на нерухомість методами машинного навчання: формальне представлення прогнозування цін на нерухомість для використання комп'ютерними системами, методи підготовки бази даних для прогнозування, формування математичного апарату прогнозування цін на нерухомість, реалізація математичного апарату прогнозування цін на нерухомість у комп'ютерних система. Побудова моделей прогнозування цін на нерухомість з використанням інформаційного забезпечення: аналіз бази даних та створення системи прогнозування цін на нерухомість, підготовка даних для побудови моделей та прогнозування, застосування моделей регуляризації Lasso, Ridge та Elastic Ne, використання градієнтного бустінгу та XGBoost моделі для прогнозування, створення агрегованої моделі та порівняння результатів. Розробка інформаційного забезпечення прогнозування цін на

нерухомість та рекомендації щодо його використання: вибір інструмента для реалізації інформаційного забезпечення прогнозування цін на нерухомість, алгоритм побудови інформаційного забезпечення, практичне використання інформаційного забезпечення у бізнесі.

5. Перелік графічного матеріалу (слайдів) 2-ох таблиць, 25 рисунків, 24 формул, 3-ох додатків, 25 слайдів презентації доповіді.

6. Календарний план виконання роботи:

№ п/п	Назва частин роботи	%	Виконання роботи	
			За планом	Фактично
1.	Вибір теми дипломної роботи	3	01.10.2020	01.10.2020
2.	Протокол кафедри ТУ про затвердження тем дипломних робіт та призначення наукових керівників	2	09.11.2020	09.11.2020
3.	Формування переліку нормативних матеріалів, літератури з проблематики дипломної роботи	10	08.01.2021	07.01.2021
4.	Складання розгорнутого плану кваліфікаційної роботи	5	18.01.2021	18.01.2021
5.	Ознайомлення наукового керівника з розгорнутим планом кваліфікаційної роботи. Внесення змін.	5	19.01.2021 – 20.01.2021	20.11.2021
6.	Підготовка розділу 1 «Методологія аналізу ринку нерухомості»	10	12.02.2021	13.02.2021
7.	Підготовка розділу 2 «Концептуалізація інформаційного забезпечення прогнозування цін на нерухомість методами машинного навчання»	14	08.03.2021	08.03.2021
8.	Підготовка розділу 3 «Побудова моделей прогнозування цін на нерухомість з використанням інформаційного забезпечення»	14	01.04.2021	01.04.2021
9.	Підготовка розділу 4 «Розробка інформаційного забезпечення прогнозування цін на нерухомість та рекомендації щодо його використання»	13	20.04.2021	20.04.2021
10.	Оформлення кваліфікаційної роботи. Підготовка висновків і пропозицій	15	29.04.2021	29.04.2021
11.	Передача кваліфікаційної роботи науковому керівникові	2	04.05.2021	04.05.2021
12.	Передача кваліфікаційної роботи рецензенту для рецензування	2	05.05.2021	05.05.2021
13.	Попередній захист кваліфікаційної роботи	5	11.05.2021	11.05.2021

14.	Подача готової роботи на кафедру	-	20.05.2021	20.05.2021

Дата видачі завдання «__1__» _____жовтня_____2020 р.

Керівник роботи: доктор технічних наук, доцент Хлевна Юлія Леонідівна
(посада, прізвище, ім’я, по батькові)

_____
(підпис)

Завдання прийняв до виконання студент групи ІАВ-21

Бура Юлія Сергіївна
(прізвище, ім’я, по батькові)

_____
(підпис)

ЗМІСТ

АНОТАЦІЯ

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА
Факультет інформаційних технологій
Кафедра технологій управління
Спеціальність 122 - Комп'ютерні науки,
освітня програма «Інформаційна аналітика та впливи»

Дипломна робота магістра Бурої Юлії Сергіївни.

Тема роботи – «Інформаційне забезпечення прогнозування цін на нерухомість методами машинного навчання».

Мета дипломної роботи магістра – дослідити методи прогнозування, які можуть бути використані для моделювання ринку нерухомості, створити інформаційне забезпечення прогнозування цін на нерухомість на основі найкращої побудованої моделі, яке може бути використано зацікавленими особами ринку нерухомості.

Об'єкт дослідження – математичне моделювання та прогнозування цін на ринку нерухомості.

Предмет дослідження – інформаційні засоби та технології управління даними і процесами при прогнозуванні цін на нерухомість.

Наукова новизна роботи – створена нова агрегована модель прогнозування цін на нерухомість на основі існуючих математичних методів та підходів, на її основі запропоновано алгоритм побудови інформаційного забезпечення, унікальність якого полягає в використанні Amazon Forecast, який дозволяє створити продукт для безпосереднього використання у бізнесі.

У роботі досліджуються основні тенденції й закономірності ринку нерухомості, а також існуючі комп'ютерні математичні методи для його моделювання. База даних складається з 2917 спостережень та 81 змінної. На основі зазначеного датасету були побудовані моделі прогнозування цін на

нерухомість з використанням таких інструментальних засобів, як MS Excel, Python та R (зокрема Jupyter Notebook та R Studio). Також наведено алгоритм розгортання та імплементації отриманих математичних моделей у комп'ютерні системи для подальшого використання зацікавленими особами.

Дипломна робота складається зі вступу, основної частини, яка включає чотири розділи, висновків, списку використаних джерел та додатків. Всього налічує 106 сторінок та перелік посилань з 70 джерел на 7-ми сторінках.

Ключові слова: ринок нерухомості, ціни на нерухомість, машинне навчання, Lasso регресія, Ridge регресія, Elastic Net регресія, градієнтний бустинг, XGboost.

ВСТУП

Актуальність дослідження. У сучасному, прогресивному світі найбільшою цінністю користуються дані. Вони нагромаджуються з неймовірною швидкістю, кожного дня до баз даних по усьому світі надходять терабайти інформації. Великі компанії готові платити значні суми лише за отримання цих баз даних, дані стали своєрідною валютою сучасного світу, а за правильної інтерпретації, аналізу й використання вони здатні перетворюватись на реальні гроші. В таких умовах актуальним є питання швидкої, продуктивної та якісної обробки й аналізу цих даних й, як результат, отримання правильних висновків і результатів, на основі яких будуть прийматись подальші рішення та будуватись стратегія. Сектор нерухомості, як в Україні та і у світі в цілому, є одним із секторів з найбільшою капіталізацією, починають інвестуватися нові будівельні проекти, через ринок нерухомості проходять тисячі об’єктів й укладаються відповідні угоди. Аналіз, моделювання й прогнозування цін об’єктів на ринку нерухомості сприятиме прийняттю правильних й найбільш вигідних рішень й укладанню відповідних угод. А у розрізі того, що обсяг використовуваних даних великий, є необхідність їх обробки із застосуванням комп’ютерних систем і технологій, зокрема методів машинного навчання, що дозволить якісно, а головне швидко й ефективно оброблювати отримані дані.

Особливому розвитку даної галузі сприяв розвиток комп’ютерно-обчислювальних систем. І, так як розробка високонавантажених, складних систем потребує ресурсів, то основні наукові дослідження зосередилися у великих транснаціональних компаній, таких як Google, IBM, Microsoft, Amazon. Над цим питанням працювали такі вчені як Anna Radzewicz, Rafal Wisniewski, Ewa Kucharska-Stasiak, Jonathan Mallinson та інші [33-35]. Так як дослідження даної галузі вимагає значних потужностей й фінансування, основне просування відбувається в країнах Європи та США.

Але питання актуальне і на теренах України, оскільки забезпечення об'єктивною інформацією осіб, які приймають рішення про проведення тих або інших операцій на ринку нерухомості є актуальною прикладною задачею. Рішення якої можна інтерпретувати з допомогою розробки інформаційної технології, в основу якої закладено аналіз, моделювання й прогнозування цін об’єктів на ринку нерухомості. Відповідно науковою задачею є імплементація раціонального методу прогнозування у таку технологію.

Мета роботи – дослідити методи прогнозування, які можуть бути використані для моделювання ринку нерухомості, створити інформаційне забезпечення прогнозування цін на нерухомість на основі найкращої побудованої моделі, яке може бути використано зацікавленими особами ринку нерухомості. Для реалізації поставленої мети необхідним є вирішення таких **завдань**:

– визначити та описати предметну область дослідження;

– виокремити основні впливи на показники цін на нерухомість;

– виокремити методи прогнозування, які можуть бути використані для моделювання цін на нерухомість;

– проаналізувати динаміку й закономірності ринку нерухомості України;

– дослідити методологію застосування даних про нерухомість в моделях машинного навчання;

– проаналізувати засади формального представлення прогнозування цін на нерухомість для прогнозування;

– дослідити доступні методи підготовки цільової змінної та бази даних для прогнозування;

– проаналізувати базу даних та підготувати дані до подальшого прогнозування;

– побудувати декілька моделей прогнозування цін на нерухомість з використанням інформаційного забезпечення, порівняти їх та, базуючись на отриманих результатах, обрати найкращу;

– розробити алгоритм побудови інформаційного забезпечення та проаналізувати практичне застосування створеного продукту у бізнесі.

Об'єктом дослідження є математичне моделювання та прогнозування цін на ринку нерухомості.

Предметом дослідження є інформаційні засоби та технології управління даними і процесами при прогнозуванні цін на нерухомість.

Методи досліджень. Теоретичною основою дослідження стали загальнонаукові методи пізнання: системність, комплексність, аналіз та синтез. Математичний апарат для вирішення задач прогнозування цін на нерухомість включає елементи регресійного аналізу та машинного навчання, зокрема використані такі методи моделювання як Lasso регресія, Ridge регресія, Elastic Net регресія, градієнтний бустінг та XGboost.

Для вирішення завдань розробки інформаційного забезпечення прогнозування цін на нерухомість методами машинного навчання використано такі інструментальні засоби, як MS Excel, Python та R (зокрема Jupyter Notebook та R Studio).

Наукова новизна одержаних результатів

Створена нова агрегована модель прогнозування цін на нерухомість на основі існуючих математичних методів та підходів, яка підвищує точність прогнозування на 5% порівняно зі стандартними підходами. На її основі запропоновано алгоритм побудови інформаційного забезпечення, унікальність якого полягає в використанні Amazon Forecast, який дозволяє створити продукт для безпосереднього використання у бізнесі.

Практичне значення отриманих результатів.

1. Розроблено алгоритмічне, інформаційне забезпечення прогнозування цін на нерухомість методами машинного навчання.

2. Розроблено алгоритм автоматизованого прогнозу й прогнозування цін на нерухомість методами машинного навчання.

Особистий внесок здобувача. Усі наукові результати, які відображено у кваліфікаційній роботі, отримані автором самостійно. Результати співавторів сумісних публікацій до тексту кваліфікаційної роботи не включено. У надрукованих статтях, опублікованих у співавторстві, магістранту належить наступне:

– побудована агрегована модель прогнозування цін на нерухомість;

– на основі створеної моделі прогнозування реалізовано інформаційне забезпечення прогнозування цін на нерухомість за допомогою Amazon Forecast та інших допоміжних інструментів.

Апробація результатів роботи. Автор виступала доповідачем на VI Information Technology and Interactions (Satellite): Conference Proceedings

Публікації. Основні наукові положення, висновки і результати магістерської кваліфікаційної роботи знайшли відображення у двох друкованих працях, з них: 1 стаття у іноземному виданні та 1 тези доповіді на конференції.

Структура та обсяг роботи. Магістерська робота складається зі вступу, 4 розділів, висновків, списку використаних джерел з 56 найменувань та додатків. Загальний обсяг курсової становить 106 сторінок, із них 79 сторінок основного тексту, який містить 25 рисунків.

РОЗДІЛ 1
МЕТОДОЛОГІЯ АНАЛІЗУ РИНКУ НЕРУХОМОСТІ

1.1 Визначення предметної області дослідження

Оцінювання нерухомості – це встановлення офіційної ціни на нерухомість з видачою відповідного сертифіката. Офіційна ціна зазвичай наближена до ринкової ціни. Така оцінка потрібна, оскільки операції з нерухомістю не є такими ж простими як, скажімо, з цінними паперами, і трапляються рідше. Будь-яка власність має бути оцінена. Однак вартість нерухомості значно відрізняється залежно від розташування – це важливий фактор в процесі оцінки. Отже, неможливо створити якусь біржу, чи централізований аукціон на землю (на відміну, наприклад, від фондових бірж на ринку цінних паперів, валютних ринках тощо). Необхідний унікальний підхід для кожного об’єкта, зазвичай оцінюванням займаються відповідні спеціалісти, але, як вже зазначалось, в Україні дана галузь жорстко монополізована й закрита, тому створення системи з аналізу нерухомості не тільки підвищить продуктивність та дозволить автоматизувати процес, але й внесе чіткість та прозорість у галузь.

Ефективне функціонування ринкової системи неможливе без ринку нерухомості, оскільки він справляє значний вплив на розвиток відносин власності та становлення середнього класу – основи суспільства, сприяє задоволенню потреб підприємців у постійних активах, значною мірою визначає рівень споживання, нагромадження та інвестування [58]. Сектор нерухомості є істотною складовою будь-якої національної економіки, оскільки нерухомість – це найважливіша частина національного багатства, країни та її регіонів [61].

Якщо говорити саме про економіку нерухомості, то можна сказати, що це застосування економічних методів на ринках нерухомості. Він намагається описати, пояснити та передбачити закономірності цін, пропозиції та попиту [33]. Тісно пов’язана галузь економіки житла є вужчою, зосереджуючись на ринках

житлової нерухомості, тоді як дослідження тенденцій нерухомості фокусується на бізнесі та структурних змінах, що впливають на галузь. Обидва базуються на аналізі часткової рівноваги (попит та пропозиція), міській економіці, просторовій економіці, основних та великих дослідженнях, опитуваннях та фінансах.

Розвиток економіки будь-якої держави не можливий без існування ринку нерухомості, оскільки всі господарюючі суб'єкти виступають учасниками цього ринку в купівлі-продажу або оренді нерухомості, необхідної для провадження їхньої діяльності. Тому ринок нерухомості вважається своєрідним індикатором розвитку економіки країни: висока активність на ньому свідчить про економічне зростання держави, потребу підприємств у розширенні виробничих і адміністративних площ, можливості громадян покращувати свої житлові умови тощо [59-61].

У роботі запропоновано оперувати наступними термінами та поняттями.

Ринок нерухомості – певний набір механізмів, за допомогою яких відбувається передача прав власності на об'єкт нерухомості, розподіл земельних прав між учасниками ринку [31].

Нерухомість – це власність, що складається із землі, будівель на ній та будь-яких природних ресурсів у межах власності, таких як води та врожаї. Нерухомість може бути розділена на чотири типи: житлова, комерційна, промислова та земельна. Нерухомість може включати майно, землю, будівлі, права на повітря над землею та підземні права під землею. Цей термін стосується реального або фізичного майна. Як бізнес-термін, нерухомість також означає виробництво, купівлю та продаж майна.

Житло – своєрідний товар з дуже тривалим терміном використання. За своєю природою цей товар не може без екстраординарних причин швидко знецінюватися, навпаки, ціни на ринку житла в усьому світі зростають [62].

1.2 Методологія формування впливів на показники цін на нерухомість

Аналіз ринку нерухомості дозволяє припустити [1], що впливи на ціни об'єктів житлової нерухомості, можна розділити на дві основні групи:

– локальні;

– глобальні.

Локальні впливи формують ціни на всі квартири в одній місцевості різні. Одна квартира більш вдало розташована, в іншій площа кухні більше, у третій зроблений гарний ремонт. Ці причини створюють всю гаму цін на житло в даний момент часу і, взагалі кажучи, слабо залежать від часу.

Вплив локальних причин можна описати тими самими оціночними коректуваннями, які оцінювачі використовують для приведення ціни одного об'єкта до ціни іншого.

Друга група причин, що формують процес ціноутворення, – це **глобальні впливи**. Вони пов'язані з макроекономічними параметрами, такими, як рівень розвитку економіки та бізнесу в місті, рівень доходів населення і рівень життя в цьому місті, а також його статус і престиж. Причому різниця в цінах на аналогічну нерухомість, що знаходиться в різних містах, також приблизно пропорційно один одному. Це дозволяє говорити про порівняння загального рівня цін в одному місті з рівнем цін в іншому і стверджувати, що співвідношення цін на аналогічну квартиру у різних містах буде приблизно пропорційно співвідношенню рівнів цін в цих містах [3].

Якщо розглядати механізм формування ринку нерухомості, то основними детермінантами попиту на житло є демографічні показники. Але інші фактори, такі як дохід, ціна житла, вартість та доступність кредиту, споживчі переваги, уподобання інвесторів, ціна замінників та ціна на доповнення, також відіграють свою роль.

Основними демографічними змінними є чисельність та приріст населення: чим більше людей в економіці, тим більший попит на житло. Для розширеного тлумачення, необхідно також враховувати розмір сім'ї, віковий склад сім'ї, кількість перших і других дітей, чисту міграцію (імміграція мінус еміграція), формування несімейних домогосподарств, кількість подвійних сімей, рівень смертності, рівень розлучень та шлюби. В економіці житла елементарною одиницею аналізу є не людина, як це відбувається у стандартних моделях часткової рівноваги. Швидше за все, домогосподарства вимагають житлових послуг: як правило, одне домогосподарство на будинок. Розмір та демографічний склад домогосподарств є різними та не зовсім екзогенними. Це ендогенно для ринку житла в тому сенсі, що зі зростанням ціни на житлово-комунальні послуги розмір домогосподарств також має тенденцію до зростання [37].

Дохід також є важливою детермінантою. Багато економістів з питань житла використовують постійний дохід, а не річний, через високу вартість придбання нерухомості. Для багатьох людей нерухомість стане найдорожчим предметом, який вони коли-небудь придбають.

Для моделювання цін на нерухомість необхідно також розуміти що таке попит на житло та як він формується. Попит на житло окремого домогосподарства може бути змодельований за допомогою стандартної теорії корисності / вибору. Функція корисності матиме наступний вигляд:

$$U = U(X_1, X_2, X_3, X_4, \dots, X_n), \tag{1.1}$$

де U – доступний дохід домогосподарства;

X_s – товари та послуги.

Функція 1.1 є числовим представленням відношення переваги, тобто здатності споживача порівнювати потенціальні товари та послуги, вона максимізує корисність отриману від споживання певного блага.

Вона може бути побудована, в якій корисність домогосподарства є функцією різних товарів та послуг (X_s). Це буде підпорядковане бюджетним обмеженням, яке можна записати наступним чином:

$$P_1X_1 + P_2X_1 + \cdots + P_nX_n = Y, \quad (1.2)$$

де U – доступний дохід домогосподарства;

X_s – різні товари та послуги;

P_s – ціни на різні товари та послуги.

Функція 1.2 свідчить про те, що гроші, витрачені на всі товари та послуги, повинні дорівнювати наявному доходу. Оскільки це нереально, модель повинна бути скоригована з урахуванням запозичень та економії. Потрібна міра багатства, доходу за життя або постійного доходу. Модель також повинна бути скоригована з урахуванням неоднорідності нерухомості. Це можна зробити, доробивши функцію корисності. Якщо житлові служби (X_4) розділити на складові компоненти ($Z_1, Z_2, Z_3, \dots, Z_n$), функцію корисності можна переписати як:

$$U = U(X_1, X_2, X_3, (Z_1, Z_2, Z_3, \dots, Z_n), \dots, X_n), \quad (1.3)$$

де U – доступний дохід домогосподарства;

X_s – різні товари та послуги;

Z_s – складові компоненти житлової нерухомісті.

Змінюючи ціну на житлово-комунальні послуги (X4) та вирішуючи пункти оптимальної корисності, можна побудувати графік попиту домогосподарств на житлові послуги. Попит на ринку розраховується шляхом підсумовування всіх індивідуальних потреб домогосподарств та є важливим фактором при формуванні цін на нерухомість.

1.3 Аналіз методів прогнозування цін на нерухомість

Наявність інформаційних прогалин перешкоджає розробці прямих моделей, що ілюструють взаємозв'язки на ринку нерухомості. Ринок нерухомості – це недосконала система, де процеси та співвідношення можуть бути передбачені з певною мірою ймовірності, а людські фактори сприяють випадковому характеру відносин у цій системі. Ринок нерухомості недосконалий через важкий доступ до інформації та недостатню кількість даних, і ці проблеми часто стикаються з оцінювачами власності на щоденній практиці [31].

Непрогнозований характер ринку нерухомості значною мірою перешкоджає розробці комплексних аналітичних моделей, що ілюструють ринкові функції. Моделювання ринку методами машинного навчання - це оптимальний інструмент для відтворення ринкових процесів в експериментальних умовах, він враховує порушення, спричинені випадковими факторами, і підтримує формування додаткової інформації про ринок нерухомості.

Ринок нерухомості – специфічна та недосконала область досліджень з міждисциплінарним та системним характером [32]. Він формується за допомогою ідентифікованих процесів та кореляцій, які часто можна передбачити із заданою ймовірністю, а також випадкових процесів та взаємозв’язків (випадковість - відсутність порядку або передбачувана поведінка).

Ймовірність – це міра випадкової події, випадкові елементи (компоненти) на ринку нерухомості пов'язані з [2]:

- відсутністю однорідних даних;
- неоднорідного доступу до даних;
- недоступності вичерпної інформації для учасників ринку нерухомості;
- невизначеність ринкових структур та функцій;
- нестабільності властивостей власності;
- емоційного підходу учасників ринку до операцій тощо.

Дані ринку нерухомості обтяжені невизначеністю, що означає, що наслідки рішень будуть різними в різних ринкових сценаріях, і ймовірність того, що даний сценарій відбудеться, невідома [37]. Вищезазначене суттєво перешкоджає розробці комплексної аналітичної моделі системи нерухомості. Людський фактор значною мірою відповідає за випадковий характер відносин на ринку нерухомості.

Наявність інформаційних прогалин перешкоджає розробці прямих моделей, що ілюструють взаємозв'язок та залежність на ринку нерухомості. Вищезазначену проблему можна вирішити за допомогою інструментів машинного навчання, які генерують додаткові ринкові дані та допомагають побудувати надійну модель.

Під час прогнозування цін на житло можна використовувати різні методи прогнозування: алгоритми екстраполяції експериментальних даних у простих інженерних розрахунках та програмних продуктах, технічний аналіз, а також більш громіздкі статистичні методи, що використовують параметричні моделі. Але вони погано справляються із завданнями з великою кількістю нечітких змінних. В останні десятиліття машинне навчання широко використовується для прогнозування в погано формалізованих умовах [5].

Результати використання цього підходу і багатьох випадках показують його переваги в порівнянні з іншими існуючими методами прогнозування: ефективність у вирішенні неформальних або погано формалізованих проблем, стійкість до частих змін зовнішнього середовища, ефективність в роботі з великою кількістю суперечливої інформації або з неповною інформацією про об'єкт передбачення.

З метою побудови найбільш якісної системи прогнозування цін на нерухомість було проведено дослідження та аналіз можливостей методів машинного навчання як ефективного засобу прогнозування вартості об’єктів нерухомості з наявними вхідними даними з мінімальною помилкою щодо реального стану ринку нерухомості.

1.4 Аналіз динаміки ринку нерухомості України

Незважаючи на стрімкий ріст вартості комунальних послуг й помітне просідання національної валюти, ринок нерухомості в останні роки в України знаходиться в зваженому, стабільному стані (рис. 1.1).

