Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)
Кафедра комп'ютерних систем та мереж
(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

бакалавр
(назва освітнього ступеня)

на тему: *Комп'ютеризована система формування ціни на нерухомість з використанням хмарних сервісів*

Виконав: студент *IV* курсу, групи *СІс-44*
спеціальності *123 «Комп'ютерна інженерія»*
(шифр і назва спеціальності)

(підпис) *Горохівський А.В.* (прізвище та ініціали)

Керівник (підпис) *Тиш Є.В.* (прізвище та ініціали)

Нормоконтроль (підпис) *Луцик Н.С.* (прізвище та ініціали)

Завідувач кафедри (підпис) *Осухівська Г.М.* (прізвище та ініціали)

Рецензент (підпис) *Цуприк Г.Б.* (прізвище та ініціали)

Тернопіль
2021

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет *комп'ютерно-інформаційних систем і програмної інженерії*
(повна назва факультету)

Кафедра *комп'ютерних систем та мереж*
(повна назва кафедри)

ЗАТВЕРДЖУЮ
Завідувач кафедри
Осухівська Г.М.
(підпис) (прізвище та ініціали)
« » 2021 р.

З А В Д А Н Н Я

НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня *бакалавр*
(назва освітнього ступеня)

за спеціальністю *123 «Комп'ютерна інженерія»*
(шифр і назва спеціальності)

студенту *Горохівському Анатолію Володимировичу*
(прізвище, ім'я, по батькові)

1. Тема роботи *Комп'ютеризована система формування ціни на нерухомість з використанням хмарних сервісів*

Керівник роботи *Тиш Євгенія Володимирівна, к.т.н., доц.*
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від «*10*» *лютого* 2021 року № *4.7-97*

2. Термін подання студентом завершеної роботи *25.06.2021 р.*

3. Вихідні дані до роботи *Набір даних для прогнозування вартості нерухомості, фактори, які впливають на ціну, хмарні сервіси інтелектуального аналізу*

4. Зміст роботи (перелік питань, які потрібно розробити)
Вступ. 1. Аналіз технічного завдання та характеристик хмарних сервісів. 2. Архітектура комп'ютеризованої системи та модель прогнозування вартості об'єктів нерухомості. 3. Побудова і програмна реалізація моделі прогнозування ціни на нерухомість. 4. Безпека життєдіяльності, основи охорони праці. Висновки

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)
1. Загальні процеси та структура при побудові сервісів машинного навчання на основі Azure Machine Learning. 2. Інфраструктура Azure Machine Learning. 3. Архітектура комп'ютеризованої системи формування ціни на нерухомість 4. Підходи для розв'язку регресійних задач 5. Фрагмент вхідного набору даних і розподіл пропущених даних. 6. Результати прогнозування ціни на нерухомість

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Безпека життєдіяльності,	*Пилипець М.І., д.т.н., проф. каф. МТ*		
основи охорони праці			

7. Дата видачі завдання _____

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	*Розробка технічного завдання*	*10.02-16.02.2021*	
2	*Аналіз технічного завдання*	*17.02-02.03.2021*	
3	*Аналіз функціональності та вартості хмарних платформ*	*03.03-18.03.2021*	
4	*Проектування архітектури комп'ютеризованої системи*	*19.03-04.04.2021*	
5	*Обґрунтування вибору моделі прогнозування ціни на нерухомість*	*22.03-03.04.2021*	
6	*Проектування та реалізація програмної моделі формування ціни на нерухомість*	*04.04-02.05.2021*	
7	*Розробка інструкцій із встановлення та налаштування параметрів комп'ютеризованої системи*	*02.05-29.05.2021*	
8	*Безпека життєдіяльності, основи охорони праці*	*01.06-08.06.2021*	
9	*Оформлення кваліфікаційної роботи*	*09.06-18.06.2021*	
10	*Попередній захист кваліфікаційної роботи*	*18.06-22.06.2021*	
11	*Захист кваліфікаційної роботи*	*22.06-27.06.2021*	

Студент _____ *Горохівський Анатолій Володимирович*
(підпис) (прізвище та ініціали)

Керівник роботи _____ *Тиш Євгенія Володимирівна*
(підпис) (прізвище та ініціали)

АНОТАЦІЯ

Комп’ютеризована система формування ціни на нерухомість з використанням хмарних сервісів // Кваліфікаційна робота на здобуття освітнього ступеня бакалавр // Горохівський Анатолій Володимирович // ТНТУ, спеціальність 123 «Комп’ютерна інженерія»// Тернопіль, 2021 // с.– 75 , рис. – 31 , табл. – 6, аркушів А1 – 6, бібліогр. – 21.

Ключові слова: комп’ютеризована система, ціна, нерухомість, хмарний сервіс.

У даній роботі спроектовано комп’ютеризовану систему формування ціни на нерухомість із застосуванням хмарних сервісів та реалізовано інтелектуальний модуль, який на основі параметрів і характеристик об’єктів нерухомості дозволяє формувати рекомендації кінцевому користувача.

При розробці комп’ютеризованої системи формування ціни на нерухомість спроектовано її архітектуру, до складу якої входять два основних компоненти: веб-сайт або платформа з продажу нерухомості та хмарний сервіс, що забезпечує функціонування інтелектуальної складової формування ціни на нерухомість.