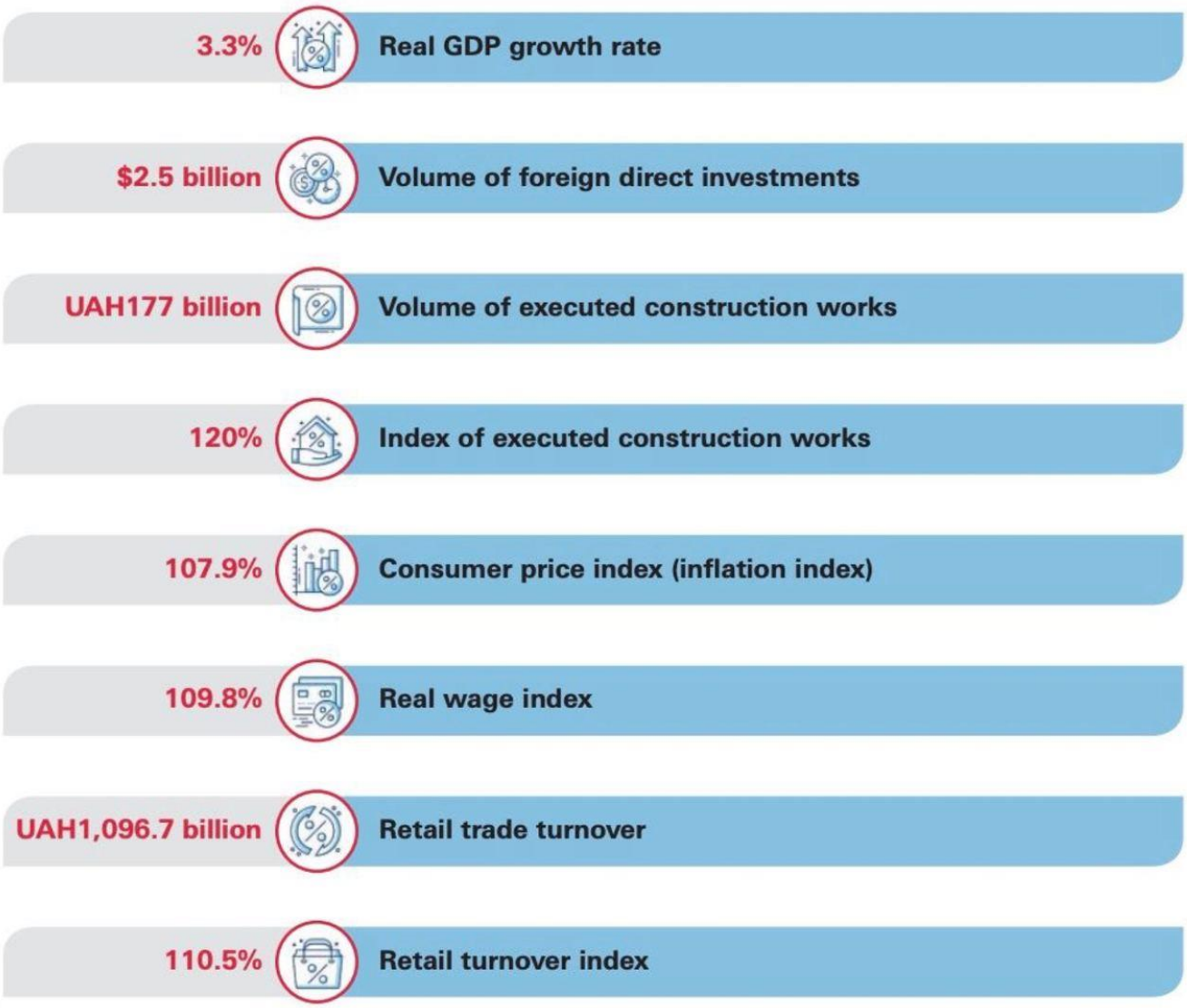

Рисунок 1.1 – Статистика ринку нерухомості України за 2020 рік. [4]

Після падіння на дивовижні 72% від свого піку в 3-му кварталі 2008 року (рис. 1.2), ціни на квартири в Києві в Україні зараз стабільні або поступово зростають, так як українська економіка відновлюється, а конфлікт з Росією

заспокоюється. Це призвело до значного зменшення корупції за оцінкою легкості ведення бізнес-столів Світовим банком [38].

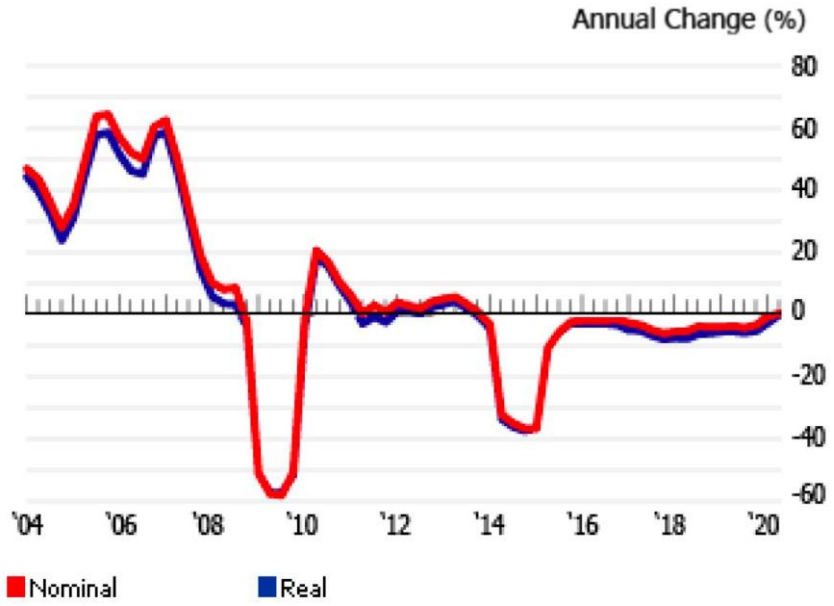

Рисунок 1.2 – Динаміка зміни цін на нерухомість в Україні за 2004-2020 рр. [38-39]

Існуючі ціни на квартири в Києві зросли на 0,39% протягом року до 2 кварталу 2020 року (стабільно в реальному вираженні), до 1035 доларів США за квадратний метр (кв. М), після скорочення на 3,34% у 2019 році, на 4,03% у 2018 році, 6,02% у 2017 році, 1,61% у 2016 році, 2,31% у 2015 році та величезне падіння у 36,62% у 2014 році, згідно з даними S&V Development [42].

Протягом останнього кварталу існуючі ціни на квартири зросли на 0,39% (0,93% у реальному вираженні).

Подібним чином, ціни на новозбудовані квартири в Києві зросли на 0,43% у-о-у у другому кварталі 2020 року (0,1% у реальному вираженні), до 930 дол. США за кв. м. Щоквартально ціни на новозбудовані квартири зростали на 0,43% у II кварталі 2020 р. (1% в реальному вираженні) [44].

Ціни на житлову нерухомість падали понад шість років, особливо у 2014 році (з падінням цін на 36,6%) через девальвацію гривні внаслідок російської війни.

Економіка України зросла на здорових 3,2% у 2019 році порівняно з роком раніше, після річного зростання на 3,3% у 2018 році, 2,5% у 2017 році та 2,4% у 2016 році [40].

Цього року, за прогнозами, економіка України скоротиться на 7,7%, перш ніж повернеться назад із зростанням на 3,6% у 2021 році через економічні наслідки від спалаху COVID-19, згідно з даними Міжнародного валютного фонду (МВФ).

Незважаючи на пандемію, Національний банк України (НБУ), центральний банк країни, залишається позитивним щодо світогляду ринку житла.

«Пандемія мала помітний вплив на ринок житла, хоча і короткочасний і обмежений», - зазначив центральний банк у своєму звіті про фінансову стабільність у червні 2020 року. «На відміну від попередніх епізодів кризи в Україні, цього разу масштабних ринкових трансформацій не передбачається. Ринок повернеться до рівноваги, оскільки карантинні обмеження поступово послаблюються. Незважаючи на кризу, попит на житло залишатиметься високим, але найближчим часом навряд чи повернеться до зростання» [45].

«Аналітики прогнозують, що ціни будуть продовжувати рости, стимульовані слабкою гривнею та низькою націнкою для забудовників. Розробники також оптимістичні. Опитування забудовників, проведене в травні, показало, що майже половина з них - удвічі більше, ніж у лютому - очікували зростання цін на житло», - зазначив центральний банк.

Не існує великих обмежень для іноземців, які купують нерухомість в Україні. Усі вторинні житлові операції (тобто перепродаж) здійснюються в

доларах США, тоді як первинні продажі вказані в гривнях, але все ще оплачуються в доларах [42].

Ціни на землю в Україні продовжують зростати (рис. 1.3). Усі регіони, крім Донецька, зареєстрували зростання цін на землю протягом року до 2-го кварталу 2020 року, базуючись на цифрах, що стосуються S&V Development [38]:

- У Київській області вартість землі зросла на 2,1% (в реальному вираженні - 1,7%) і становила 1421 долар США за 100 кв.
- В Одеській області ціни на землю зросли на 3,4% (3% в реальному вираженні) - до 2200 доларів США за 100 кв.
- У Львівській області ціни на землю зросли на 2,2% (в реальному вираженні - 1,9%) до 909 доларів США за 100 кв.
- У Дніпропетровській області ціни на землю зросли на 1,6% (в реальному вираженні - 1,3%) до 1120 доларів США за 100 кв.
- У Харківській області ціни на землю дещо зросли на 1,2% (0,8% у реальному вираженні) до 931 доларів США за 100 кв.
- Донецька область - єдина область, яка зафіксувала річний спад на 8,3% (-8,7% у реальному вираженні) в середньому до 1088 доларів США на 100 кв.

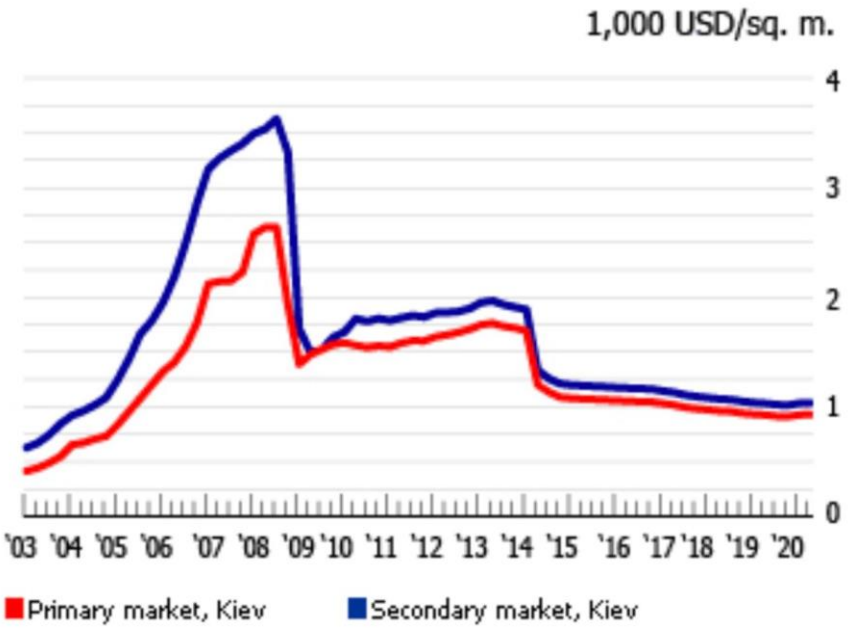

Рисунок 1.3 – Середня ціна на нерухомість в Україні за 2003-2020 рр. [42-44]

Оренда квартир у Києві падає з початку світової кризи. З 2012 по 2019 рік орендні ставки в столиці впали приблизно на 45%, виходячи з даних S&V Development. Але цього року ринок оренди показав ознаки поліпшення (рис. 1.4).

У серпні 2020 року:

- орендна плата за однокімнатні квартири становила 276 доларів США на місяць, що на 1,5% більше, ніж у серпні 2019 року;
- у двокімнатних квартирах орендна плата становила 374 долари США на місяць у серпні 2020 року, збільшившись на 1,9% роком раніше;
- орендна плата за трикімнатні квартири становила 444 долари США на місяць, що на 1,8% більше, ніж рік тому.

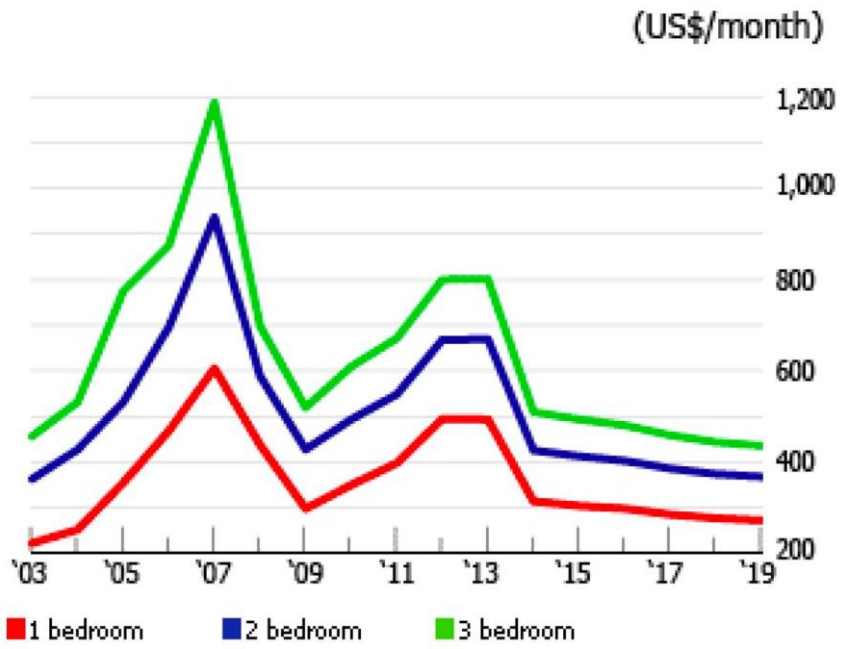

Рисунок 1.4 – Динаміка цін на оренду квартири в Україні за 2003-2019 рр. [41, 42]

Найдорожчим місцем розташування в Києві є Шевченківський район, орендна плата за трикімнатні квартири становить близько 591 долар США на місяць у серпні 2020 року, за ним слідують Печерський район (560 доларів США на місяць) та Оболонський район (457 доларів США на місяць).

В останні роки більшість котирувань орендної плати перейшли на національну валюту, гривню, щоб захистити орендодавців від коливань валюти.

За даними Державної служби статистики, кількість квартир в Україні в 2019 році зросла на 1,6% - до 17,38 млн. одиниць (рис. 1.6). Це означає приблизно 280 000 додаткових квартир - найбільший приріст з 1995 року.

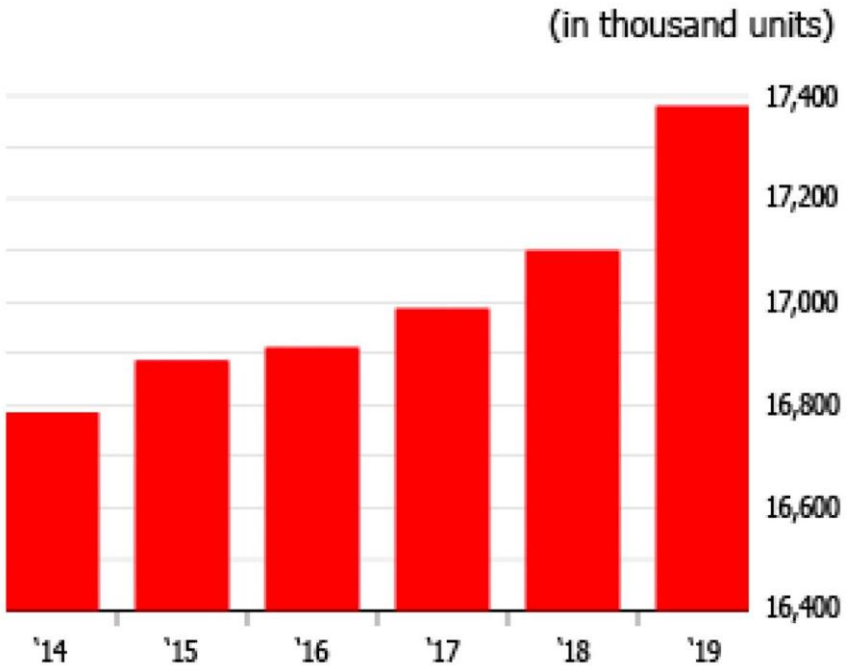

Рисунок 1.5 – Загальна кількість побудованого житла в Україні за 2014-2019 рр. [40]

Так само загальна площа житлового фонду зросла на 1,8% р / р до 1,01 млрд. кв. м. за той самий період [45].

В 2019 році в державі прийняли в експлуатацію 9366,8 тис. квадратних метрів житла. З них 242,5 тис. кв. м. (у т.ч. 2,6% загального обсягу житла) прийняли відповідно з порядком прийняття в експлуатацію об'єктів, побудованих без дозволу на виконання спеціальних будівельних робіт. Такі дані опублікувала Державна служба статистики України.

В будинках, з двома й більше квартирами, прийнято в експлуатацію 56,0% від загального об'єма житла. В одноквартирних будинках цей показник дорівнює 43,7 и 0,3% від загального обсягу житла. В 2019 загальна площа прийнятого в експлуатацію житла зменшилась на 15,2% у порівнянні з 2015 роком.

В містах було прийнято в експлуатацію 6502,9 тис. кв. м. чи 69,4% від загальної площі житла. Для сільської місцевості цей показник складає 2863,9 тис. кв. м. чи 30,6%.

Загальна динаміка ринку нерухомості Києва, на приклад, має наступний вигляд: середня вартість кв. м. (як на первинному ринку, так і на вторинному) весь останній час (протягом останніх 4 років) знижалась, а ось вартість оренди залишилась сталою. В середньому по Києву просять за 1 кв. м. житла близько 1250 доларів США [38].

За даними Державної статистики України, в 2019 році загалом було сдано 112576 квартир, загальною площею 9024,5 тис. кв. м. Середній розмір квартири у кв. м. складає 80,2. Тому можна виокремити окрему тенденцію: Найпопулярнішим «товаром» на ринку є квартира в місті, приблизно двох кімнатна.

Загалом обсяг будівництва складає близько 60 млрд грн, з них будівництво житлових приміщень – 13,5 млрд, нежитлових – 14,4 млрд, ще 27 млрд – будівництво інженерних споруд [40].

Незважаючи на падіння курсу гривні, певну нестабільність, ринок нерухомості в України може бути одним з найбільш капіталізованих та впевнених, що є цілком обґрунтовано. Отримана інформація про ситуацію на ринку нерухомості в регіоні може допомогти скласти більш цілісну картинку поточної ситуації, оцінити місто загалом як об’єкт нерухомості, що в результаті дозволить більш якісно спрогнозувати ціни окремої будівлі, адже ми матимемо краще уявлення як конкретно про ситуацію в регіоні, так і про стан ринку нерухомості загалом.

1.5 Методологія застосування даних про нерухомість в моделях машинного навчання

Для якісного застосування математичних методів та правильного моделювання й прогнозування важливим є знання прикладної галузі, де власне ці методи застосовуються. У даному пункті будуть розглянуті підходи й методологія аналізу ринку нерухомості, його особливості.

Перш за все варто провести аналіз, базуючись на власній логіці та емпіричних спостереженнях. Зрозуміти, числові значення якого діапазону повинні виходити для того чи іншого поля, придивитись до кожної змінної й спробувати зрозуміти її значення для даної проблеми. Якщо мова йде про обробку великих таблиць та баз даних із сотнями полів, ми можемо створити список змінних, де для зручності будемо занотовувати інформацію [7, 8]. Наш список (таблиця) матиме наступні колонки:

- *Змінна* – назва змінної;
- *Тип* – ідентифікатор типу даних. Є два можливих значення цього поля: «числовий» (значення є числами) та «категоріальний» (значення є приналежністю змінної до певної категорії);
- *Сегмент* – ідентифікатор сегменту змінної. Ми намагаємось емпірично розбити вибірку на декілька сегментів (скільки ми вважаємо необхідним) та відносимо кожну змінну до певного сегменту. Наприклад, для об’єкта нерухомості можна виокремити такі сегменти: будівля, простір та локація. Під «будівлею» ми маємо на увазі змінні, які відповідають фізичним характеристика об’єкта. Під «простором» ми розуміємо змінні, які кажуть про масштаб об’єкта, його площу й т.п. Під «локацією» ми маємо на увазі змінні, які містять інформацію про розташування об’єкта, сусідів і т.д.;

- *Очікування* – наші очікування від впливу змінної на значення прогнозованої величини. Ми можемо використовувати категоріальні змінні «Низька», «Середня», «Висока» як можливі варіанти;

- *Заключення* – наші кінцеві уявлення про важливість змінної. Можемо використовувати таку ж шкалу оцінок, як і для очікування.

Якщо з визначенням значення типу й сегменту вся прямо й зрозуміло (просто віднести змінну до тієї чи іншої категорії), то очікування більш суб'єктивна а комплексна характеристика. Ми повинні розробити певне дерево рішень, алгоритм, задавати питання й залежно від відповіді віддавати змінній той чи інший пріоритет.

Наприклад, для об'єкта нерухомості алгоритм може виглядати наступним чином:

1. Чи інформація вже описувалась частково чи повністю в якійсь іншій змінній, тобто чи існує між ними колінеарність. Якщо так, то відносимо змінній «низьке» очікування.

2. Чи турбуємось ми про значення даної змінної, коли самі купуємо будинок. Наприклад, чи настільки важливим для нас є тип кладки при виборі будівлі нашої мрії, чи можливе це не перша необхідність. Якщо змінна не є важливою для нас, сміливо віддаємо їх «середнє» очікування.

3. Якщо дана змінна і є важливою в принципі, то наскільки важливою є різниця значень даної змінної. Наприклад, якість зовнішнього оздоблення для нас є важливим, але наскільки критично є те, що воно не «відмінне», а «добре». Якщо відмінності у значеннях не є критичними, то відносимо «високе» очікування.

Отже, у даному розділі було поставлено методологію аналізу ринку нерухомості, зокрема описано що таке ринок нерухомості, виокремлено основні впливи на показники цін на нерухомість, також проаналізовано динаміку й закономірності ринку нерухомості України за 2004-2020 рр.

Далі необхідно визначити методи й підходи які будуть використовуватися для прогнозування цін на нерухомість, підготувати базу даних для застосування у прогнозуванні, за допомогою обраних методів побудувати моделі, обрати найкращу та на її основі створити інформаційне забезпечення.

РОЗДІЛ 2
КОНЦЕПТУАЛІЗАЦІЯ ІНФОРМАЦІЙНОГО ЗАБЕЗПЕЧЕННЯ ПРОГНОЗУВАННЯ ЦІН НА НЕРУХОМІСТЬ МЕТОДАМИ МАШИННОГО НАВЧАННЯ

2.1 Формальне представлення прогнозування цін на нерухомість для використання комп'ютерними системами

Ринок нерухомості сучасної України складає центральну ланку всієї системи ринкових відносин. Поведінка ринку нерухомості характеризується постійними змінами збільшення або зменшення вартості квадратного метра нерухомості залежно від значень багатьох факторів, більшість з яких нечіткі і навіть суперечать один одному. Більше того, така залежність апріорно невідома. Тому прогнозування вартості нерухомості є складним, в якому зацікавлені як продавці, так і покупці [16].

За різними джерелами, для оцінки вартості нерухомості використовується близько 20 параметрів, таких як: місце розташування, плани поверхів, площа, підлога, тип будівлі, кількість поверхів, наявність парковки, якість оздоблення, відстань від метро, перехрестя, екологічний стан тощо. Залежно від конкретної ситуації деякі з них суперечать один одному, мають більшу вагу або не зрозумілі.

Спробуємо формально представити оцінку вартості нерухомості. Наприклад, ціна одиниці площі окремого об'єкта нерухомості C_0 в даний момент часу *t* складається з двох компонентів [48]:

$$C_0(t, l_i) = P(g_1, g_2, \dots, g_n, t) + P_0(l_1, l_2, \dots, l_i, \dots, l_m), \qquad (2.1)$$

де l_i- вектор локальних чинників,

t – одиниця часу.

Функція $P(g_1, g_2, \dots, g_n, t)$ описує вплив глобальних макроекономічних факторів і являє собою загальний рівень цін в місті (регіоні), єдиний для всіх

об'єктів у певний момент часу. Величини $P_0(l_1, l_2, \dots, l_i, \dots, l_m)$ являють собою внесок локальних відмінностей. Вони різні для кожного об'єкта і залежать від набору його характеристик.

Ціну житла у загальному вигляді пропонується представити як функцію:

$$C_e = f(S, K, F, R, P), \tag{2.2}$$

де C_e – ціна квадратного метра житла,

S – собівартість будівництва житла,

K – комплексний показник характеристик міста,

F – чинники попиту і пропозиції,

R – ризики інвестування,

P – прибуток будівельної організації.

У свою чергу собівартість житлового будівництва описується наступною залежністю [56]:

$$S = f(Z_r, Z_i, Z_b, Z_o, N), \tag{2.3}$$

де S – собівартість будівництва квадратного метра житла,

Z_r – витрати на придбання (оренди) земельної ділянки,

Z_i – витрати на влаштування інжерених комунікацій,

Z_b – витрати на виконання будівельно-монтажних робіт,

Z_o – непередбачені та інші витрати,

N – податки, які входять у собівартість.

2.2 Методи підготовки бази даних для прогнозування

Підготовка даних («data preparation», «data preprocessing») – це процес трансформації необроблених даних, завдяки чому вчені та аналітики даних можуть запускати їх за допомогою алгоритмів машинного навчання, щоб розкрити ідеї або зробити прогнози [33].

Більшість алгоритмів машинного навчання вимагають форматування даних у дуже конкретному вигляді, тому набори даних зазвичай вимагають певної підготовки, перш ніж вони зможуть дати корисну інформацію. Деякі набори даних мають значення, які відсутні, є недійсними або іншими способами важкі для обробки алгоритмом. Якщо дані відсутні, алгоритм не може їх використовувати. Якщо дані недійсні, алгоритм дає менш точні або навіть оманливі результати. Деякі набори даних є відносно чистими, але їх потрібно формувати (наприклад, агрегувати або обертати), а багатьом наборам даних просто бракує корисного бізнес-контексту (наприклад, погано визначені значення ідентифікаторів), отже, необхідність збагачення функцій. Хороша підготовка даних дає чіткі та добре підготовлені дані, що призводить до більш практичних, точних результатів моделі.

Процес підготовки даних може ускладнюватися такими проблемами, як [18]:

1. *Пропущені або неповні спостереження.* Ця проблема виникає через те, що важко отримати кожну точку даних для кожного запису в наборі даних. Відсутні дані іноді відображаються як порожні клітинки, значення (наприклад, NULL або N/A) або певний символ, наприклад знак питання. Необхідно проганяти набір даних задля заповнення таких значень.

2. *Викиди або аномалії*. Викид - це точки даних, яка суттєво відрізняється від інших спостережень. На цьому кроці потрібно буде визначити наявні викид та спробувати зрозуміти, чому вони містяться в даних. Залежно від того, чому вони містяться в даних, можна видалити їх із набору даних або зберегти. Існує кілька способів ідентифікувати відхилення: Z-score/standard deviations або Interquartile Range (IQR).

3. *Неправильно відформатовані / структуровані дані.* Необхідно з’ясувати з яким типом даних ми будемо працювати - чи вони кількісні, категоріальні, тощо. Це особливо важливо для цільової змінної, оскільки тип даних звужує, яку модель машинного навчання ми будемо використовувати та

впливає на сам результат прогнозування. Тут будуть корисні функції Pandas, такі як df.describe () та df.dtypes.

4. *Нормалізація даних*. Коли числові значення мають різні виміри, більшість алгоритмів машинного навчання погано працюють. Алгоритм k-найближчих сусідів є найкращим прикладом, коли функції з різними масштабами не працюють добре. Таким чином, нормалізація або стандартизація даних може допомогти у вирішенні цієї проблеми.

Усі вищенаведені пункти необхідно перевірити у власному датасеті, та при наявності будь-яких відхилень або проблем позбутися від них для їхнього ефективного використання та прогнозування у подальшому.

2.3 Формування математичного апарату прогнозування цін на нерухомість

У даному пункті будуть виокремлені й проаналізовані основні математичні моделі, які покладені в основу нашої моделі машинного навчання та на основі яких ми будемо аналізувати й прогнозувати ринок нерухомості. Будуть розглянуті окремі математичні складові кожної моделі та сформовані моделі у цілому.

2.3.1 Регресійний аналіз

Першим й найпростішим методом буде застосування **лінійної регресії**. Лінійна регресія – метод моделювання залежності між скаляром y та векторною змінною X. У випадку, якщо X також є скаляром, регресію називають простою. Загалом лінійна регресія це лінійна функція, загальна модель якої визначається у виді [18]:

$$y = \beta_0 + \beta_1 x_1 + \cdots + \beta_k x_k + u, \tag{2.4}$$

де y – залежна змінна,

$(x_1, x_2, \dots, x_k)$ – вектор незалежних змінних;

$(\beta_0, \beta_1, \dots, \beta_k)$ – вектор параметрів;

u – випадкова похибка, розподіл якої в загальному випадку залежить від незалежних змінних, але математичне сподівання якої рівне нулю.

Задача лінійної регресії полягає у оцінці вектора параметрів на основі деяких експериментальних значень y та $(x_1, x_2, \dots, x_k)$.

Лінійна регресія проста в застосуванні та інтуїтивно зрозуміла, але, очевидно, для вирішення поставленої задачі вона є дуже примітивною. Тому розглянемо між складні та комплексні види регресії [15].

Регресія за методом найменших квадратів (коли ми намагаємось мінімізувати суми квадратів відхилення функції від шуканих змінних) часто може бути нестійкою, тобто сильно залежною від навчальних даних, що зазвичай призводить до перенавчання моделі [15]. Запобігти цьому допомагає **регуляризація**, сенс якої в «стягуванні» моделі у ході налаштування вектору коефіцієнтів β таким чином, щоб вони в середньому стали менше по абсолютній величині, ніж це було при оптимізації методом найменших квадратів. Також наша вибірка має аж близько 100 незалежних змінних, дуже вірогідно, що серед них присутня мультиколінеарність – наявність лінійної залежності між двома або більше факторними змінними у регресивній моделі. Для усунення описаних особливостей будемо використовувати рідж-регресію, так звану регуляризацію (додавання деякої додаткової інформації, щоб знайти рішення некоректно сформованої задачі) Тихонова, або як її ще називають, гребеневу регресію – один з методів пониження розмірності, що допоможе усунути погану обумовленість матриці X^TX і нестійкість оцінок коефіцієнтів регресії.

Розглянемо СЛР $Ax=b$, обумовленість матриці показує наскільки матриця близька до матриці неповного рангу (для квадратних матриць – до виродженості), тобто якщо матриця A погано обумовлена, то навіть незначні зміни у b (навіть зміни близькі до 0.001%) спричинюють вагомі зміні x. Оцінки, наприклад, можуть

мати неправильний знак або значення, які істотно переважають ті, які неприйнятні з емпіричних уявлень. Застосування гребеневої регресії нерідко виправдовується тим, що за допомогою нього при правильного використання можна отримати менше значення середнього квадрату помилки. Метод варто використовувати, якщо: а) сильна обумовленість; б) сильно різнятся власні значення або деякі з них близькі до нуля; в) в матриці X є майже лінійно залежні стовпці [14].

Іншими словами, нехай ми маємо звичайну лінійну регресію, описану вище. Ми намагаємось підібрати такі коефіцієнти β_i, які мінімізують похибку, тобто мінімізують вираз (метод найменших квадратів) [10]:

$$L = \sum_{i=1}^{n} (y_i - \hat{y})^2 \tag{2.5}$$

Суть методу **рідж-регресії** полягає в тому, щоб накласти на похибку штраф, тобто додати певний вираз, щоб зменшити величини коефіцієнтів β_i та запобігти надмірність комплексності моделі , й задачі вже буде мінімізувати наступний вираз [30]:

$$L = \sum_{i=1}^{n} (y_i - \hat{y})^2 + \lambda \times \sum_{i=1}^{n} \widehat{\beta_i^2} \tag{2.6}$$

Параметр λ у даному виразі підбирається самим користувачем, і суть завдання для користувача при застосування рідж-регресії полягає у тому, щоб підібрати такий штраф λ, який би мінімізував похибку.

Для демонстрації покладемо $\widehat{y_i} = \hat{\beta} \times x_i$. Методом найменших квадратів, при підстановці попереднього виразу й прирівнюючи похідну до нуля (щоб знайти найменше значення), отримаємо:

$$\hat{\beta} = \frac{\sum_{i=1}^{n} x_i y_i}{\sum_{i=1}^{n} x_i^2} \tag{2.7}$$

При використанні рідж-регресії при мінімізації похибки ми також повинні враховувати штраф, тобто нам потрібно мінімізувати похибку, яка складалася б з 2х: похибки найменших квадратів та штрафу. У результаті отримаємо:

$$\hat{\beta} = \frac{\sum_{i=1}^{n} x_i y_i}{\sum_{i=1}^{n} x_i^2 + \lambda} \tag{2.8}$$

Тобто, як ми бачимо з попередніх прикладів, введення штрафу (та його збільшення) призводить до зменшення значень коефіцієнтів лінійної регресії.

Іншим, схожим методом є **LASSO регресія**. Даний метод застосовується при схожих умовах, відмінність є те, що ми вже накладаємо штраф на суму модулів оцінених коефіцієнтів $|\beta|$, тобто лассо-регресія вже ставить задачу мінімізувати наступний вираз [29]:

$$L = \sum_{i=1}^{n} (y_i - \hat{y})^2 + \lambda \times \sum_{i=1}^{n} |\widehat{\beta}_\iota| \tag{2.9}$$

У нашому дослідженні ми також будемо використовувати **Elastic Net моделі** для побудови інформаційного забезпечення. Метод Elastic Net вперше з'явився в результаті критики щодо LASSO регресії, вибір змінних якого може бути занадто залежним від даних і, отже, нестабільним. Рішення полягає в штрафів рідж та LASSO регресій, щоб отримати найкращий прогноз з обох моделей. Elastic Net націлений на мінімізацію наступної функції втрат [19]:

$$L_{enet}(\hat{\beta}) = \frac{\sum_{i-1}^{n} (y_i - x_i' \hat{\beta})^2}{2n} + \lambda \left(\frac{1-\alpha}{2} \sum_{j-1}^{m} \hat{\beta}_j^2 + \alpha \sum_{j-1}^{m} |\hat{\beta}_j| \right), \tag{2.10}$$

де α - параметр змішування між рідж-регресією (α = 0) та LASSO регресією (α = 1).

Отже, якщо лінійна модель містить багато змінних-предикторів або якщо ці змінні корелюють, стандартні оцінки параметрів OLS мають велику дисперсію, що робить модель ненадійною.

Щоб запобігти цьому, необхідно використовувати регуляризацію - техніку, яка дозволяє зменшити цю дисперсію ціною введення деякого біасу. Знаходження оптимального значення цього параметру дозволяє мінімізувати загальну похибку моделі.

Ми описали три популярні методи регуляризації, кожен з яких спрямований на зменшення розміру коефіцієнтів [27]:

- Рідж-регресія, яка вводить штрафи для суми квадратних коефіцієнтів (штраф L2);
- LASSO регресія, яка вводить штраф суми абсолютних значень коефіцієнтів (штраф L_1);
- Elastic Net модель - комбінація Рідж-регресії та LASSO.

Значення відповідних штрафів може бути підібраний шляхом перехресної перевірки, щоб знайти найкращу модель для прогнозування.

2.3.2 Нормальний розподіл та його властивості

Як правило, машинне навчання застосовується на великому наборі даних. По-перше, немає сенсу використовувати складні регресійні техніки для вибірки, яка має одновимірний лінійний розподіл. Майже всі спостереження містять в собі сотні атрибутів і полів. Результатом спостереження є сума багатьох випадкових слабо взаємозалежних величин, кожна з яких вносить малий вклад відносно загальної суми. Розподілом, який найкраще описує задану ситуацію, є **нормальний розподіл**, який також називають розподілом Гауса, що характеризується густиною ймовірності [28]:

$$f(x;\ \mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}} exp\ (-\frac{(x-\ \mu)^2}{2\sigma^2}) \qquad (2.11)$$

де μ – математичне сподівання (коефіцієнт зсуву), σ^2 – дисперсія випадкової величини (коефіцієнт масштабу).

Розподіл із $\mu = 0$ та $\sigma^2 = 1$ називається стандартним нормальним розподілом.

Логнормальний розподіл – це розподіл, логарифм якої має нормальний розподіл. Більшість значень, які характеризуватимуть об'єкт нерухомості, є випадковими. Об'єкт може мати як 20 квадратних метрів площі, так і 200. Але різниця для невеликих величин матиме більший зміст, ніж для великих [31]. Наприклад, оцінка будинків площею 30 кв.м. та 60 кв.м. буде відрізнятися більше, ніж будинків 230 кв.м. та 260 кв.м., тому у цьому випадку більш доцільним є використання логнормального розподілу, де функція густини [50]:

$$f(x;\ \mu, \sigma) = \frac{1}{x\sigma\sqrt{2\pi}} exp\ (-\frac{(ln\ ln\ x\ -\ \mu)^2}{2\sigma^2}) \qquad (2.12)$$

Побудовані моделі необхідно перевіряти, наскільки точний прогноз вони дають. Для цього будемо використовувати перехресну перевірку – метод оцінювання достовірності математичної моделі з метою перевірки, наскільки результати статистичного аналізу узагальнюються на незалежному наборі даних. Переважно даний метод використовується саме в задачах прогнозування для оцінку точності моделі. Одноразова перехресна перевірка передбачає розбиття вибірки на взаємодоповнювані підвибірки з метою проведення аналізу на одній частині (що називається навчальним набором) і перевірки аналізу на іншій частині (що називається контрольним або тестовим набором) [31]. Для зниження дисперсії здійснюється багаторазова перехресна перевірка із застосуванням різних розбиттів, і результати цих перевірок усереднюються.

Для більш детального дослідження вибірки на наявність нормального розподілу використовують **коефіцієнти асиметрії та ексцесу** [49]. Асиметрією γ_1 розподілу ймовірностей випадкової величини називають відношення центрального моменту третього порядку μ_3 до куба середнього квадратичного відхилення σ^3:

$$\gamma_1 = \frac{\mu_3}{\sigma^3}, \tag{2.13}$$

де μ_3 – центральний момент третього порядку,

σ^3 – дисперсія.

Асиметрія додатня, якщо «довша частина» вибірки знаходиться праворуч від математичного сподівання, і навпаки, якщо «довша частина» знаходиться праворуч, то асиметрія від'ємна.

Коефіцієнт ексцесу γ_2 характеризує «крутість», тобто стрімкість підвищення кривої розподілу у порівнянні з нормальною кривою. Коефіцієнт обчислюється за формулою [46]:

$$\gamma_2 = \frac{\mu_4}{\sigma^4} - 3, \tag{2.14}$$

де μ_4 – центральний момент четвертого порядку,

σ^2 – дисперсія.

Для нормального розподілу $\frac{\mu_4}{\sigma^4} = 3$, із чого випливає що ексцес нормального розподілу дорівнює нулю. Якщо коефіцієнт відмінний від нуля, то крива щільності не збігається з кривою нормального розподілу та має вищу, більш «гострішу» вершину при додатному ексцесі, якщо ексцес від'ємний, то має нижчу, більш «плоскішу» вершину [47].

2.3.3 Композиція алгоритмів за допомогою дерев рішень

Якщо попередні техніки є варіаціями та вдосконаленням регресії, то наступний метод використовує дещо інший підхід, в основі якого лежать **дерева рішень**. Дерева ухвалення рішень (також можуть називатися деревами класифікації або регресійними деревами) використовуються в статистиці та аналізі даних для прогнозованих моделей. Структура дерева містить елементи: «листя» і «гілки». На ребрах («гілках») дерева ухвалення рішення записані атрибути, від яких залежить цільова функція, в «листі» записані значення цільової функції, а в

інших вузлах - атрибути, за якими розрізняються випадки. Щоб класифікувати новий випадок, треба спуститися по дереву до листа і видати відповідне значення. Подібні дерева рішень широко використовуються в інтелектуальному аналізі даних. Мета полягає в тому, щоб створити модель, яка прогнозує значення цільової змінної на основі декількох змінних на вході [55]. Кожен лист являє собою значення цільової змінної, зміненої в ході руху від кореня по листа. Кожен внутрішній вузол відповідає одній з вхідних змінних. Дерево може бути також «вивчено» поділом вихідних наборів змінних на підмножини, що засновані на тестуванні значень атрибутів. Це процес, який повторюється на кожному з отриманих підмножин. Рекурсія завершується тоді, коли підмножина у вузлі має ті ж значення цільової змінної, таким чином, воно не додає цінності для пророкувань.

Регулювання глибини дерева – це техніка, яка дозволяє зменшувати розмір дерева рішень, видаляючи ділянки дерева, які мають маленьку вагу. Одне з питань, який виникає в алгоритмі дерева рішень – це оптимальний розмір кінцевого дерева. Так, невелике дерево може не охопити ту чи іншу важливу інформацію щодо вибіркового простору. Тим не менше, важко сказати, коли алгоритм повинен зупинитися, тому що неможливо спрогнозувати, додавання якого вузла дозволить значно зменшити помилку [6]. Ця проблема відома як «ефект горизонту». Тим не менш, загальна стратегія обмеження дерева зберігається, тобто видалення вузлів реалізується в разі, якщо вони не дають додаткової інформації. Необхідно зазначити, що регулювання глибини дерева повинно зменшити розмір навчальної моделі дерева без зменшення точності її прогнозу або за допомогою перехресної перевірки.

Навчання на деревах рішень використовують їх як моделі для прогнозування значення об’єкта (представленого в «листях») використовуючи дані, отримані при спостереженні об’єкта (представленого в «гілках»).

Поширеною технікою машинного навчання, яка опирається на дерева рішень, є **градієнтний бустінг** – техніка, яка використовується для вирішення задач класифікації та регресії, результатом якої є модель прогнозування у формі ансамбля (збірки) декількох простих моделей, зазвичай саме дерев рішень. Градієнтний бустінг – один с найкращих способів, націлених на побудову композиції. Ми будемо будувати композицію наступного вигляду [30]:

$$a_N(x) = \sum_{n=1}^{N} b_n(x), \tag{2.15}$$

де a_N – композиція з N базових алгоритмів, b_n – базовий алгоритм.

Ми не усереднюємо, а сумуємо алгоритми, оскільки кожен наступний коректує помилки попереднього. Також будемо вважати, що ми маємо певну функцію втрат *L (y, z)*, яка вимірює значення помилки для одного об'єкта, прикладом функції може бути звичайна функція методу найменших квадратів, яка була описана вище.

Побудову моделі ми починаємо з ініціалізації, будуємо перший базовий алгоритм $b_0(x)$, який не повинен бути надто важким, це може бути найпростіша функція виду $b_0(x) = 0$ (де на виході ми будемо отримувати константу), або середня відповідь по всій навчальній вибірці [36]:

$$b_0(x) = \frac{1}{l}\sum_{i=1}^{l} y_i. \tag{2.16}$$

Будемо дійти методом індукції й покладемо, що ми вже побудували *N-1* алгоритмів (для *N=1* це буде означати, що ми побудували лише початковий алгоритм $b_0(x)$). Наше завдання – зрозуміти, яким повинен бути наступний навчальний алгоритм $b_n(x)$. Задача буде виглядати так [36]:

$$\sum_{i=1}^{l} L(y_i, a_{N-1}(x_i) + b_i(x_i)) \rightarrow min. \tag{2.17}$$

Ми сумуємо втрати на всій навчальній вибірці: суми вже побудованої структури $a_{N-1}(x_i)$ й нового алгоритм у $b_i(x_i)$, й будемо намагатися вибрати останній таким чином, щоб мінімізувати помилку композиції.

Для початку спростимо собі задачу й спробуємо дати відповідь на питання які значення наш новий алгоритм повинен приймати на об’єктах навчальної вибірки [36]:

$$\sum_{i=1}^{l} L(y_i, a_{N-1}(x_i) + s_i) \rightarrow min, \quad (2.18)$$

s_i – зсув прогнозу на *i*-му об’єкті.

Отже, ми отримаємо наступну задачу оптимізації. Нам потрібно знайти такий вектор $s = (s_1, s_2, \dots, s_l)$, який буде мінімізувати дану функцію [35]:

$$F(s) = \sum_{i=1}^{l} L(y_i, a_{N-1}(x_i) + s_i) \rightarrow min. \quad (2.19)$$

Вектор, який якнайбільше зменшує функцію, це антиградієнт, оскільки він направлений в сторону найшвидшого зменшення функції. Отже:

$$s = -\nabla F = (-L'_z(y_1, a_{N-1}(x_1)), \dots, -L'_z(y_l, a_{N-1}(x_l))), \quad (2.20)$$

де $-L'_z(y_l, a_{N-1}(x_l))$ – зсув по *l*-му об’єкту.

Отже, ми вже зрозуміли, як саме необхідно зсунути прогнози вже побудованої композиції, щоб зменшити значення функції втрат. Ми будемо налаштовувати наступний алгоритм, $b_N(x)$ так, щоб він був якомога ближче до зсувів s_i і близкість будемо вимірювати за допомогою середньоквадратичного відхилення. Функціонал даної задачі буде виглядати так [36]:

$$b_N(x) = argmin \frac{1}{l} \sum_{i=1}^{l} (b(x_i) - s_i)^2. \quad (2.21)$$

До того ж, вся інформація про функцію витрат L міститься в зсувах, градієнті.

На кінцевому етапі, після знайдення алгоритму $b_N(x)$, ми додаємо його до композиції.

Модель XGBoost означає Extreme Gradient Boosting - це конкретна реалізація методу градієнтного бустінга, яка використовує більш точну апроксимацію даних для пошуку найкращої моделі дерева рішень [32]. Цей метод використовує низку

чудових прийомів, які роблять його надзвичайно успішним, особливо зі структурованими даними. Найважливішими є:

1. Обчислення градієнтів другого порядку, тобто других часткових похідних функції втрат (подібно до методу Ньютона), що надає більше інформації про напрямок градієнтів і про те, як дійти до мінімуму функції втрат. Хоча регулярне посилення градієнта використовує функцію втрат базової моделі (наприклад, дерева рішень) як проксі-сервер для мінімізації помилок загальної моделі, XGBoost використовує похідну 2-го порядку як наближення.

2. Вдосконалена регуляризація (L1 та L2), що покращує узагальнення моделі.

XGBoost моделювання також має такі додаткові переваги, як збільшена швидкість навчання моделі і його можна розподілити / розподілити між кластерами.

У Розділі 3 ми перевіримо який з методів буде найбільш ефективним для прогнозування цін на нерухомість.

2.4 Реалізація математичного апарату прогнозування цін на нерухомість у комп'ютерних системах

Звичайно, для обчислень великих обсягів даних та вирахування результату, а головне, високої швидкості обчислень, необхідне використання комп'ютерних обчислювальних машин. Для обробки даних в принципі існує багато інструментів та технологій, але найпопулярнішими є Python та R. Яку мову використовувати повністю залежить від користувача, математичні та статистичні методи, описані вище, реалізовані в обох середовищах. В дані роботі буде розглянута робота в Python, але все те ж саме, за аналогічним принципом реалізовано і в R.

При роботі нам знадобляться:

- Pandas – для більш легкого опрацювання даних;

- Matplotlib – для візуалізації;
- NumPy та SciPy – для наукових розрахунків;
- Seaborn – для візуалізації статистичних даних;
- Sklearn – бібліотека машинного навчання, яка містить всі вищеописані техніки (крім градієнтного бустінгу);
- Xgboost – для градієнтного бустінгу.

Далі будуть розглянуті основні принципи на алгоритми роботи функції зі сторони користувача.

1. Метод *sklearn.model_selection.cross_val_score*

```
cross_val_score(estimator, X, y=None, groups=None, scoring=None, cv=None,
                n_jobs=1, verbose=0, fit_params=None,
                pre_dispatch='2*n_jobs'):
```

Метод приймає у якості аргументів наступні параметри:

estimator – об'єкт, який перевіряється на правильність даних;

X – дані, які повинні відповідати;

y – дані, які намагаємся передбачити;

scoring – метод перевірки, у нашому випадку середньоквадратична похибка;

cv – визначає стратегію розбиття (кількість розбиттів).

2. Клас *sklearn.linear_model.Ridge* для моделювання гребневої регресії, параметри якого мають наступний вигляд:

```
def __init__(self, alpha=1.0, fit_intercept=True, normalize=False,
             copy_X=True, max_iter=None, tol=1e-3, solver="auto",
             random_state=None):
```

alpha – сила регуляризації. Більше значення відповідає сильнішій регуляризації. По факту, єдиний параметр, маніпуляції з яким можуть змінити результат.

3. Клас *sklearn.linear_model.LassoCV* для моделювання лассо-регресії, параметри наступні:

```
def __init__(self, eps=1e-3, n_alphas=100, alphas=None, fit_intercept=True,
             normalize=False, precompute='auto', max_iter=1000, tol=1e-4,
             copy_X=True, cv=None, verbose=False, n_jobs=1,
             positive=False, random_state=None, selection='cyclic'):
```

alphas – список альф для обрахунку моделі. Інші параметри, як і в моделі гребневої регресії, не несуть великої цінності;

Параметри *xgboost* зрозумілі інтуїтивно з їх назви та не повинні викликати питань (аналогічні вищезгаданим). В цілому структура програмних проектів є прозорою і чіткою, повна документація може бути знайде на веб-ресурсі відповідного програмного пакету.

Отже, у цьому розділі було концептуалізовано основні методі та підходи для вирішення задачі прогнозування цін на нерухомість. По-перше, були вивчені та виокремлені теоретичні основи побудови моделей регресії, їх математична реалізація та логіка. Також розглянуті теоретично інші, більш складні поняття й підходи машинного навчання які можуть бути використані для моделювання ринку нерухомості, серед яких регуляризація, гребенева регресія, регресія LASSO, Elastic Net регресія та методи градієнтного бустінгу.

Крім того, були описані основні аспекти й надана інформація по реалізації даних методів й моделей за допомогою комп’ютерних систем, а саме середовища мови програмування Python та допоміжних бібліотек.

РОЗДІЛ 3
ПОБУДОВА МОДЕЛЕЙ ПРОГНОЗУВАННЯ ЦІН НА НЕРУХОМІСТЬ З ВИКОРИСТАННЯМ ІНФОРМАЦІЙНОГО ЗАБЕЗПЕЧЕННЯ

3.1 Аналіз бази даних та створення системи прогнозування цін на нерухомість

У даному розділі буде проаналізована база об’єктів нерухомості невеликого містечка, з населенням близько 50 тис. жителів та розташуванням відносно далеко від великих центрів. У містечку переважає приватна забудова, поверховість у більшості випадків складає до двох поверхів. База даних складається з 2917 спостережень та 81 змінної, з яких 36 є кількісними, 43 категоріальними, а також змінна Id (номер змінної) та «Ціна продажі» («SalePrice») [9]. Детальний опис всіх бази даних наведений в Додатку А, опис атрибутів (змінних) в Додатку Б.

Для проведення дослідження поділимо дані на дві частини – тренувальну та тестову. Поділ на частини відбувається порівну: половина рядків знаходиться в першій частині, інша – в другій, обидві частини мають однакові поля (стовпці). Відмінність між ними у тому, перша частина має стовпчик «Ціна продажі» (яка є цільовою змінною, котру ми будемо намагатися спрогнозувати), а в другій він відсутній. Нам необхідно встановити (передбачити) ціну об’єктів нерухомості другої частини, базуючись на першій частині.

Нульовим кроком необхідно підключити необхідне програмне забезпечення. Як вже зазначалось, ми будемо використовувати мову програмування Python працюючи в Jupyter Notebook. Керівництво з встановлення Python знаходиться на офіційному сайті й передбачає лише декілька базових кроків. Встановлення середовища Jupyter Notebook відбувається в одну стрічку в терміналі, інформація також може бути знайдена на офіційному сайті середовища.

Підключаємо необхідні бібліотеки (див. пункт 1.3) для роботи з даними та візуалізації:

```
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
import matplotlib.pyplot as plt  # Matlab-style plotting
import seaborn as sns
from scipy.stats import norm, skew #for some statistics
from scipy import stats
```

Завантажуємо необхідний датасет для роботи [9] та розархівовуємо його:

```
import zipfile

path_to_zip_file = "C:/Users/Julia/Desktop/LabsPython/house-prices-dataset.zip"
with zipfile.ZipFile(path_to_zip_file, 'r') as zip_ref:
    zip_ref.extractall()
```

Та перевіряємо наявність необхідних даних в директорії:

```
import os

path = "C:/Users/Julia/Desktop/LabsPython/"

with os.scandir(path) as listOfEntries:
    for entry in listOfEntries:
        print(entry.name)
```

```
.ipynb_checkpoints
data_description.txt
house-prices-dataset.zip
PythonNotebook.ipynb
sample_submission.csv
test.csv
train.csv
```

Як описано вище, ми маємо 2 набора даних: *train.csv* (тренувальна вибірка) та *test.csv* (тестова вибірка). Перший набір даних буде використовуватися для побудови й навчання моделі, виявлення певних закономірностей та залежностей, побудови алгоритму. Він містить в собі вже ключове поле «Ціна продажі» («SalePrice»). Другий набір даних вже буде використовуватися для тестування отриманої моделі й оцінки її точності за допомогою середньоквадратичної

логарифмічної похибки. Цей набір даних вже не має поля «Ціна продажі», його нам і потрібно знайти та спрогнозувати.

Виведемо перші п'ять рядів тренувального датасету.

```
train.head(5)
```

	Id	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandContour
0	1	60	RL	65.0	8450	Pave	NaN	Reg	Lvl
1	2	20	RL	80.0	9600	Pave	NaN	Reg	Lvl
2	3	60	RL	68.0	11250	Pave	NaN	IR1	Lvl
3	4	70	RL	60.0	9550	Pave	NaN	IR1	Lvl
4	5	60	RL	84.0	14260	Pave	NaN	IR1	Lvl

Виведемо перші п'ять рядів тестового датасету.

```
test.head(5)
```

	Id	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandContour
0	1461	20	RH	80.0	11622	Pave	NaN	Reg	Lvl
1	1462	20	RL	81.0	14267	Pave	NaN	IR1	Lvl
2	1463	60	RL	74.0	13830	Pave	NaN	IR1	Lvl
3	1464	60	RL	78.0	9978	Pave	NaN	IR1	Lvl
4	1465	120	RL	43.0	5005	Pave	NaN	IR1	HLS

Було вирішено видалити стовпчик «Id» з об’єднаного датасету, так як він не несе необхідної інформації для прогнозування. Перевіримо знову розмірність даних після видалення змінної «Id».

```
print("\nThe train data size after dropping Id feature is : {} ".format(train.shape))
print("The test data size after dropping Id feature is : {} ".format(test.shape))
```

```
The train data size after dropping Id feature is : (1460, 80)
The test data size after dropping Id feature is : (1459, 79)
```

Отже, було проаналізовано та запущено наш датасет, який буде використовуватися для подальшого прогнозування.

3.2 Підготовка даних для побудови моделей та прогнозування

Для подальшої роботи з завантаженим набором даних нам необхідно його підготувати. Розіб'ємо підготовку даних на два етапи – підготовка цільової змінної та підготовка атрибутів бази даних.

3.2.1 Підготовка цільової змінної «Ціна продажі» для моделювання

Щоб підготувати цільову змінну «Ціна продажі», нам необхідно перевірити її на наявність нульових значень та викидів, а також нормалізувати.

Нульові значення. Для початку перевіримо, чи не має перший набір даних *train.csv*, нульових значень в ключовому полі «Ціна продажі», наявність яких би негативно впливала на результат:

```
train['SalePrice'].describe()
```

```
count      1460.000000
mean     180921.195890
std       79442.502883
min       34900.000000
25%      129975.000000
50%      163000.000000
75%      214000.000000
max      755000.000000
Name: SalePrice, dtype: float64
```

Мінімальне значення дорівнює 34900 > 0, відповідно набір даних не має нульових значень в «Ціні продажу».

Викиди. Далі проаналізуємо цільову змінну «Ціна продажі» на наявність викидів. Для цього побудуємо графік залежності ціни продажі від площі будинку («GrLivArea»).

```
fig, ax = plt.subplots()
ax.scatter(x = train['GrLivArea'], y = train['SalePrice'])
plt.ylabel('SalePrice', fontsize=13)
plt.xlabel('GrLivArea', fontsize=13)
plt.show()
```

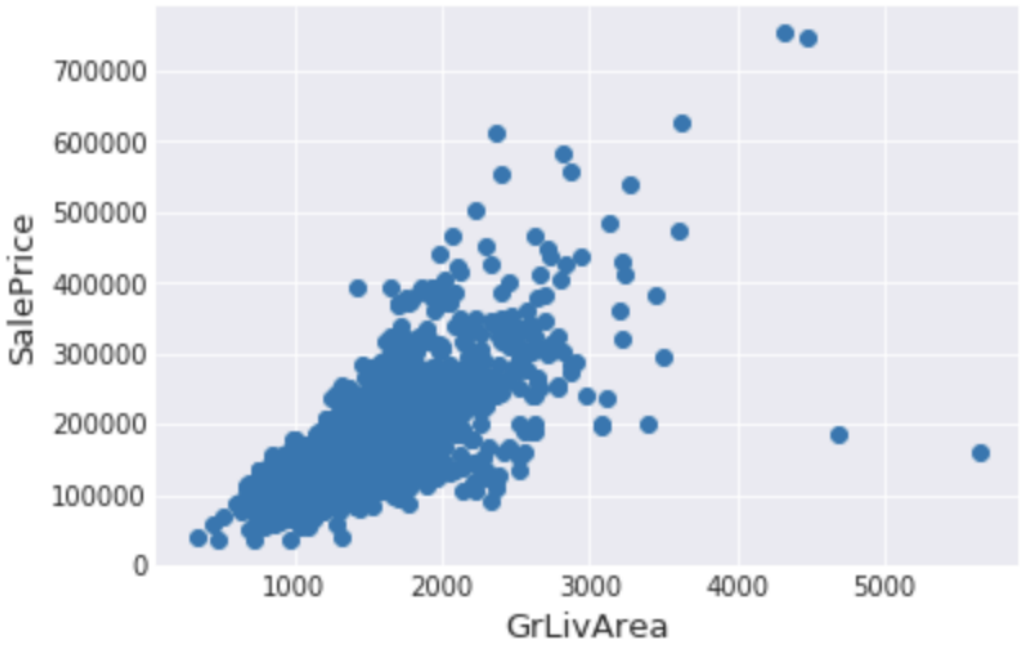

Рисунок 3.1 – Графік залежності ціни продажі «SalePrice» від площі будинку «GrLivArea».

З рис. 3.1 видно, що внизу праворуч наявні два викиди, які мають низьку ціну за відносно велику площу, що логічно неможливо. Тож, в даному випадку, можемо їх безпечно видалити.

```
train = train.drop(train[(train['GrLivArea']>4000) & (train['SalePrice']<300000)].index)
```

Побудуємо графік без викидів (рис. 3.2)

```
fig, ax = plt.subplots()
ax.scatter(train['GrLivArea'], train['SalePrice'])
plt.ylabel('SalePrice', fontsize=13)
plt.xlabel('GrLivArea', fontsize=13)
plt.show()
```

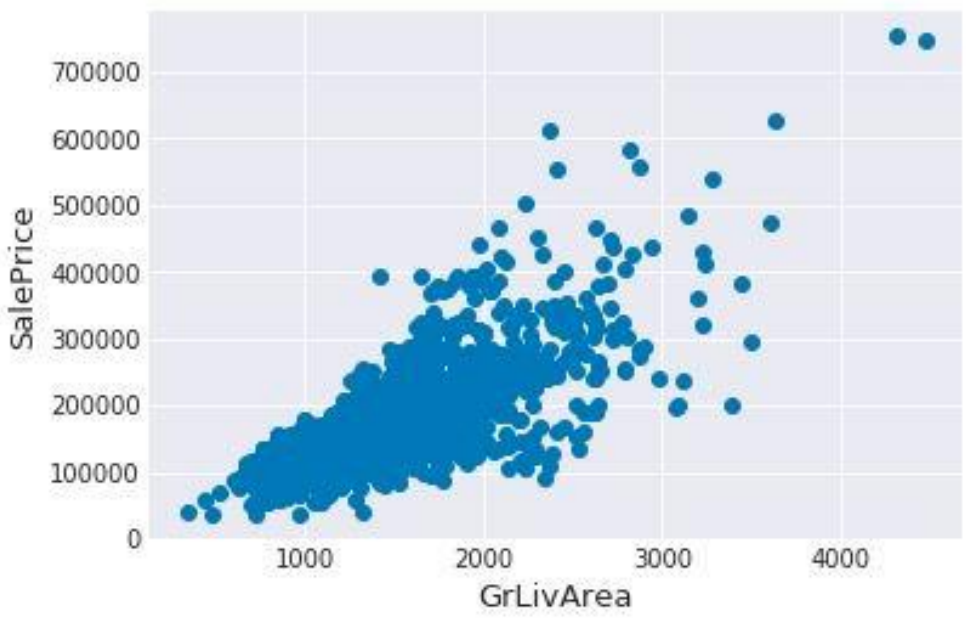

Рисунок 3.2 – Графік залежності ціни продажі «SalePrice» від площі будинку «GrLivArea» без викидів.

Нормалізація цільової змінної. Для початку порівняємо наш ряд «Ціна продажі» з тренувального датасету з нормальним розподілом, побудувавши гістограму розподілу (рис. 3.3):

```
sns.distplot(train['SalePrice'] , fit=norm);

#Обчислимо параметри mu та sigma
(mu, sigma) = norm.fit(train['SalePrice'])
print( '\n mu = {:.2f} and sigma = {:.2f}\n'.format(mu, sigma))

#Завершимо графік розподілу
plt.legend(['Normal dist. ($\mu=$ {:.2f} and $\sigma=$ {:.2f} )'.format(mu, sigma)],
            loc='best')
plt.ylabel('Frequency')
plt.title('SalePrice distribution')
```

Та графік ймовірностей (QQ-plot) (рис. 3.3):

```
fig = plt.figure()
res = stats.probplot(train['SalePrice'], plot=plt)
plt.show()
```

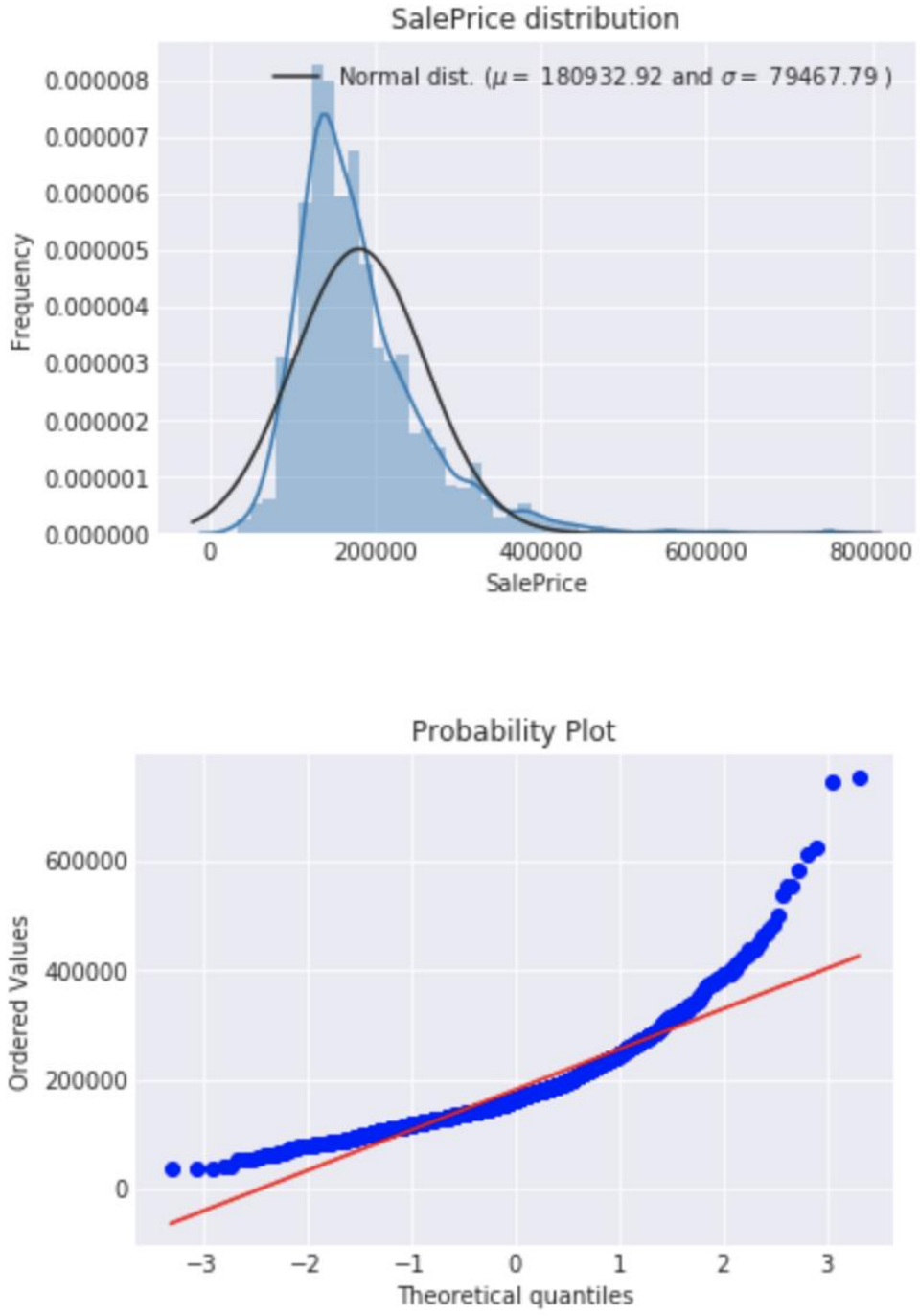

Рисунок 3.3 – Порівняння графіків розподілу даного набору даних й нормального розподілу.

З рис. 3.3 видно що розподіл цільової змінної:

- подібний до нормального, але дещо відхиляється від нього;
- має позитивний коефіцієнт асиметрії;
- демонструє гостровершинність.

Давайте тепер більш детально розглянемо два останніх пункти й власне визначимо коефіцієнти асиметрії та ексцесу (гостровершинності).

```
print("Асиметрія: %f" % train['SalePrice'].skew())
print("Екцес: %f" % train['SalePrice'].kurt())
```

```
Асиметрія: 1.882876
Екцес: 6.536282
```

Обидва коефіцієнти далекі від нульового значення, що доводить відсутність нормального розподілу в даній вибірці. Оскільки прогнозні моделі люблять нормально розподілені дані, нам потрібно нормалізувати цю змінну.

Для нормалізації застосуємо доволі цікавий «трюк» – логарифмічну трансформацію: *log (price+1)*.

```
train['SalePrice'] = np.log1p(train['SalePrice'])
```

Побудуємо ще одну гістограму, вже з набором даних, до якої була застосована логарифмічна нормалізація (рис. 3.4):

```
sns.distplot(train['SalePrice'], fit=norm);
fig = plt.figure()
res = stats.probplot(train['SalePrice'], plot=plt)
```

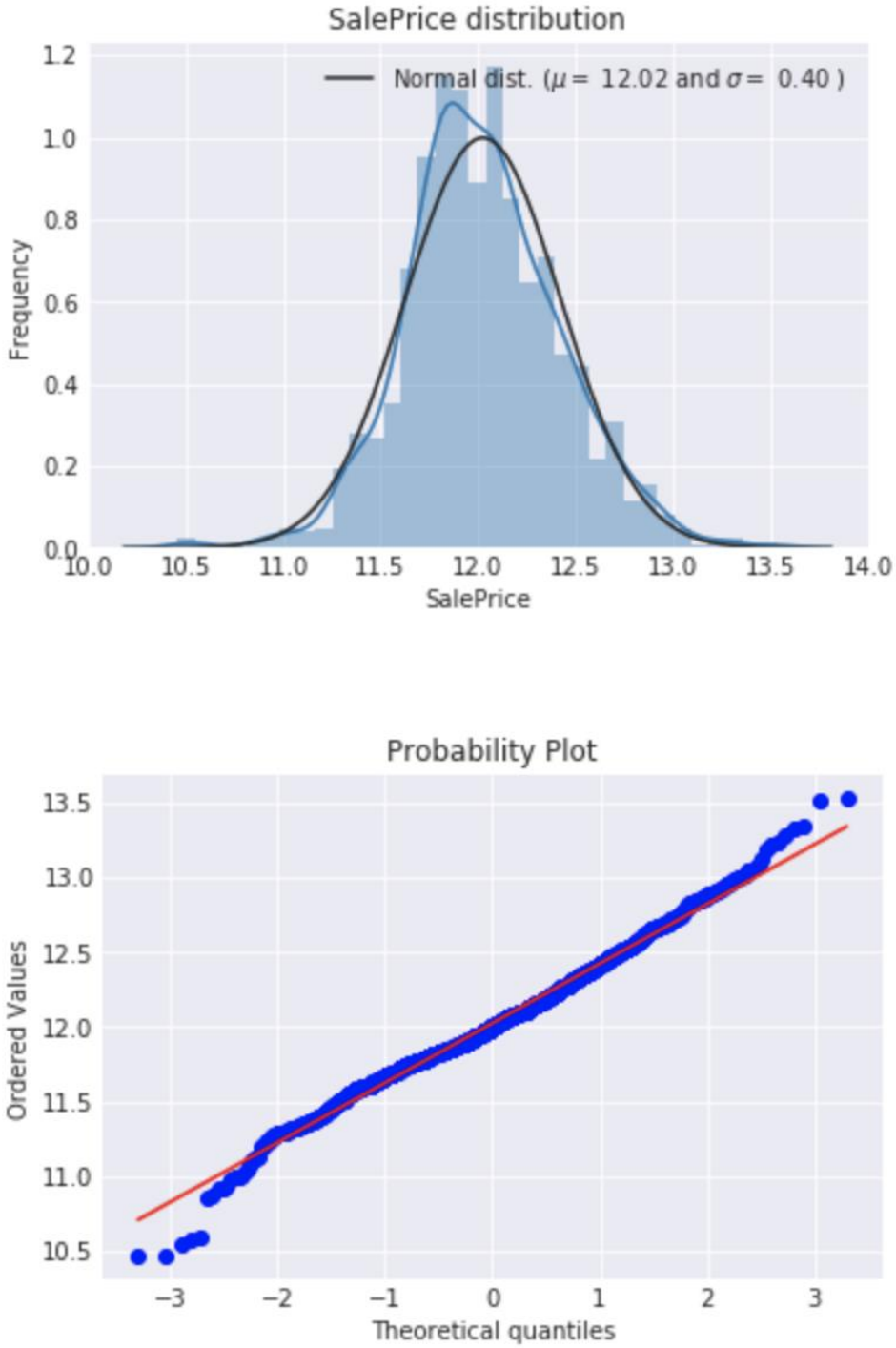

Рисунок 3.4 – Порівняння графіків розподілу логарифмічно нормалізованого набору даних й нормального розподілу.

Тож, «Ціна продажі» тепер схожа на нормально розподілену, позбавлена від нульових значень, викидів та асиметрії.

3.2.2 Підготовка атрибутів бази даних

Для більш ефективної обробки атрибутів, об'єднаємо тестовий та тренувальний датасети в один дата фрейм.

```
ntrain = train.shape[0]
ntest = test.shape[0]
y_train = train.SalePrice.values
all_data = pd.concat((train, test)).reset_index(drop=True)
all_data.drop(['SalePrice'], axis=1, inplace=True)
print("all_data size is : {}".format(all_data.shape))
```

```
all_data size is : (2917, 79)
```

Щоб підготувати атрибути для подальшого прогнозування, нам необхідно також перевірити їх на наявність пропущених значень, позбавити ряди від асиметрії, нормалізувати їх та перевірити на мультиколінеарність.

Пропущені значення. Для того, щоб виявити пропущені значення в базі даних, обчислимо відсоток цих значень по кожній змінній.

```
all_data_na = (all_data.isnull().sum() / len(all_data)) * 100
all_data_na = all_data_na.drop(all_data_na[all_data_na == 0].index).sort_values(ascending=False)[:30]
missing_data = pd.DataFrame({'Missing Ratio' :all_data_na})
missing_data.head(20)
```

Маємо наступний результат, наведений на рис. 3.5.

19 атрибутів мають відсутні значення, 5 понад 50% усіх даних. Найчастіше НС означає відсутність предмета, описаного за атрибутом, наприклад, відсутність басейну, огорожі, відсутність гаража та підвалу. Детальні значення по кожній змінній наведені а Додатку В.

Для позбавлення від пропущених значень використаємо такий підхід, як імітація пропущених значень. Для цього, необхідно пройтися по кожній змінній,

яка має пропущені значення та присвоїти цим пропущеним значенням дані відповідно до їхнього формату.

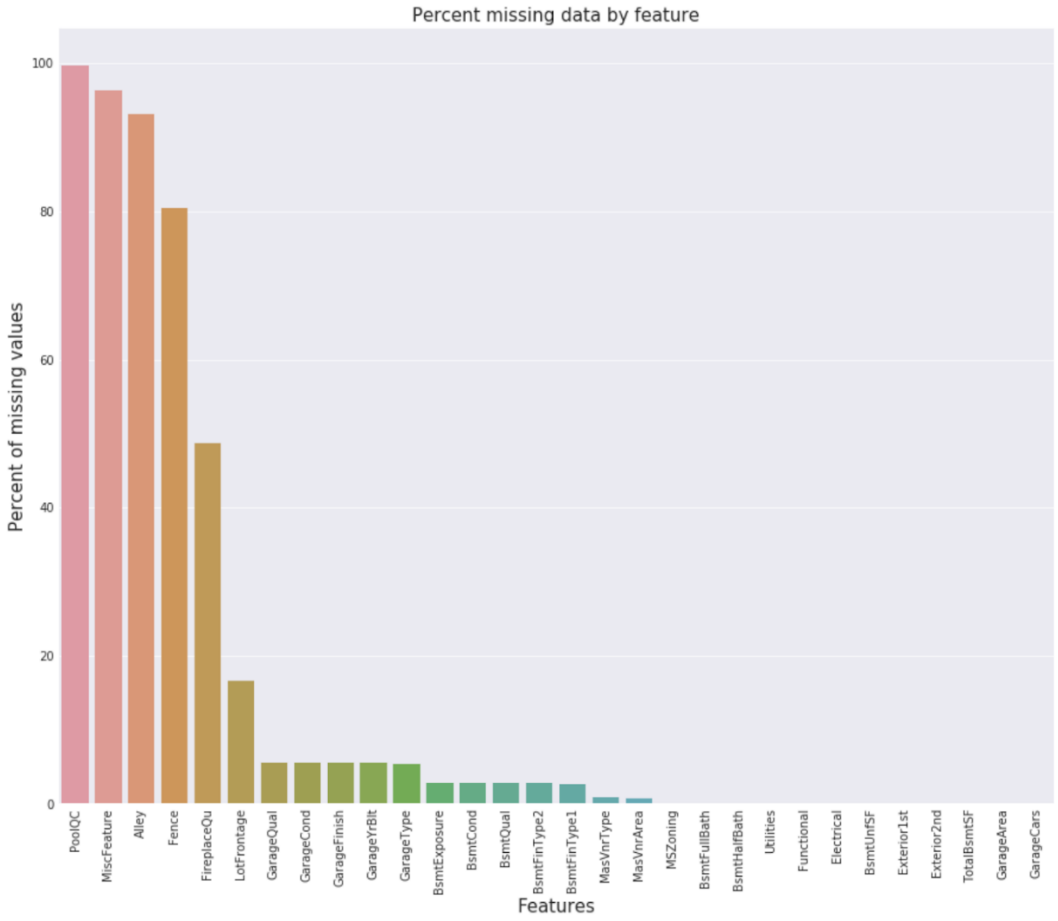

Рисунок 3.5 – Відсоток пропущених значень у деяких змінних.

Наприклад, змінна *PoolQC (pool quality)* має найбільший відсоток пропущених значень - більше 99%. В описі даних [9] описано, що NA для даної змінної означає «Без басейну». Це має сенс та говорить про те, що більшість будинків взагалі не мають басейну. Замінимо пропущені значення для всього ряду.

```
all_data["PoolQC"] = all_data["PoolQC"].fillna("None")
```

Пройдемося так по кожній змінній з рис. 3.5. Знову розрахуємо відсоток пропущених значень:

```
all_data_na = (all_data.isnull().sum() / len(all_data)) * 100
all_data_na = all_data_na.drop(all_data_na[all_data_na == 0].index).sort_values(ascending=False)
missing_data = pd.DataFrame({'Missing Ratio' :all_data_na})
missing_data.head()
```

	Missing Ratio

Пропущених значень немає.

Позбавлення від асиметрії. Асиметрія - міра симетрії розподілу. Симетричний набір даних матиме асиметрію рівну 0. Таким чином, і нормальний розподіл матиме асиметрію 0. Асиметрія по суті вимірює відносний розмір двох хвостів. Як правило, перекіс повинен становити від -1 до 1. У цьому діапазоні дані вважаються досить симетричними.

Спочатку виявимо у яких саме рядах наявна асиметрія в розподілі (табл. 3.1).

```
numeric_feats = all_data.dtypes[all_data.dtypes != "object"].index

# Перевірка викривлених числових атрибутів
skewed_feats = all_data[numeric_feats].apply(lambda x: skew(x.dropna())).sort_values(ascending=False)
print("\nSkew in numerical features: \n")
skewness = pd.DataFrame({'Skew' :skewed_feats})
skewness.head(10)
```

Таблиця 3.1 – Наявність асиметрії в деяких змінних

Variable	**Skew**
MiscVal	21.939672
PoolArea	17.688664
LotArea	13.109495
LowQualFinSF	12.084539
3SsnPorch	11.372080

Продовження таблиці 3.1

LandSlope	4.973254
KitchenAbvGr	4.300550
BsmtFinSF2	4.144503
EnclosedPorch	4.002344
ScreenPorch	3.945101

Застосуємо метод «Box Cox Transformation» для атрибутів з високою асиметрією. Використаємо scipy функцію boxcox1p, яка обчислює перетворення Box-Cox 1+x. Параметр λ = 0 еквівалентний log1p, який ми використовували вище для нормалізації цільової змінної.

```
skewness = skewness[abs(skewness) > 0.75]
print("There are {} skewed numerical features to Box Cox transform".format(skewness.shape[0]))

from scipy.special import boxcox1p
skewed_features = skewness.index
lam = 0.15
for feat in skewed_features:
    #all_data[feat] += 1
    all_data[feat] = boxcox1p(all_data[feat], lam)
```

```
There are 59 skewed numerical features to Box Cox transform
```

Нормалізація атрибутів. Для нормалізації даних було також застосовано логарифмічну трансформацію, як і для ряду «Ціна продажі».

Мультиколінеарність. Перевіримо чи існує залежність між нашими рядами даних. Нас буде цікавити дві речі - залежність «Ціни продажі» від інших полів та залежність полів між собою (для виявлення мультиколінеарності). Для цього побудуємо мапу залежності між полями (рис. 3.6).

```
corr_matrix = train.corr()
f, ax = plt.subplots(figsize=(12, 9))
sns.heatmap(corr_matrix, vmax=.8, square=True)
```

З отриманого результату (рис. 3.6) видно, шо ключове поле «Ціна продажі» сильно залежна від полів «Загальна оцінка» (що дуже передбачувано, адже дано поле описує стан і якість будинку загалом), «Жила площа» (що є також цілком прогнозованим), «Площа фундаменту» (іншими словами площа основи будинку), та характеристиками прибудинкового гаража: «Площа гаража», «Кількість авто» та ін., що вже є не таким й передбачуваним, адже на першу думку гараж не є ключовою характеристикою об’єкта нерухомості.

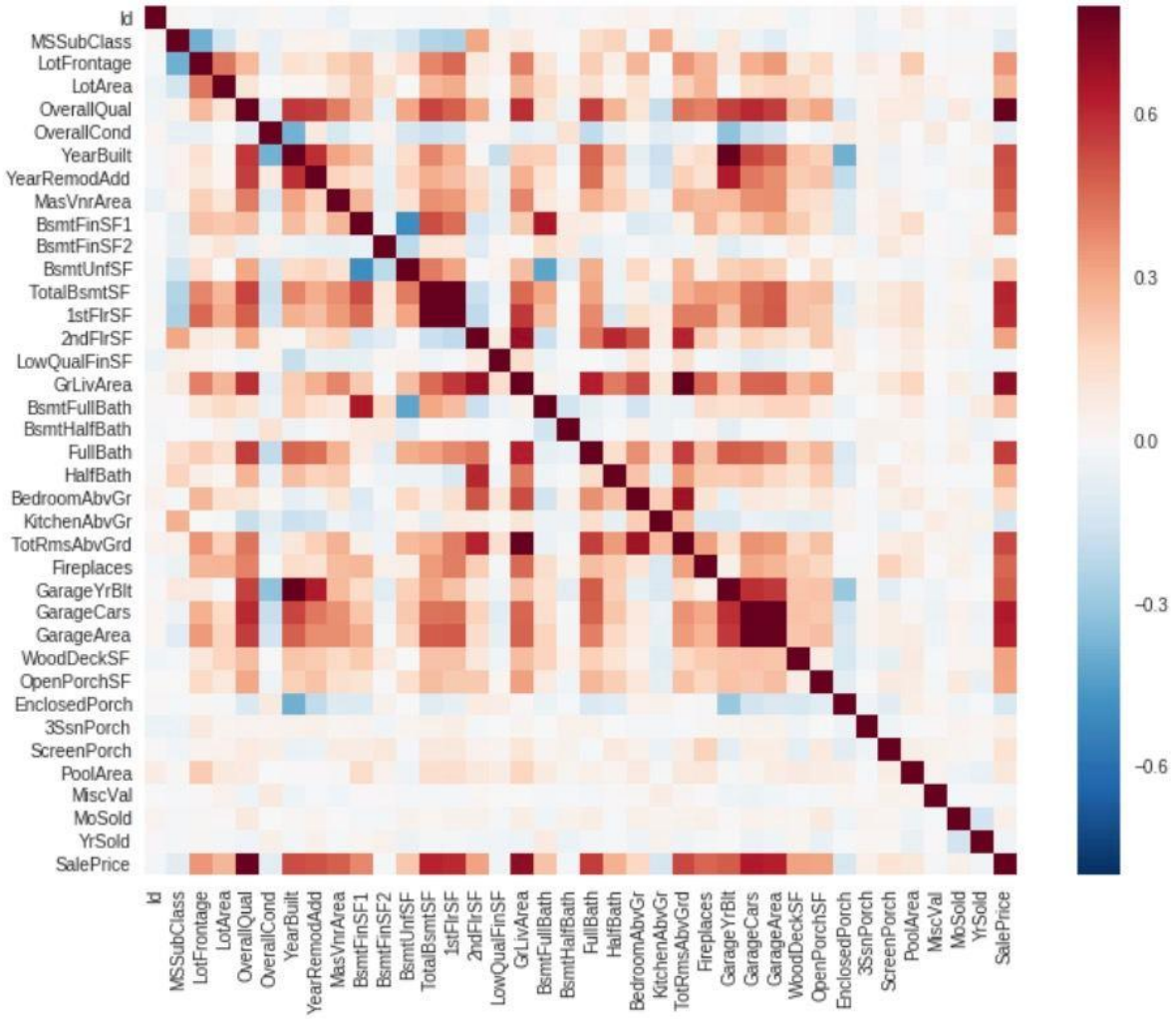

Рисунок 3.6 – Мапа залежностей між полями.

Мультиколінеарність однозначно присутня й з нею необхідно боротися. Сильно колінеарними є поля «Площа фундаменту» і «Площа першого поверху», що зовсім не дивно, адже в більшості варіантів ці поля мають ідентичні значення. Також колінеарними є поля «Рік побудови» (будівлі) та «Рік побудови гаража», що є також очевидно, адже у більшості випадків гараж будується в одну чергу з будинком.

Тож, наші атрибути та цільова змінна були підготовлені для подальшого прогнозування, зокрема дані були нормалізовані, позбавлені від викидів та нульових значень, позбавлені від мультиколінеарності.

3.3 Застосування моделей регуляризації Lasso, Ridge та Elastic Net

Як зазначалося в пункті 1.3, для побудови прогнозних моделей будемо використовувати бібліотеку *Sklearn.* Для початку створюємо матриці для *Sklearn* (матриці, на основі яких буде будуватися й навчатися майбутня модель):

```
x_train = all_data[:train.shape[0]]
x_test = all_data[train.shape[0]:]
y = train.SalePrice
```

Зараз ми збираємось використовувати впорядковані лінійні моделі регресії з регуляризацією (regularized linear regression models) з модуля *scikit learn*. Ми будемо використовувати обидві: l_1 (Lasso) і l_2 (Ridge) впорядкованості. Також я визначу функцію, яка буде повертати перехресно перевірену середньоквадратичну похибку (cross-validation rmse error), так що ми зможемо оцінити наші моделі й обрати найкращу.

```
from sklearn.linear_model import Ridge, RidgeCV, ElasticNet, LassoCV, LassoLarsCV
from sklearn.model_selection import cross_val_score

def rmse_cv(model):
    rmse= np.sqrt(-cross_val_score(model, x_train, y, scoring="neg_mean_squared_error", cv = 5))
    return(rmse)
```

```
model_ridge = Ridge()
```

Головний параметр для Ridge моделі – альфа – параметр впорядкованості, який показує, наскільки гнучкою наша модель є. Чим краще модель впорядкована, тим менше вона є схильною до перенавчання.

```
alphas = [0.05, 0.1, 0.3, 1, 3, 5, 10, 15, 30, 50, 75]
cv_ridge = [rmse_cv(Ridge(alpha = alpha)).mean()
            for alpha in alphas]
```

```
cv_ridge = pd.Series(cv_ridge, index = alphas)
cv_ridge.plot(title = "Validation - Just Do It")
plt.xlabel("alpha")
plt.ylabel("rmse")
```

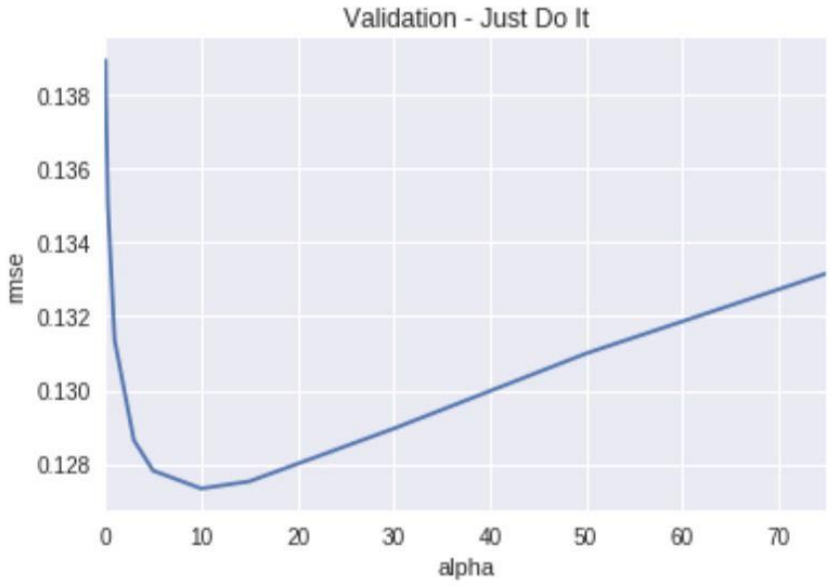

Рисунок 3.7 – U-подібна крива перевірки.

Давайте розглянемо більш детально рис. 3.5. Коли альфа занадто велика, впорядкованість сильніша й модель не може захопити всі особливості й нюанси в даних. З іншого боку, коли ми дозволяємо моделі бути більш гнучкою (альфа

мале), модель починає перенавчатися. Значення альфа = 10 є оптимальним, що видно з рисунку вище.

```
score = rmsle_cv(KRR)
print("Kernel Ridge score: {:.4f} ({:.4f})\n".format(score.mean(), score.std()))
```

RMSE похибка для моделі Ridge складає приблизно 0.1153, стандартне відхилення - 0.0075.

Продовжимо дослідження побудовою моделі Elastic Net:

```
ENet = make_pipeline(RobustScaler(), ElasticNet(alpha=0.0005, l1_ratio=.9, random_state=3))
```

Разрахуємо RMSE похибку для даної моделі:

```
score = rmsle_cv(ENet)
print("ElasticNet score: {:.4f} ({:.4f})\n".format(score.mean(), score.std()))
```

RMSE похибка моделі Elastic Net становить 0.1116, стандартне відхилення - 0.0074. Бачимо, що дана модель впоралася краще за Ridge регресію.

Спробуємо побудувати модель Лассо та дізнатися її похибку та стандартне відхилення:

```
lasso = make_pipeline(RobustScaler(), Lasso(alpha =0.0005, random_state=1))
```

```
score = rmsle_cv(lasso)
print("\nLasso score: {:.4f} ({:.4f})\n".format(score.mean(), score.std()))
```

RMSE похибка для моделі Лассо складає 0.1115, стандартне відхилення - 0.0074. Бачимо, що Лассо модель демонструє більш кращий результат за Ridge регресію (похибка менше на 3,4%) та приблизно однаковий з моделлю Elastic Net. Було вирішено використовувати дану модель для передбачення на тестовому наборі даних. Ще одна заслуга Лассо в тому, що він виконує проектування ознак

за нас – назначає коефіцієнти ознак, які він вважає непотрібними, до нуля. Давайте більш детально подивимось на отримані коефіцієнти:

```
coef = pd.Series(model_lasso.coef_, index = x_train.columns)
```

Лассо обрало 111 змінних й виключило інші 177 змінні. Спробуємо запустити Лассо декілька раз на само навантажених (bootstrapped) зразках, коли поточний зразок я виходом попередньої ітерації, тобто ми проганяємо Лассо декілька раз на одному й тому ж зразку, враховуючи вихід попередньої ітерації.

Також давайте більш детально подивимось на найважливіші коефіцієнти:

```
imp_coef = pd.concat([coef.sort_values().head(10),
                      coef.sort_values().tail(10)])
```

```
matplotlib.rcParams['figure.figsize'] = (8.0, 10.0)
imp_coef.plot(kind = "barh")
plt.title("Коэффициенты в моделе Лассо")
```

Найбільш важливою позитивною ознакою, як ми бачимо з рис. 3.8, є GrLivArea – «Жила площа». Стовідсотково це має зміст. Потім йдуть декілька ознак місцерозташування. Деякі з негативних ознак не мають особливого сенсу – скоріш за все, причиною цього є незбалансовані критичні змінні.

Рисунок 3.8 – Коефіцієнти змінних у моделі Лассо.

Давайте подивимось на точкову діаграму передбачення і похибки:

```
matplotlib.rcParams['figure.figsize'] = (6.0, 6.0)

preds = pd.DataFrame({"передбачення":model_lasso.predict(x_train), "true":y})
preds["похибка"] = preds["true"] - preds["передбачення"]
preds.plot(x = "передбачення", y = "похибка",kind = "scatter")
```

На виході ми матимемо графік (рис. 3.9):

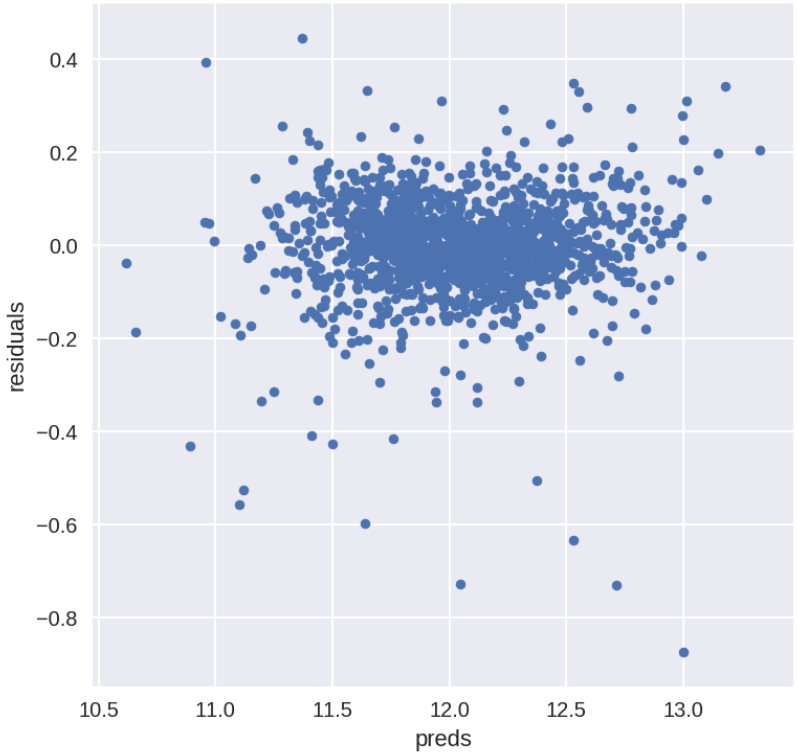

Рисунок 3.9 – Точкова діаграма передбачення і похибки.

Отже, було побудовано 3 моделі регуляризації – Lasso, Ridge та Elastic Net. Найкращий результат отримано за допомогою Lasso регресії.

3.4 Використання градієнтного бустінгу та XGBoost моделі для прогнозування

Для побудови моделі Extreme Gradient Boosting (*xgboost)* додаємо її до нашого робочого середовища:

```
import xgboost as xgb
```

Визначаємо тестову й тренувальну вибірки як спеціальний об’єкт бібліотеки *xgboost.* Надаємо відповідні параметри в модель.

```
dtrain = xgb.DMatrix(x_train, label = y)
dtest = xgb.DMatrix(x_test)

params = {"max_depth":2, "eta":0.1}
model = xgb.cv(params, dtrain,  num_boost_round=500, early_stopping_rounds=100)
```

Вираховуємо середньоквадратичні похибки для тестової й тренувальної вибірок:

```
model.loc[30:,["test-rmse-mean", "train-rmse-mean"]].plot()
```

На виході отримаємо рис. 3.10, на якому видно, що значення похибки на тренувальній вибірці становиться сталою вже після 150 перевірок.

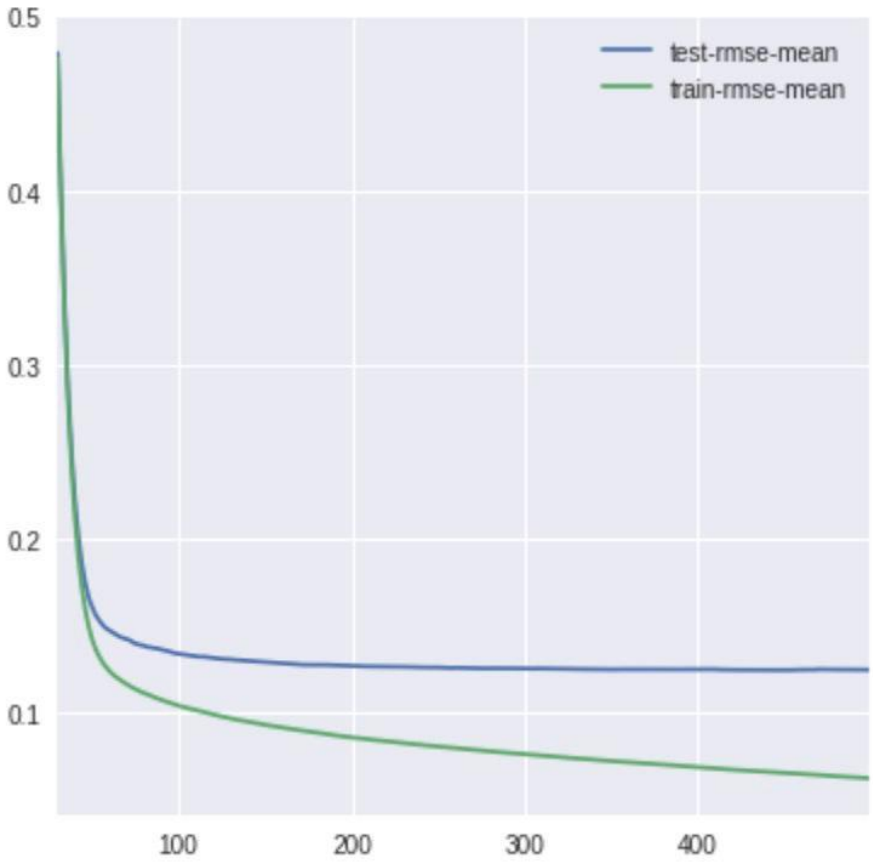

Рисунок 3.10 – Середньоквадратичні похибки для тестової й тренувальної вибірок.

Тепер власне створюємо модель з відповідними параметрами й намагаємось натренувати її:

```
model_xgb = xgb.XGBRegressor(n_estimators=360, max_depth=2, learning_rate=0.1)
#параметри були налаштовані використовуючи xgb.cv
model_xgb.fit(x_train, y)
```

```
XGBRegressor(base_score=0.5, colsample_bylevel=1, colsample_bytree=1, gamma=0,
       learning_rate=0.1, max_delta_step=0, max_depth=2,
       min_child_weight=1, missing=None, n_estimators=360, nthread=-1,
       objective='reg:linear', reg_alpha=0, reg_lambda=1,
       scale_pos_weight=1, seed=0, silent=True, subsample=1)
```

Разрахуємо RMSE похибку для XGBoost моделі:

```
score = rmsle_cv(model_xgb)
print("Xgboost score: {:.4f} ({:.4f})\n".format(score.mean(), score.std()))
```

RMSE похибка становить 0.1161, стандартне відхилення - 0.0079.

Для порівняння побудуємо модель звичайного градієнтного бустінгу *(gboost)*:

```
GBoost = GradientBoostingRegressor(n_estimators=3000, learning_rate=0.05,
                                   max_depth=4, max_features='sqrt',
                                   min_samples_leaf=15, min_samples_split=10,
                                   loss='huber', random_state =5)
```

Разрахуємо RMSE похибку для даної моделі:

```
score = rmsle_cv(GBoost)
print("Gradient Boosting score: {:.4f} ({:.4f})\n".format(score.mean(), score.std()))
```

RMSE похибка моделі градієнтного бустингу становить 0.1177, стандартне відхилення - 0.0080. Можемо зробити висновок, що модель Extreme Gradient Boosting має кращий результат прогнозування нащої цільової змінної, порівняно зі звичайним градієнтним бустингом.

3.5 Створення агрегованої моделі та порівняння результатів

Ми знаходимось на фінальному етапі нашого дослідження. Спробуємо створити додатково 2 агреговані моделі щоб подивитися чи зможемо мі покращити результат прогнозування на тестових даних.

Перша агрегована модель. Для першої агрегованої моделі візьмемо Лассо модель та модель Extreme Gradient Boosting (так як вони показали найкращі результати серед серед 2-х застосованих методів), порівняємо їх результати (рис. 3.11).

```
xgb_preds = np.expm1(model_xgb.predict(x_test))
lasso_preds = np.expm1(model_lasso.predict(x_test))
```

```
predictions = pd.DataFrame({"xgb":xgb_preds, "lasso":lasso_preds})
predictions.plot(x = "xgb", y = "lasso", kind = "scatter")
```

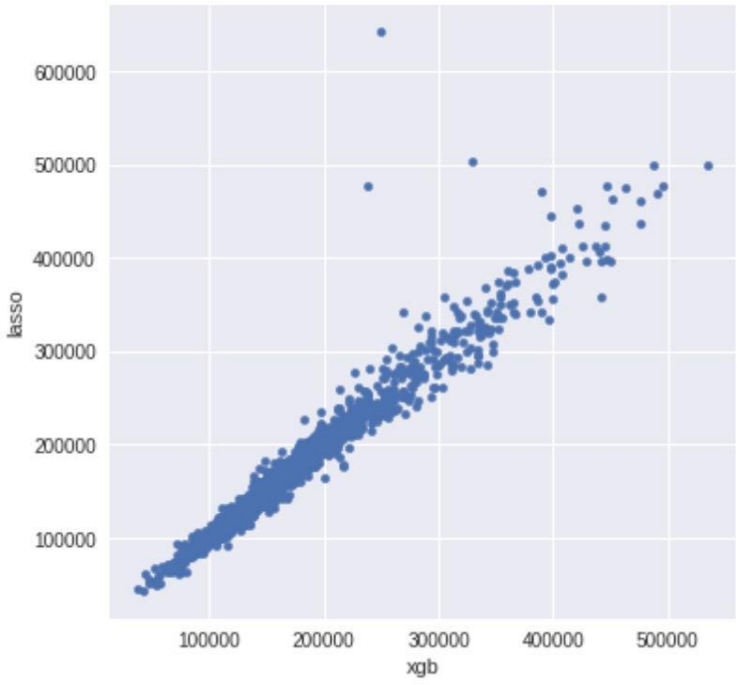

Рисунок 3.11 – Порівняння результатів LASSO та градієнтного бустінгу.

Має зміст взяти зважене середнє некореляційних результатів – це зазвичай зменшує похибку, хоча конкретно в нашому випадку це не сильно допомагає. Додамо Так як LASSO показала себе дещо краще, то додамо до неї дещо вищий ваговий коефіцієнт.

```
preds = 0.7*lasso_preds + 0.3*xgb_preds
```

Розрахуємо кінцеві передбачення для тестової вибірки:

```
solution = pd.DataFrame({"id":test.Id, "SalePrice":preds})
solution.to_csv("output.csv", index = False)
```

Обчислена RMSE похибка для першої агрегованої моделі складає 0.1184. Цей результат не є найкращим серед побудованих моделей.

Давайте побудуємо *другу агреговану модель*.

Як ми бачили з попередніх розрахунків, найменш точний прогноз дала модель звичайного градієнтного бустінгу, тому ми спробували побудувати агреговану модель прогнозування ціни об'єкту нерухомості без неї. Тобто, для другої агрегованої моделі було взято Лассо регресію, модель Extreme Gradient Boosting, Elastic Net та Ridge регресію.

```
averaged_models = AveragingModels(models = (ENet, GBoost, KRR, lasso))

score = rmsle_cv(averaged_models)
print(" Averaged base models score: {:.4f} ({:.4f})\n".format(score.mean(), score.std()))
```

Отриманий результат: RMSE = 0.1091, а стандартне відхилення становить 0.0075. Дана похибка є найменшою з усіх оцінених моделей. Подивимося який результат дана модель дасть на тренувальному датасеті.

```
stacked_averaged_models.fit(train.values, y_train)
stacked_train_pred = stacked_averaged_models.predict(train.values)
stacked_pred = np.expm1(stacked_averaged_models.predict(test.values))
print(rmsle(y_train, stacked_train_pred))
```

Бачимо, що RMSE похибка становить 0.0781, що є найкращим з прорахованих похибок. Кінцевий результат буде виглядати наступним чином (продемонстровано 5 перших передбачень, решта – в файлі *output.csv*). Він буде являти собою таблицю з двох колонок. Перша колонка буде відповідати прогнозованій ціні продажі конкретного об'єкта нерухомості а друга – його унікальному ідентифікатору (власне щоб ми могли однозначно встановити пару будинок – прогнозована цін).

	SalePrice	id
0	120129.316383	1461
1	153008.789313	1462
2	181999.671772	1463
3	195952.284998	1464
4	197477.703212	1465

В даному розділі були продемонстровані й застосовані на практиці описані моделі для побудови інформаційного забезпечення прогнозування цін на нерухомість. Був продемонстрований повний процес обробки, аналізу й прогнозування даних: від занесення умови у середовище Python й отримання кінцевого результату для кожного будинку з тестової та тренувальної вибірок.

Підсумовуючи, алгоритм нашого інформаційного забезпечення для прогнозування виглядає наступним чином. Ми отримали базу даних, в якій містилась інформація про близько 600 об'єктів нерухомості. Кожен об'єкт характеризувався досить великою кількістю атрибутів: від типу крівлі до місця

розташування. Цільовою змінною була «Ціна продажі», яка підсумовувала всі характеристики й базуючись на них визначала вартість об'єкта. Ціллю роботи було встановлення зв'язків між характеристиками та виокремлення певної закономірності формування ціни. Застосовуючи різні математичні підходи, теорію ймовірності та математичну статистику, а також їх реалізацію в комп'ютерних системах ми й будували наше інформаційне забезпечення. Спочатку ми підготували наші дані, нормалізували їх та провели базовий аналіз.

Ключовим моментом було застосування регуляризації (рідж-регресії, LASSO та Elastic Net) та дерев рішень (градієнтного бустінгу та Extreme Gradient Boosting), а також створення двох агрегованих моделей (перша складалася з LASSO та Extreme Gradient Boosting, друга - із Лассо, Extreme Gradient Boosting, Elastic Net та Ridge) для побудови моделі та прогнозу. Оцінювання моделей проводилось за допомогою RMSE похибки.

Оцінка якості прогнозу моделей – RMSE похибка та стандартне відхилення новедено у табл. 3.2.

Таблиця 3.2 – Результати прогнозування моделей

Model	**RMSE**	**Standard deviation**
Lasso	0.1115	0.0074
ElasticNet	0.1116	0.0074
Ridge	0.1153	0.0075
Gradient Boosting	0.1177	0.0080
Xgboost	0.1161	0.0079
Aggregated Model 1	0.1184	0.0076
Aggregated Model 2	***0.1091***	***0.0075***

Отже, було побудовано 7 моделей для прогнозування цін на нерухомість. Найкращий результат продемонструвала друга агрегована модель, у результаті

чого кінцеву модель прогнозування було вирішено подати у вигляді об’єднання вище перерахованих чотирьох моделей. Кінцева модель була випробувана на тестовому наборі даних, де RMSE похибка склала 0.0781. Кінцевий прогноз тестової вибірки знаходиться в csv файлі. Перша колонка відповідає прогнозованій ціні, друга – ідентифікатору будинку. Таким чином, прогнозована ціна будинку 1461 складає 120 129 гр. од., будинку 1462 – 153 008 гр. од. і т. д.

РОЗДІЛ 4
РОЗРОБКА ІНФОРМАЦІЙНОГО ЗАБЕЗПЕЧЕННЯ ПРОГНОЗУВАННЯ ЦІН НА НЕРУХОМІСТЬ ТА РЕКОМЕНДАЦІЇ ЩОДО ЙОГО ВИКОРИСТАННЯ

4.1 Вибір інструмента для реалізації інформаційного забезпечення прогнозування цін на нерухомість

Для організації взаємодії різних інформаційних систем з різними користувачами та між собою, дані потрібно відповідним чином описати в усіх системах на різних рівнях, тобто вирішити проблему їх інформаційної сумісності в найширшому розумінні. Це досягається створенням інформаційного забезпечення [69].

Інформаційне забезпечення – це сукупність форм документів, нормативної бази та реалізованих рішень щодо обсягів, розміщення та форм існування інформації, яка використовується в інформаційній системі [62]. Інформаційне забезпечення прогнозування цін на нерухомість – це автоматизована експертна система, яка з використанням математичних методів та моделей, а також комп’ютерних технологій на основі заданої бази даних надає візуалізовані результати визначення ринкової вартості житла, які можуть бути легко отримані та швидко проаналізовані зацікавленими сторонами.

Для реалізації інформаційного забезпечення прогнозування цін на нерухомість було вирішено використовувати такий інструмент як Amazon Forecast. Amazon Forecast – це повністю керована послуга, яка використовує машинне навчання для отримання високоточних прогнозів.

Сьогодні компанії використовують все – від простих електронних таблиць до складного програмного забезпечення для фінансового планування, щоб спробувати точно прогнозувати майбутні результати бізнесу, такі як попит на продукцію, потреби в ресурсах або фінансові показники. Ці інструменти будують

прогнози, переглядаючи історичну серію даних, яка називається даними часових рядів. Наприклад, такі інструменти можуть намагатися передбачити майбутні продажі плащів, дивлячись лише на попередні дані про продажі з основним припущенням, що майбутнє визначається минулим. Цей підхід може скласти зусилля для отримання точних прогнозів для великих наборів даних, що мають нерегулярні тенденції. Крім того, йому не вдається легко поєднати ряди даних, які змінюються з часом (наприклад, ціна, знижки, веб-трафік та кількість співробітників), із відповідними незалежними змінними, такими як характеристики товару та місця розташування магазину.

Amazon Forecast – це повністю керована послуга, тому немає серверів для надання, а також моделей машинного навчання для побудови, навчання або розгортання. Ви платите лише за те, що використовуєте, і при цьому немає мінімальних зборів і жодних попередніх зобов'язань.

Перевагами використання даного продукту є:

- На 50% точніші прогнози. Amazon Forecast забезпечує прогнози, які є на 50% точнішими за допомогою машинного навчання, щоб автоматично виявити, як впливають дані часових рядів та інші змінні, такі як характеристики продукту та місця зберігання. Ви можете краще зрозуміти, як ці складні взаємозв'язки в кінцевому рахунку впливають на попит, ніж те, що може дати лише огляд даних часових рядів. Моделі, які створює Amazon Forecast, унікальні для ваших даних, а це означає, що прогнози відповідають вашому бізнесу.
- Скорочення часу прогнозування. За допомогою Amazon Forecast можна досягти рівня точності прогнозування, який займав місяці інженерії всього за кілька годин. Ви можете імпортувати дані часових рядів та пов’язані дані в прогноз Amazon із вашої бази даних Amazon S3. Звідти Amazon Forecast автоматично завантажує ваші дані, перевіряє їх та визначає ключові атрибути, необхідні для прогнозування. Потім Amazon Forecast навчає та оптимізує вашу

власну модель і розміщує їх у високодоступному середовищі, де її можна використовувати для формування прогнозів вашого бізнесу. Завдяки автоматичній обробці складного машинного навчання, необхідного для побудови, підготовки, налаштування та розгортання моделі прогнозування, Amazon Forecast дозволяє швидко створювати точні прогнози.

- Можливість створювати практично будь-який прогноз часових рядів. Для ведення вашого бізнесу потрібно декілька типів прогнозів часових рядів - від грошових потоків до попиту на продукцію до планування ресурсів. Amazon Forecast дозволяє створювати прогнози практично для будь-якої галузі та випадків використання, включаючи роздрібну торгівлю, логістику, фінанси, ефективність реклами та багато іншого. Використовуючи машинне навчання, Amazon Forecast може працювати з будь-якими історичними даними часових рядів і використовувати велику бібліотеку вбудованих алгоритмів для автоматичного визначення найкращих результатів для вашого конкретного типу прогнозу.
- Захищеність даних. Кожна взаємодія з Amazon Forecast захищена шифруванням. Будь-який вміст, оброблений Amazon Forecast, зашифровується за допомогою ключів клієнта через Службу управління ключами Amazon і шифрується в спокої в регіоні AWS, де ви використовуєте послугу. Адміністратори можуть також контролювати доступ до прогнозу Amazon за допомогою політики дозволів AWS Identity and Access Management (IAM) - забезпечуючи захист та конфіденційність конфіденційної інформації.

Для розгортання інформаційного забезпечення було розроблено такий алгоритм потоку даних (рис. 4.1), щоб система мала самостійних характрер та могла використовуватися безпосередньо підприємствами або юридичними особами, які працюють на ринку нерухомості.

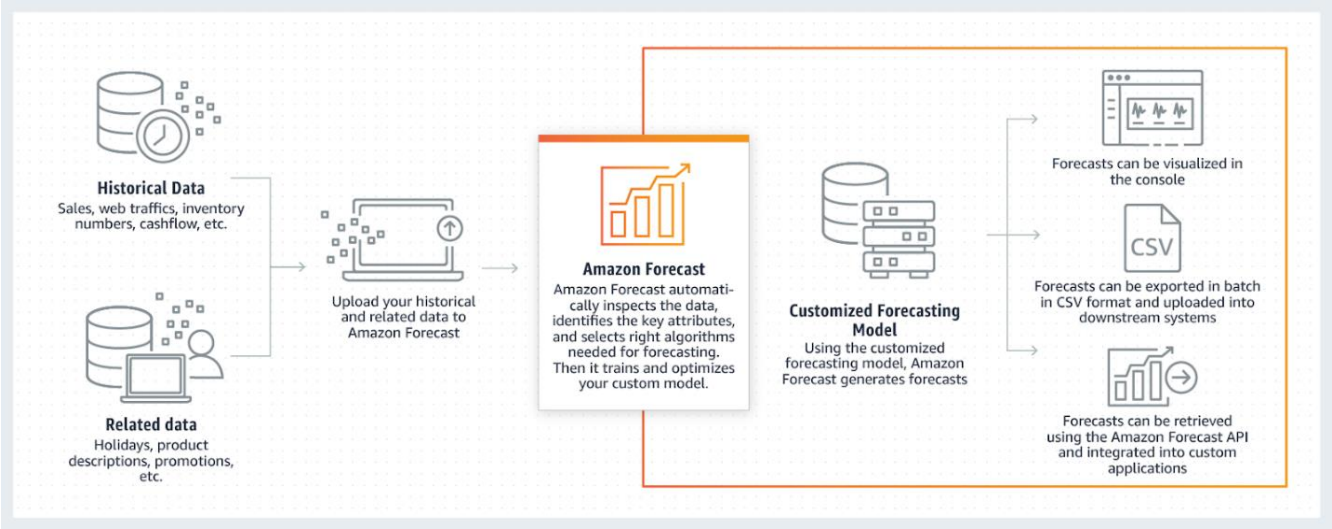

Рисунок 4.1 – Схема автоматизованого створення моделі прогнозування цін на нерухомість за допомогою Amazon Forecast.

Процес роботи системи починається з завантаження історичних даних, а також підвантаження пов’язаних даних, котрі можуть впливати на нашу модель. Далі дані передаються до системи Amazon Forecast, шифруються та обробляються там. На самому Amazon Forecast необхідно налаштувати параметри побудови нашої обраної моделі (Castomized Forecasting Model), вибрати методи машинного навчання (у нашому випадку побудова агрегованої моделі, яка складається з Лассо-регресії, Extreme Gradient Boosting, Elastic Net та Ridge), обрати технології підготовки даних (очищення від пропущених, нульових значень та викидів, нормалізація, перевірка на мультиколінеарність, тощо), вказати атрибути. Після цього відбувається опрацювання даних, їх обробка, підготовка, навчання моделі , прогноз, візуалізація та створення звіту для аналізу.

Отже, було проаналізовано основні переваги Amazon Forecast як інструмента для подальшого створення нашого інформаційного забезпечення прогнозування цін на нерухомість.

4.2. Алгоритм побудови інформаційного забезпечення прогнозування цін на нерухомість

Прогнозування попиту чи продажів чого часто здійснюється за допомогою інструментів 20-го століття, таких як Microsoft Excel. Переваги Excel з точки зору інтерактивності, знайомства та популярності очевидні. Однак обмеження Excel як ручного інструменту, що працює на одній машині, блокують практику масштабування обсягу даних, кількості необхідних прогнозів та можливості експериментувати з новішими методами. Набір інструментів у хмарі може перенести методи прогнозування у 21 століття та дозволити більш точні та частіші прогнози.

Ми створимо алгоритм прогнозування, який починається з запитів до бази даних, розміщеного в S3, який трансформується за допомогою Amazon Athena (керований Presto), розробляється за допомогою блокнота Jupyter в SageMaker, модель будується за допомогою Amazon Forecast, а візуалізація даних та результатів прогнозу відбувається за допомогою QuickSight. Вигодами від використання власних служб AWS є вартість, масштаб та інтеграція більшості послуг AWS. Однак можна також замінити будь-який із інструментів іншими, наприклад PowerBI для візуалізації або RDBMS для запитів та трансформації даних.

На рис. 4.2 наведено детальний алгоритм роботи сервісів AWS (у тому числі Amazon Forecast) для прогнозування цін на нерухомість.

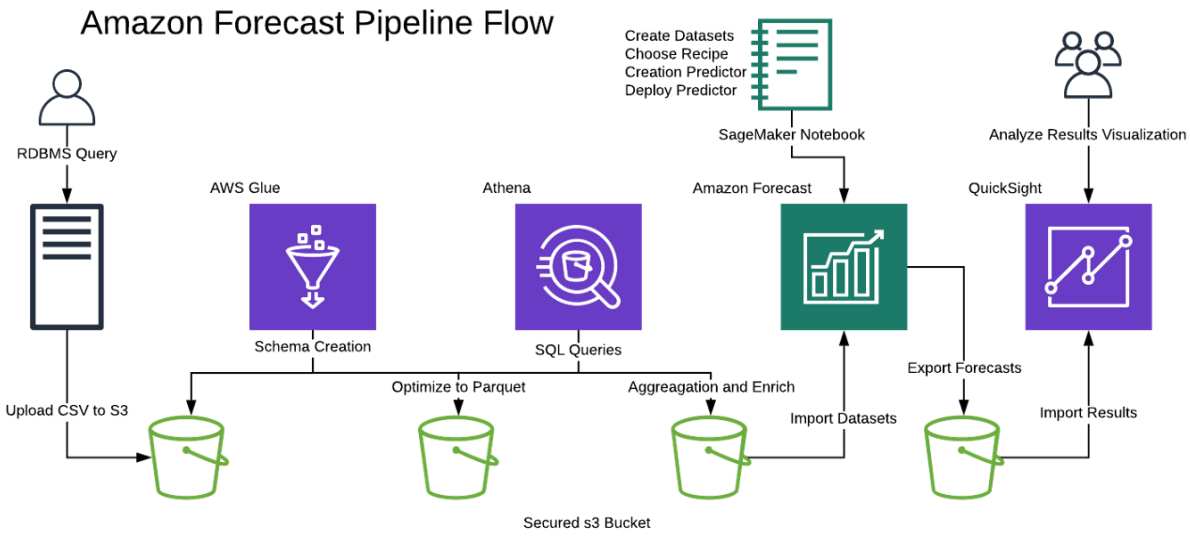

Рисунок 4.2 – Детальний алгоритм роботи Amazon Forecast для прогнозування цін на нерухомість.

Перш за все необхідно забезпечити захист даних у хмарі, для того щоб наша система працювала з мінімальним ризиком. Найкраще почати зі створення нових систем машинного навчання, які не можуть бути розвинені в поточній інфраструктурі. Так як ми обираємо цей шлях, нам потрібно знайти безпечні та зрозумілі способи отримання даних, необхідних для алгоритмів машинного навчання. Найкращими методами запуску надійної системи є створення нового сегмента в S3, з шифруванням за замовчуванням, створенна як приватна кінцева точка, що дає доступ лише з внутрішньої організації VPC, і ввімкнення аудиту за допомогою CloudTrail.

Використання Amazon Athena для перетворення даних. Як тільки дані надійно зберігаються в сегменті S3, можна розпочати процес обробки даних, створення моделей, підключення бібліотек, підготовку даних. Прогнозування не є науковою проблемою, оптимізацією даних чи автоматизацією процесів. Це поєднання всього вищесказаного. Тим не менше, процес починається з

перетворення даних плоских файлів у формат, що забезпечує ефективні запити щодо них, для дослідження даних, агрегування та інших подібних маніпуляцій. Менші набори даних можна було завантажувати в Excel або подібні програмні засоби. Однак дані тут є більш детальними та включають кількарічні щоденні дані будинки нерухомості. Частковий перегляд десятків файлів у S3 подано на рис. 4.3.

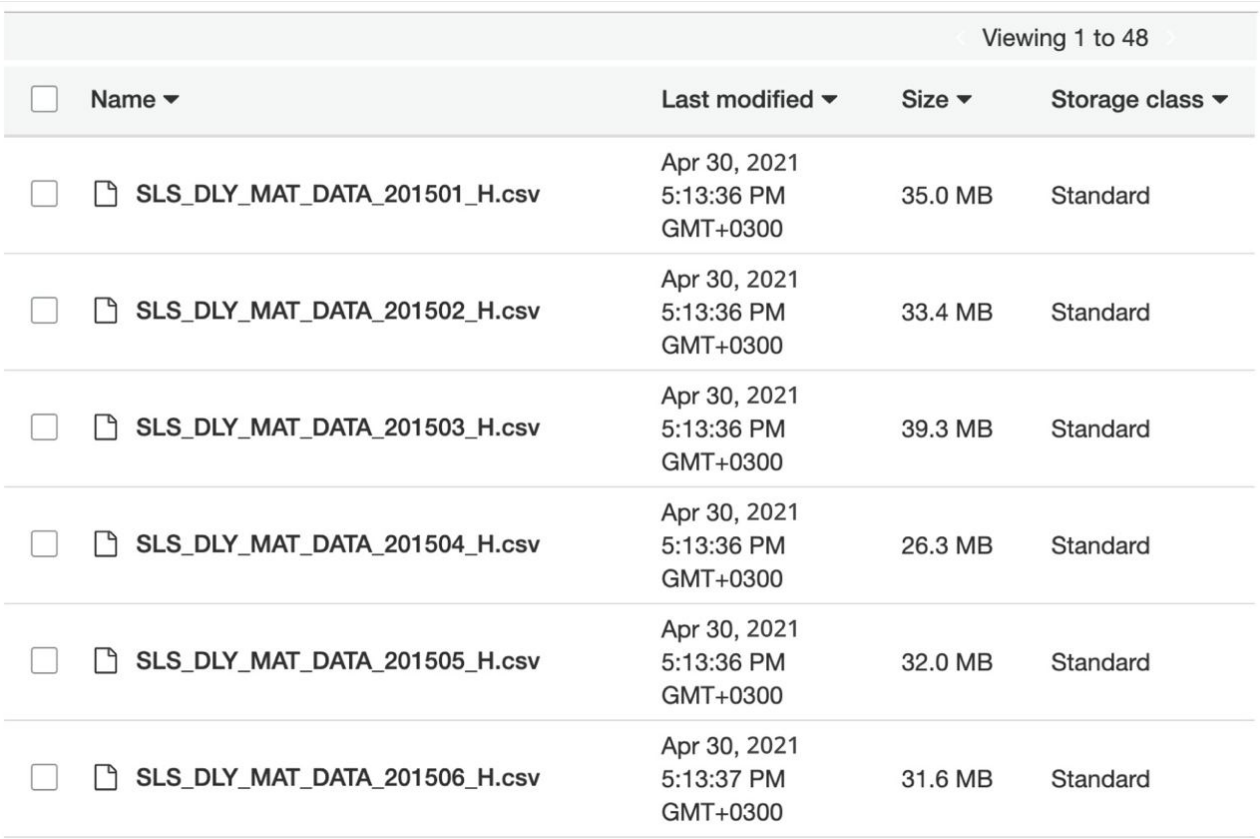

Рисунок 4.3 – Частина файлів про об’єкти нерухомості у S3.

Простий спосіб увімкнути плоскі файли CSV, щоб дозволити запити SQL, - це використання AWS Glue Crawler, який аналізує формат файлів даних і створює схему роботи. Візард у Glue може провести вас через процес, використовуючи Конфігурацію безпеки (1), Розташування даних (2), Дозвіл доступу (3) та Цільову базу даних (4) (рис. 4.4):

Рисунок 4.4 – Конфігурація даних у AWS Glue Crawler.

Через пару хвилин наступна таблиця з’явиться в каталозі даних Glue і буде доступна для запитів із використанням синтаксису SQL із Amazon Athena (рис. 4.5):

Рисунок 4.5 – Схема необроблених даних у каталозі даних AWS Glue.

Хоча дані тепер готові до фільтрації, агрегування та інших перетворень, рекомендується стиснути дані та розділити їх, щоб різко зменшити Create Table As Select (CTAS) в Athena:

```
/*Original Data Compressed*/

CREATE TABLE forecast.compressed_data

WITH (
     format='PARQUET',
     external_location='s3://forecast-xxxxxx/sagemaker/data/',
     partitioned_by = ARRAY['year']
) AS
SELECT *, year(calendar_date) as year FROM forecast.data;
```

Давайте перевіримо, що ми покращили завдяки цій трансформації. По-перше, давайте порахуємо, скільки різних продуктів ми маємо у вихідних даних у форматі CSV:

```
SELECT count (distinct item_id) FROM forecast.data where
year(calendar_date) = 2018;
```

(Час роботи: 2,95 секунди, скановані дані: 114,16 КБ)

Час роботи подібний, однак вартість у 10 000 разів дешевша. Тим не менше, будь-який інший запит, який ми будемо використовувати щодо оброблених та розділених даних, матиме суттєву (принаймні * 100) кращу продуктивність.

Модульна та гнучка архітектура, яку ми можемо побудувати у хмарі, - це повна зміна гри в галузі великих даних та машинного навчання. Час, необхідний для побудови робочого алгоритму, здатність зосередитись на будь-якій його частині та оптимізувати його до екстремального рівня, а також здатність випускати його на виробництво в будь-якому масштабі будуть цікаві багатьом людям зі сфери нерухомості, особливо людям, що мають досвід використання раніше подібних але з використанням локальної інфраструктури.

Побудова моделі прогнозування. Тепер, коли ми маємо свої дані, готові до аналізу, використовуючи різні інструменти – бібліотеки python, як Pandas та Scikit-

learn, або інструменти SQL для візуалізації та звітування, ми можемо звернутися до підготовки нашої першої базової моделі прогнозування за допомогою Amazon Forecast. Ми не будемо переходити безпосередньо до наймодерніших моделей з найвищим рівнем роздільної здатності деталей, оскільки більшість проектів машинного навчання вимагають багатьох (часто невдалих) експериментів, які призводять нас до найкращого результату, необхідного бізнесу. Кожен бізнес має різні вимоги, різні продукти, різні споживачі на різних ринках, і не існує універсального рішення, яке може працювати на першому випробуванні. Алгоритм, який ми будуємо тут, призначений для швидких експериментів та ітерацій, щоб полегшити процес розвідки та розробки наших моделей прогнозування.

Для першої моделі ми візьмемо найвищі агрегації даних і спробуємо побудувати прогноз на ціни житлових об'єктів. Вихідні дані містять більше деталей (рис. 4.6):

Results

	SalesPrice	GrLivArea	MSZoning	OverallQual	ID	BsmtCond	Date Sold
1	[redacted]	80	015	015063	000000000000344725	0010086927	2018-04-29 00:00:00.000
2	[redacted]	80	015	015063	000000000000344725	0010086928	2018-04-18 00:00:00.000

Рисунок 4.6 – Зразок сирої бази даних.

Ми знову використаємо Athena для створення зведених даних для моделі:

```
/*train_category training data aggregated by category*/
CREATE TABLE forecast.train_category
WITH (
 format='TEXTFILE',
 external_location='s3://XXXX/sagemaker/train_category/',
 field_delimiter = ','
) AS
SELECT category as item_id, date_format(calendar_date, '%Y-%m-%d')
as timestamp, sum(sales) as demand FROM forecast.compressed_data
WHERE calendar_date < CAST('2018-01-01' AS DATE)
GROUP BY 1,2
ORDER BY 2,1
```

У коді слід зазначити:

- Рядок 4: format = «TEXTFILE», Amazon Forecast очікує отримати файли формату CSV, а не Parquet. Дані стискаються за допомогою gzip, який за замовчуванням використовується для команд Create-Table-As-Select (CTAS).
- Рядок 8: категорія як item_id, для простоти ми перетворюємо рівень категорії в назву item_id, що є типовою назвою стовпця в Amazon Forecast для ідентифікатора ряду.
- Рядок 8: date_forma t(calendar_date, '%Y-%m-%d') як позначка часу, ще одна умова AF щодо назви стовпця та його формату.
- Рядок 8: ціна продаж, останній обов'язковий стовпець для Amazon Forecast та назва за замовчуванням.
- Рядок 9: FROM forecast.compressed_data, використовуючи стиснуту версію даних, які ми створили вище, для ефективності.
- Рядок 10: where calendar_date < CAST('2018–01–01' AS DATE), створюючи розділення даних для навчання та тесту.

Імпортування набору даних до прогнозу Amazon. Для вдосконалених прогнозування в Amazon ми використовуватимемо додаткові набори даних, такі як метадані елементів та відповідні часові ряди. Однак для першої простої моделі ми будемо використовувати лише цільові часові ряди.

Ми можемо використовувати консоль управління AWS для створення набору даних та імпорту даних із S3; однак, оскільки ми хочемо створити автоматизований алгористм, ми будемо використовувати для цього Python SDK. Я також використовую ноутбуки Jupyter на Amazon SageMaker, щоб зробити розробку більш інтерактивною та спростити обмін.

Через кілька хвилин завдання імпорту пройшло успішно і перетворюється на «Active». За допомогою блокнота Jupyter перевіряю, чи має сенс статистика даних набору даних:

Аналогічним чином тренуємо дані.

Створення прогнозів за допомогою Predictor. Щоб сформувати прогноз за допомогою предиктора, який ми щойно навчили, нам потрібно спочатку його розгорнути.

Коли предиктор буде розгорнуто та активний, ми можемо перейти до консолі управління AWS і отримати просту візуалізацію прогнозу на перші два місяці року для однієї з категорій (item_id = «012») (рис. 4.7):

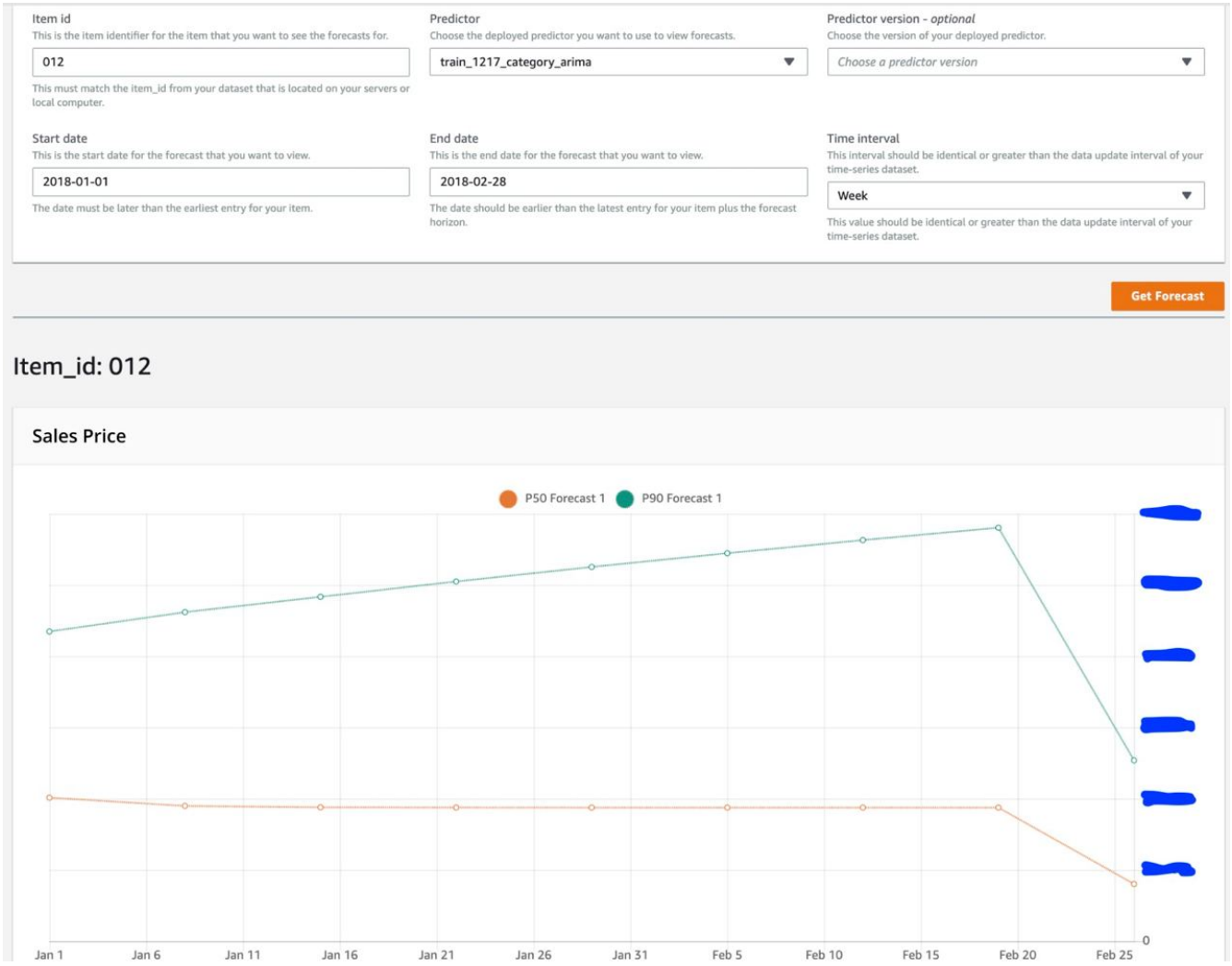

Рисунок 4.7 – Результат візуалізації щотижневого прогнозу цін на нерухомість за допомогою QuickSight.

Однак нам потрібен більш масштабований спосіб побачити точність прогнозів, порівняти його за різними категоріями, а пізніше порівняти з іншими рецептами. Ми хочемо використати інструмент візуалізації, такий як QuickSight, і для нього нам потрібно отримати прогнози до S3 і зробити на них запит за допомогою Athena.

Експорт прогнозів до S3 для подальшого аналізу. Ми створимо ще один AWS Glue Crawler для аналізу вихідних файлів прогнозу та генеруємо систему, яка дозволить нам зчитувати та запрашувати дані за допомогою Athena та QuickSight. Ми використаємо візард та отримаємо подібне значення сканера, за винятком вказівки на вихідну папку та створення таблиці з префіксом «output_» (рис. 4.8):

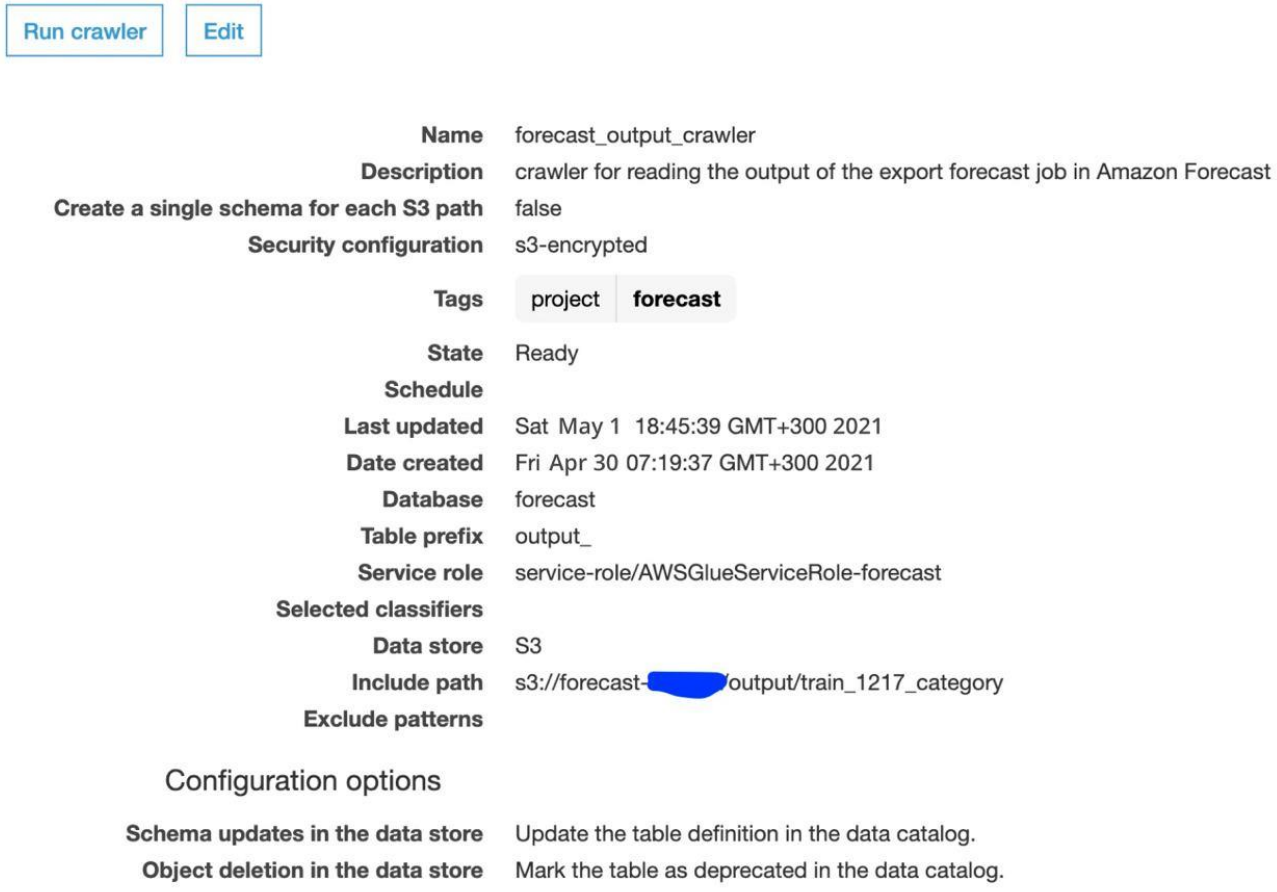

Рисунок 4.8 – AWS Glue Crawler для вихідних файлів прогнозу.

Завантаження даних через Athena займає кілька секунд, а потім кілька клацань, щоб створити першу візуалізацію (рис. 4.9):

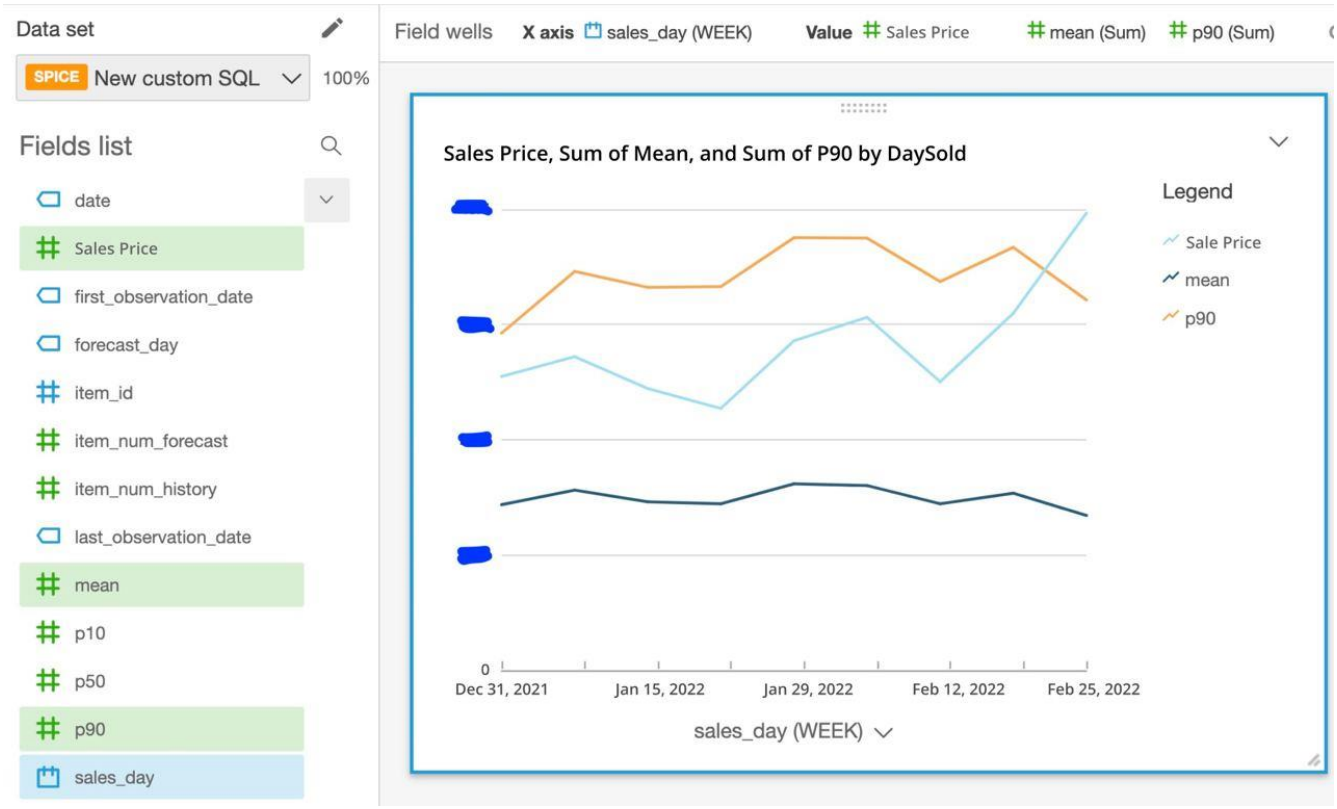

Рисунок 4.9 – Візуалізація прогнозу цін на нерухомість в QuickSight.

Отже, на основі агрегованої моделі, ми побудували інформаційне забезпечення прогнозування цін на нерухомість для потенціального використання зацікавлених осіб та підприємств зі сфери нерухомості. Ми використовували комбінацію AWS Glue Crawler, Athena, Amazon Forecast та QuickSight. У майбутньму ми можемо розглянути шляхи підвищення точності прогнозу, додавши більше наборів даних, а також продовжити комбінувати різні підходи моделювання часових рядів (DeepAR, Prophet тощо); також вивчимо способи подальшої автоматизації процесу та дозволимо щодня оновлювати моделі без ручної роботи, використовуючи Step Function, Lambda та інші допоміжні сервіси.

4.3. Практичне використання інформаційного забезпечення прогнозування цін на нерухомість у бізнесі

Індустрія нерухомості стала важливою частиною сучасних фінансових ринків. В даний час ряд дослідників і практиків досліджували сферу нерухомості, використовуючи статистичні методи та методи штучного інтелекту.

Для прийняття правильних рішень важливим є точний, надійний, науково обґрунтований прогноз. Але прийняття правильного рішення щодо ціни є складним, особливо якщо мова йде про нового клієнта, нестабільний ринок або великий пул різноманітних клієнтів. Знання того, як будуть змінюватися майбутні ціни на нерухомість, дозволить адаптувати свої ціни або знижки з заданою частотою, щоб бізнеси могли оптимізувати свою націнку залежно від попиту, запасів тощо.

Використання побудованого інформаційного забезпечення дозволить зацікавленим сторонам орієнтуватися в актуальних цінах на нерухомість та не девальвувати ринок. Знаючи про розвиток майбутніх цін, бізнеси можуть одночасно купувати та продавати актив, щоб отримати прибуток від дисбалансу на ринку. Це торгівля, відома як арбітраж, що дозволяє використовувати різницю в цінах однакових або подібних фінансових інструментів на різних ринках.

Зацікавленими сторонами у використанні даного інформаційного забезпечення прогнозування цін на нерухомість можуть бути наступні економічні суб’єкти [54]:

- *продавці (орендодавці)* – можуть використовувати прогнозні ціни на житло аби досягти оптимального прибутку у даний або майбутній період продажі/здачі в оренду житла;
- *покупці (орендарі)* – для визначення майбутньої ціни на житло (покупка/оренда), щоб прийняти ефективне рішення з точки зору власних потреб та можливостей;

- професійні учасники ринку нерухомості. Наприклад, *менеджери фондів нерухомості*, які за допомогою прогнозування можуть розробляти ефективніші інвестиційні стратегії, використовуючи інформаційного забезпечення прогнозування цін на нерухомість. Це сприятиме інвестиційній ефективності ринку аукціонів нерухомості та допоможе досягти ефективних фінансових ринків. Крім того, це може допомогти досягти стійких економічних вигод відповідним зацікавленим сторонам на ринках нерухомості.

Також можна виділити потенціальних неінституціональних учасників ринку, які зможуть використовувати побудоване інформаційного забезпечення.

Це такі суб’єкти як [55]:

- брокери;
- оцінювачі нерухомості;
- фінансисти (банкіри);
- девелопери та редевелопери;
- керуючі нерухомістю, що займаються фінансовим управлінням та технічною експлуатацією об'єкта;
- проектувальники і будівельники;
- учасники фондового ринку нерухомості;
- аналітики;
- маркетологи; фахівці зі зв'язків з громадськістю та рекламі, які займаються просуванням об'єктів і послуг на ринку;
- інформаційно-аналітичні видання та інші ЗМІ;
- фахівці з інформаційних технологій;
- співробітники і члени національних і міжнародних професійних об'єднань ринку нерухомості.

Отже, у цьому розділі ми побудували інформаційне забезпечення прогнозування цін на нерухомість на основі агрегованої моделі, а також було виокремлено основних зацікавлених сторін, які в подальшому можуть використовувати створене інформаційне забезпечення, що відіграватиме важливу роль у підтримці економічного зростання ринку нерухомості.

ВИСНОВКИ

У сучасному, прогресивному світі надзвичайно актуальним є питання швидкого та якісного збору, обробки й аналізу даних. Ці дані можна використовувати для отримання правильних висновків і результатів, на основі яких можуть прийматись подальші бізнес-рішення.

Аналіз ринку нерухомості дозволяє припустити, що впливи на ціни об'єктів житлової нерухомості, можна розділити на дві основні групи: локальні та глобальні. Якщо детальніше розглядати механізм формування ринку нерухомості, то основними детермінантами попиту на житло є демографічні показники. Але інші фактори, такі як дохід, ціна житла, вартість та доступність кредиту, споживчі переваги, уподобання інвесторів, ціна замінників та ціна на доповнення, також відіграють свою роль.

Сектор нерухомості, як в Україні так і у світі, є одним із секторів з найбільшою капіталізацією, на якому активно з’являються нові будівельні проекти й укладаються серйозні угоди. Аналіз, моделювання й прогнозування цін об’єктів на ринку нерухомості допоможе в прийнятті правильних та вигідних рішень для інвестування. Але процес прогнозування вартості нерухомості є складним, так як даний ринок достатньо чутливий до сторонніх подій та непередбачуваний. Класичні методи не дають бажаного результату в цьому випадку. Для таких ситуацій прогнозування нечітких змінних почали використовувати методи машинного навчання, які дають більш точні результати в порівнянні з іншими методами. Були описані основні принципи функціонування ринку нерухомості, фактори ціноутворення та способи визначення впливу того чи іншого фактору на ціну об’єкта. Була дана оглядова річна характеристика сучасного ринку нерухомості України за 2004-2020 рр.

Були визначені та практично застосовані основні підходи машинного навчання для вирішення задач регресії. По-перше, були вивчені та виокремлені

теоретичні основи роботи даних моделей, їх математична реалізація та логіка. Застосовані дані моделі були на прикладі ринку нерухомості, враховуючи його особливості. Таким чином була побудована універсальна модель прогнозування ціни об’єктів нерухомості, яка дозволяє на основі вхідних параметрів, тобто характеристик будинку (типу крівлі, площі, кількості машин в гаражі) визначати ціну продажі даного об’єкту. Варто зазначити те, що дана модель була натренована на даних, що представляють ринок нерухомості невеликого містечка, тому для адаптації даної моделі до ринку нерухомості мегаполіса необхідно навчити її на даних відповідного ринку нерухомості. Це допомогло не тільки налаштувати моделі для роботи саме з об’єктами нерухомості, а й краще розуміти отриманий результат і процес побудови моделі, що допомогло уникнути помилок.

Були також розглянуті на практиці й описані теоретично інші, більш складні поняття й підходи машинного навчання до моделювання ринку нерухомості (які можна застосовувати й на інших наборах даних), серед яких регуляризація, гребенева регресія, регресія LASSO, Elastic Net регресія, дерева рішень та методи градієнтного бустінгу.

Крім того, були описані основні аспекти й надана інформація по реалізації даних методів й моделей за допомогою комп’ютерних систем, а саме середовища мови програмування Python.

База даних, но основі якої проводиться дослідження, складається з інформації про 1300 об’єктів нерухомості невеликого містечка населенням близько 50 тис. жителів та розташуванням відносно далеко від великих центрів. Цільова змінна – «Ціна продажі». Для моделювання цільової змінної використовується 79 незалежних змінних, таких як: місце розташування, плани поверхів, площа, підлога, тип будівлі, кількість поверхів, якість оздоблення, відстань від метро, перехрестя, екологічний стан тощо. Усі дані порівну розділені на тренувальну та тестову частини.

До того, як почати працювати з набором даних необхідно його підготувати: почистити, заповнити пропуски, нормалізувати, позбавитися від мультиколінеарності тощо. За допомогою логарифмічної нормалізації цільової змінної було підготовано її для якісного моделювання в подальшому.

Для прогнозування було обрано 5 моделей: гребенева регресія, LASSO регресія, Elastic Net регресія, градієнтний бустінг та Xgboost модель. Найкращий результат на нашій вибірці показала LASSO регресія. Для того, щоб уникнути одностороннього підходу й розширити нашу модель, ми створили 2 агреговані моделі (ансамблі) – перша складалася з LASSO та Extreme Gradient Boosting, друга – із Лассо, Extreme Gradient Boosting, Elastic Net та Ridge).

Результатом стало отримання прогнозів ціни об'єктів нерухомості певного ринку (як вже зазначалось раніше, модель є універсальною й може застосовуватись на будь-яких даних ринку нерухомості). Найкращий результат продемонструвала друга агрегована модель – отриманий прогноз має логарифмічну середньоквадратичну похибку 0.1091, тому його можна вважати досить точним. Агрегована модель покращує точність прогнозування на 5% порівняно з іншими моделями. Це готовить про те, що краще використовувати комбінацію моделей для отримання найбільш точного прогнозу.

На основі обраної моделі було побудовано інформаційне забезпечення за допомогою комбінації інструментів AWS Glue Crawler, Athena, Amazon Forecast та QuickSight, які дозволяють використовувати його як продукт для подальшого використання зацікавленими учасниками ринку нерухомості.

Інформаційне забезпечення прогнозування цін на нерухомість – це автоматизована експертна система, яка з використанням математичних методів та моделей, а також комп'ютерних технологій на основі заданої бази даних надає візуалізовані результати визначення ринкової вартості житла, які можуть бути легко отримані та швидко проаналізовані зацікавленими сторонами. Першим

кроком для реалізації забезпечення було створено детальний алгоритм роботи Amazon Forecast для прогнозування цін на нерухомість, а також наведено план імпорту даних в допоміжні інструменти, правильна їх обробка та отримання репорту, який може бути використаний кінцевими споживачами інформаційного забезпечення.

Використання даного побудованого інформаційного забезпечення дозволить зацікавленим сторонам орієнтуватися в актуальних цінах на нерухомість та не девальвувати ринок. Знаючи про розвиток майбутніх цін, бізнеси можуть одночасно купувати та продавати актив, щоб отримати прибуток від дисбалансу на ринку. Це торгівля, відома як арбітраж, що дозволяє використовувати різницю в цінах однакових або подібних фінансових інструментів на різних ринках.

Отже, в даній роботі було використано такі методами машинного навчання для прогнозування цін на нерухомість: LASSO Regression, Elastic Net Regression, Ridge Regression, Gradient Boosting Regression, XGBoost. Найбільшу похибку дала модель градієнтного бустінгу. Тому, з метою покращення точності була побудована агрегована модель, за допомогою чого ми підвищили точність прогнозу на 5%. Також був побудований алгоритм реалізації інформаційного забезпечення для прогнозування цін на нерухомість, яке може в подальшому бути використана підприємствами та юридичними особами, які функціонують на ринку нерухомості.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Воронін В.О. Дослідження ринку нерухомості. Проблеми, тенденції, прогнозування / В.О. Воронін, Е. В. Лянце, М. М. Мамчин // Вісник Національного університету «Львівська політехніка», 2010. – № 690. – С. 540–552.

2. Данченко О.Б. Огляд сучасних методологій управління ризиками в проектах / О.Б. Данченко // Управління проектами та розвиток виробництва. – 2014. – № 1. – С. 16–25.

3. Єфіменко І.А. Інституалізація ринку нерухомості в трансформаційній економіці: дис. ... канд. екон. наук: 08.00.01. – Харків, 2007. – 148 с.

4. Офіційний веб-сайт Державного комітету статистики України. [Електронний ресурс]. – Режим доступу: http://www.ukrstat.gov.ua/

5. Поліщук Є.А. Ринок нерухомості як сфера діяльності девелоперських компаній: дис. канд. екон. наук: 08.00.08. – Київ, 2009. –19 с.

6. Професіональний інформаційно-аналітичний ресурс, присвячений машинному навчанню, розпізнаванню образів й інтелектуальному аналізу даних MachineLearning.ru. Режим доступу: http://www.machinelearning.ru

7. Тесля Ю.М. Науково-методологічні засади мета-методології впливу на управління проектами на основі концепції несилової взаємодії/ Ю.М. Тесля Ю. Л. Хлевна, А.О. Хлевний // Тези доповідей III Міжн. науково-практичної конф. «Інформаційні технології та взаємодії», 8-10 листопада 2016 р. / М-во освіти і науки України, КНУ ім. Тараса Шевченка та ін.. – К.:, 2016. – С. 113 – 115.

8. Хлевна Ю.Л. Експертний метод формування інформаційного простору мета-методології управління проектами / Ю.Л. Хлевна // Управління розвитком складних систем. – 2018. №35. – с. 61-66.

9. New Zeland Agricultural and Resourse Economics Society (Inc.). House Price Prediction: Hedonic Price Model vs. Artificial Neural Network.

URL:https://researcharchive.lincoln.ac.nz/bitstream/handle/10182/5198/House_%20price_%20prediction.pdf?sequence=1&isAllowed=y

10. A Course in Machine Learning by Hal Daumé III - ciml.info 2012, 189 p.

11. Bayesian Reasoning and Machine Learning by David Barber - Cambridge University Press 2011, 644 p.

12. Big Data Visualization by James D. Miller - Packt Publishing 2017, 304 p.

13. Gaussian Processes for Machine Learning by Carl E. Rasmussen, Christopher K. I. Williams - The MIT Press 2005, 266 p.

14. Inductive Logic Programming: Theory and Methods by Stephen Muggleton, Luc de Raedt - ScienceDirect 1994, 51 p.

15. Information Theory, Inference, and Learning Algorithms by David J. C. MacKay - Cambridge University Press 2003, 640 p.

16. Introduction To Machine Learning by Nils J Nilsson – 1997, 209 p.

17. Learning Predictive Analytics with Python by Ashish Kumar - Packt Publishing 2016, 354 p.

18. Machine Learning & Data Science Landscape by Christina Voskoglou, Mark Wilcox, Stijn Schuermans – VisionMobile 2017, 46 p.

19. Machine Learning, Neural and Statistical Classification by D. Michie, D. J. Spiegelhalter - Ellis Horwood 1994, 298 p.

20. Mastering Python Data Analysis by Magnus Vilhelm Persson, Luiz Felipe Martins - Packt Publishing 2016, 284 p.

21. Microsoft Developer Network. Режим доступу: https://msdn.microsoft.com/

22. Python Data Analysis, 2nd Edition by Armando Fandango - Packt Publishing 2017, 330 p.

23. Practical Data Analysis, 2nd Edition by Hector Cuesta, Sampath Kumar - Packt Publishing 2016, 338 p.

24. Python Data Analysis Cookbook by Ivan Idris – Packt Publishing 2016, 462p.

25. Practical Machine Learning by Sunila Gollapudi – Packt Publishing 2016, 468 p.

26. Python for Data Analysis, 2E By Wes McKinney – O'Reilly Media 2016, 550 p.

27. Teslya, I. & Khlevna, I. (2017). Structure of knowledge in the project management meta-methodology. Management of Development of Complex Systems, 78 – 85 [in Ukrainian].

28. The Elements of Statistical Learning: Data Mining, Inference, and Prediction by T. Hastie, R. Tibshirani, J. Friedman - Springer 2009, 764 p.

29. Michał Oleszak, Regularization: Ridge, Lasso and Elastic Net. 2020. Vol. 36, № 5. P. 561–570. URL: https://www.shirin-glander.de/2018/11/ml_basics_gbm/ (Last accessed: 02.11.2020).

30. XGBoost Tutorials. Introduction to Boosted Trees. 2020. Vol. 12, № 1. P. 321–238. URL: https://xgboost.readthedocs.io/en/latest/tutorials/model.html (Last accessed: 05.02.2021).

31. Du, H.; Mulley, C. Transport accessibility and land value: A case study of Tyne andWear. RICS Res. Paper Ser. 2007, 7, 52.

32. R. Cellmer, K. Szczepankowska. / The 9th Conference Environmental Engineering. Selected Papers, Article number: enviro.2014.113.

33. Radzewicz, A.; Wiśniewski, R. 2011. The uncertainty of real estate market, Journal of the Polish Real Estate Scientific Society19(1): 47–59.

34. Kucharska-Stasiak, E. 2014.Uncertainty of property valuation as a subject of academic research, Real Estate Management and Valuation 21(4): 17–25. http://dx.doi.org/10.2478/remav-2013-0033

35. French, N.; Mallinson, M. 2000. Uncertainty in property valuation. The Nature and Relevance of uncertainty and how it might be measured and reported, Journal of Property and Finance 18(1): 13–32. http://dx.doi.org/10.1108/14635780010316636

36. Harris, Richard (2016). "The Rise of Filtering Down". Social Science History. 37 (4): 515–549. doi:10.1017/S0145553200011950.

37. Kawaguchi, Y., (2013), Real Estate Economics, Seibunsha, Tokyo.

38. Ринок нерухомості України. [Електронний ресурс]. – Режим доступу: https://minfin.com.ua/ua/realty/

39. The cost of renting apartments in Kiev (S&V Development). URL: http://www.svdevelopment.com/ru/web/indicators/

40. World Economic Outlook Database (International Monetary Fund). URL: https://www.imf.org/external/pubs/ft/weo/2020/01/weodata/index.aspx

41. Monetary and Financial Statistics June 2020 (National Bank of Ukraine). URL: https://bank.gov.ua/admin_uploads/article/MFS_2020-06_en.pdf?v=4

42. Financial Sector Statistics (National Bank of Ukraine). URL: https://bank.gov.ua/en/statistic/sector-financial

43. Housing stock (State Statistics Service of Ukraine). URL: https://ukrstat.org/en/operativ/operativ2020/zf/zf/2019_e.htm

44. Ukraine Doing Business 2020 (World Bank). URL: https://www.doingbusiness.org/content/dam/doingBusiness/country/u/ukraine/UKR.pdf

45. Ease of Doing Business rankings (World Bank). URL: https://www.doingbusiness.org/en/rankings

46. Rumelhart, D.E.; Hinton, G.E.; Williams, R.J. Learning representations by back-propagating errors. Nature1986, 323, 533–536.

47. Lin, C.T.; Lee, C.G. Neural Fuzzy Systems: A Neuro-Fuzzy Synergism to Intelligent Systems; Prentice Hall: Upper Saddle River, NJ, USA, 1996.

48. Holland, J. Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence; University of Michigan Press: Ann Arbor, MI, USA, 1975; pp. 439–444.

49. Lewis, C.D. Industrial and Business Forecasting Methods; Butterworth Scientific: London, UK, 1982.

50. Zhang, R.; Du, Q.; Geng, J.; Liu, B.; Huang, Y. An improved spatial error model for the mass appraisal of commercial real estate based on spatial analysis: Shenzhen as a case study. Habitat Int. 2015, 46, 196–205.

51. Leamer, E.E. Housing is the Business Cycle. NBER Working Paper No. 13428. 2007. Available online: http://www.nber.org/papers/w13428 (accessed on 9 August 2020).

52. Beimer, W.; Maennig, W. Noise effects and real estate prices: A simultaneous analysis of different noise sources. Transp. Res. Part D 2017, 54, 282–286.

53. Ferlan, N.; Bastic, M.; Psunder, I. Influential Factors on the Market Value of Residential Properties. Inz. Ekon. Eng. Econ. 2017, 28, 135–144.

54. Singh, A.; Sharma, A.; Dubey, G. Big data analytics predicting real estate prices. Int. J. Syst. Assur. Eng. Manag. 2020.

55. Mangialardo, A.; Micelli, E. Simulation Models to Evaluate the Value Creation of the Grass-Roots Participation in the Enhancement of Public Real Estate Assets with Evidence from Italy. Buildings 2017, 7, 100. [CrossRef]

56. Cellmer, R.; Szczepankowska, K. Simulation Modeling in a Real Estate Market. In Proceedings of the 9th International Conference Environmental Engineering, Vilnius, Lithuania, 22–23 May 2014.

57. Bura Y., Khlevna I. House price modeling by machine learning. // Information Technology and Interactions (Satellite): Conference Proceedings, December 04, 2020, Kyiv, Ukraine / Taras Shevchenko National University of Kyiv and [etc]; Vitaliy Snytyuk (Editor). Kyiv: Stylos, 2020.– P. 124 – 126.

58. Манн Р.В. Ціноутворюючі фактори на регіональному ринку нерухомості / Р.В. Манн // «Економіка та держава» Міжнародний науковопрактичний журнал. – 2006. – № 3. – С. 49–51

59. Манн Р.В. Регіональні ринки нерухомості: особливості розвитку в Україні / Р.В. Манн // Науковий вісник Полтавського національного технічного університету ім. Юрія Кондратюка «Економіка і регіон». – 2004. – № 1 (2). – С. 53–55.

60. Геєць В.М. Нестабільність та економічне зростання / В.М. Геєць. – К. : Ін-т економ. прогнозування, 2000. – 341 с.

61. Лебідь Н.П., Гайдук В.Я. Регіональні особливості ціноутворення на ринку об’єктів нерухомості, що приватизуються // Власність в Україні. – 2000. – № 1. – С. 42–58.

62. Ринок нерухомості: навч. посібник / д.е.к., професор А.М. Асаул, д.е.к., професор В.І. Павлов, д.е.к., професор І.І Пилипенко, к.е.н., доц. Н.В. Павліха, к.е.н., доц. І.В. Кривовязюк. - К.: ІВЦ Держкомстату України, 2004. – 387 с.

63. Экономика недвижимости: учебное пособие / Д.В. Виноградов, С.Ю. Дерябин, - Владим. гос. унт им. А.Г. и Н.Г. Столетовых, Владимир: Изд-во Владим. Гос.ун-та, 2011.– 193 с.

64. Stevenson, S.; Young, J.; Gurdgiev, C. A comparison of the appraisal process for auction and private treaty residential sales. J. Hous. Econ. 2010, 19, 145–154.

65. Do, A.Q.; Grudnitski, G. A neural network approach to residential property appraisal. Real Estate Apprais. 1992, 58, 38–45.

66. Nanda, S.R.; Mahanty, B.; Tiwari, M.K. Clustering Indian stock market data for portfolio management. Expert Syst. Appl. 2010, 37, 8793–8798.

67. Patro, S.; Sahoo, P.P.; Panda, I.; Sahu, K.K. Technical analysis on financial forecasting. arXiv 2015, arXiv:1503.03011.

68. Patro, S.; Sahu, K.K. Normalization: A preprocessing stage. arXiv 2015, arXiv:1503.06462.

69. Lewis, C.D. Industrial and Business Forecasting Methods; Butterworth Scientific: London, UK, 1982.

70. Zhang, R.; Du, Q.; Geng, J.; Liu, B.; Huang, Y. An improved spatial error model for the mass appraisal of commercial real estate based on spatial analysis: Shenzhen as a case study. Habitat Int. 2015, 46, 196–205.

ДОДАТКИ

ДОДАТОК А

Опис бази даних

Набір даних описує продаж приватної житлової нерухомості в Амесі, штат Айова з 2006 по 2010 рік. Дані були зібрані Діном Де Коком для використання у сфері data science.

Набір даних містить 2930 спостережень та велику кількість пояснювальних змінних (23 номінальних, 23 порядкових, 14 дискретних та 20 безперервних) в оцінці ціни на нерухомість.

Quantitative: 1stFlrSF, 2ndFlrSF, 3SsnPorch, BedroomAbvGr, BsmtFinSF1, BsmtFinSF2, BsmtFullBath, BsmtHalfBath, BsmtUnfSF, EnclosedPorch, Fireplaces, FullBath, GarageArea, GarageCars, GarageYrBlt, GrLivArea, HalfBath, KitchenAbvGr, LotArea, LotFrontage, LowQualFinSF, MSSubClass, MasVnrArea, MiscVal, MoSold, OpenPorchSF, OverallCond, OverallQual, PoolArea, ScreenPorch, TotRmsAbvGrd, TotalBsmtSF, WoodDeckSF, YearBuilt, YearRemodAdd, YrSold

Qualitative: Alley, BldgType, BsmtCond, BsmtExposure, BsmtFinType1, BsmtFinType2, BsmtQual, CentralAir, Condition1, Condition2, Electrical, ExterCond, ExterQual, Exterior1st, Exterior2nd, Fence, FireplaceQu, Foundation, Functional, GarageCond, GarageFinish, GarageQual, GarageType, Heating, HeatingQC, HouseStyle, KitchenQual, LandContour, LandSlope, LotConfig, LotShape, MSZoning, MasVnrType, MiscFeature, Neighborhood, PavedDrive, PoolQC, RoofMatl, RoofStyle, SaleCondition, SaleType, Street, Utilities,

ДОДАТОК Б

Опис атрибутів бази даних

SalePrice - the property's sale price in dollars. This is the target variable that you're trying to predict.

MSSubClass: The building class

MSZoning: The general zoning classification

LotFrontage: Linear feet of street connected to property

LotArea: Lot size in square feet

Street: Type of road access

Alley: Type of alley access

LotShape: General shape of property

LandContour: Flatness of the property

Utilities: Type of utilities available

LotConfig: Lot configuration

LandSlope: Slope of property

Neighborhood: Physical locations within Ames city limits

Condition1: Proximity to main road or railroad

Condition2: Proximity to main road or railroad (if a second is present)

BldgType: Type of dwelling

HouseStyle: Style of dwelling

OverallQual: Overall material and finish quality

OverallCond: Overall condition rating

YearBuilt: Original construction date

YearRemodAdd: Remodel date

RoofStyle: Type of roof

RoofMatl: Roof material

Exterior1st: Exterior covering on house

Exterior2nd: Exterior covering on house (if more than one material)

MasVnrType: Masonry veneer type

MasVnrArea: Masonry veneer area in square feet

ExterQual: Exterior material quality

ExterCond: Present condition of the material on the exterior

Foundation: Type of foundation

BsmtQual: Height of the basement

BsmtCond: General condition of the basement

BsmtExposure: Walkout or garden level basement walls

BsmtFinType1: Quality of basement finished area
BsmtFinSF1: Type 1 finished square feet
BsmtFinType2: Quality of second finished area (if present)
BsmtFinSF2: Type 2 finished square feet
BsmtUnfSF: Unfinished square feet of basement area
TotalBsmtSF: Total square feet of basement area
Heating: Type of heating
HeatingQC: Heating quality and condition
CentralAir: Central air conditioning
Electrical: Electrical system
1stFlrSF: First Floor square feet
2ndFlrSF: Second floor square feet
LowQualFinSF: Low quality finished square feet (all floors)
GrLivArea: Above grade (ground) living area square feet
BsmtFullBath: Basement full bathrooms
BsmtHalfBath: Basement half bathrooms
FullBath: Full bathrooms above grade
HalfBath: Half baths above grade
Bedroom: Number of bedrooms above basement level
Kitchen: Number of kitchens
KitchenQual: Kitchen quality
TotRmsAbvGrd: Total rooms above grade (does not include bathrooms)
Functional: Home functionality rating
Fireplaces: Number of fireplaces
FireplaceQu: Fireplace quality
GarageType: Garage location
GarageYrBlt: Year garage was built
GarageFinish: Interior finish of the garage
GarageCars: Size of garage in car capacity
GarageArea: Size of garage in square feet
GarageQual: Garage quality
GarageCond: Garage condition
PavedDrive: Paved driveway
WoodDeckSF: Wood deck area in square feet
OpenPorchSF: Open porch area in square feet
EnclosedPorch: Enclosed porch area in square feet
3SsnPorch: Three season porch area in square feet

ScreenPorch: Screen porch area in square feet

PoolArea: Pool area in square feet

PoolQC: Pool quality

Fence: Fence quality

MiscFeature: Miscellaneous feature not covered in other categories

MiscVal: $Value of miscellaneous feature

MoSold: Month Sold

YrSold: Year Sold

SaleType: Type of sale

SaleCondition: Condition of sale

ДОДАТОК В

Відсоток та кількість пропущених значень у деяких змінних

Variable	Total	Percent
PoolQC	1453	0.995205
MiscFeature	1406	0.963014
Alley	1369	0.937671
Fence	1179	0.807534
FireplaceQu	690	0.472603
LotFrontage	259	0.177397
GarageCond	81	0.055479
GarageType	81	0.055479
GarageYrBlt	81	0.055479
GarageFinish	81	0.055479
GarageQual	81	0.055479
BsmtExposure	38	0.026027
BsmtFinType2	38	0.026027
BsmtFinType1	37	0.025342
BsmtCond	37	0.025342
BsmtQual	37	0.025342
MasVnrArea	8	0.005479
MasVnrType	8	0.005479
Electrical	1	0.000685
Utilities	0	0.000000