Практична реалізація інтелектуального модуля формування ціни на нерухомість містить аналіз вхідного набору даних, препроцесинг даних, «feature engineering» та реалізацію моделей на основі алгоритму XGBoost, Lasso, нейронної мережі, а також ансамблю XGBoost+Lasso. У результаті експериментальних досліджень досягнуто найкращого результату на основі метрики середньоквадратичного відхилення на рівні 0,11792.

ABSTRACT

Computer-aided system of estate property price formation using cloud services // Bachelor's work // Horokhivskiy Anatolii Volodymyrovych // TNTU, speciality 123 «Computer engineering»// Ternopil, 2021 // p.– 75 , fig. – 31 , tab. – 6, posters A1 – 6, ref. – 21.

Keywords: computer system, price, property, cloud service.

In this work, a computerized system of real estate pricing with the use of cloud services and implemented an intelligent module, which based on the parameters and characteristics of real estate allows you to make recommendations to the end user.

When developing a computerized real estate pricing system, its architecture was designed, which includes two main components: a website or real estate sales platform and a cloud service that ensures the functioning of the intellectual component of real estate pricing.

The practical implementation of the intelligent module of real estate pricing includes analysis of the input data set, data preprocessing, "feature engineering" and implementation of models based on the algorithm XGBoost, Lasso, neural network, as well as the ensemble XGBoost + Lasso. As a result of experimental studies, the best result was achieved on the basis of the standard deviation metric at the level of 0,11792.

ЗМІСТ

ПЕРЕЛІК ОСНОВНИХ УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ І СКОРОЧЕНЬ

БД	База даних
КС	Комп’ютеризована система
ПЗ	Програмне забезпечення
ER	Entity-Relationships
RMSE	Root Mean Square Error
UML	Unified Modelling Language

ВСТУП

Сучасні темпи розвитку різних галузей людської життєдіяльності характеризуються значним зростанням і застосуванням нових технологій підтримки як технологічних і виробничих процесів, так і організаційних та обслуговуючих. Галуззю, яка постійно розвивається і нарощує збут є сфера будівництва та реалізації об'єктів нерухомості. Вклад у нерухомість дозволяє зберегти заощадження від інфляційних процесів та забезпечити стабільний прибуток при оренді приміщень. Зважаючи на те, що даний сегмент економіки є розвинутим практично у всіх країнах світу, важливим завданням є автоматизація та підвищення його ефективності для забезпечення конкурентоспроможності гравців ринку.

На даний час доволі потужно використовуються засоби онлайн продажу об'єктів нерухомості, що дозволяє споживачам робити вибір саме того житла, яке задовольняє його потреби та бюджет не покидаючи своєї локації.

Однак насичення дилерського ринку у вигляді агенцій нерухомості при реалізації об'єктів нерухомості вимагає залучення все нових маркетингових інструментів та засобів підбору оптимального житла. Одним з таких інструментів є сервіс для формування ціни на нерухомість, який може бути корисний як для самих агенцій, так і для кінцевих споживачів.

Власне розв'язанню практичної задачі побудови комп'ютеризованої системи формування ціни на нерухомість із застосуванням хмарних сервісів присвячена дана кваліфікаційна робота. Використання хмарних сервісів дозволяє забезпечити гнучкість і масштабованість такого рішення і є доволі простим при інтеграції або міграції існуючого веб-ресурсу з продажу об''ктів нерухомості.

Актуальність побудови і впровадження такої системи полягає у здатності збільшити дохідну частину агенцій нерухомості з однієї сторони, та зменшити витрати на придбання житла кінцевим споживачем з іншої.

1 АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ ТА ХАРАКТЕРИСТИК ХМАРНИХ ПЛАТФОРМ

1.1 Аналіз вимог до комп'ютеризованої системи формування ціни на нерухомість

Комп'ютеризована система формування ціни на нерухомість з використанням хмарних сервісів призначена для прогнозування вартості житла з врахуванням факторів, які впливають на її значення. Для того, щоб спрогнозувати вартість нерухомості доцільно скористатись відкритими даними агенцій нерухомості, які функціонують на ринку в певному регіоні або країні. Система повинна враховувати особливості інфраструктури та місця розташування об'єкту нерухомості, його тип (квартира, котедж або житловий будинок), матеріал з якого виконано стіни, якість ремонту та ряд інших факторів.

Проектована комп'ютеризована система може бути реалізована у вигляді окремого повноцінного сервісу, або як складова підсистема існуючої інформаційної інфраструктури, наприклад, електронного магазину чи сайту агенції нерухомості.

Система повинна реалізовувати функціональність та володіти інтерфейсом для формування запитів щодо параметрів об'єкту нерухомості, а у відповідь на сформований запит генерувати результат щодо можливої його вартості.

До складу комп'ютеризованої системи повинен входити інтелектуальний модуль, що реалізує алгоритми машинного навчання та видає адекватні результати на запити користувачів. Користувачами, які зацікавлені у використанні такої системи є особи, які зацікавлені у купівлі або продажі житлової нерухомості. Оскільки, дані, якими оперують агенції нерухомості доволі громіздкі, то для цього доцільно скористатись хмарними сервісами.

Доцільність застосування таких сервісів пов’язана з уникненням необхідності залучення фахівців з налаштування локальної інфраструктури при моделюванні роботи інтелектуальної складової системи, а також сховища для зберігання даних. Окрім цього, наявність готових сервісів машинного навчання в cloud дає змогу використовувати апробовані та найкращі рішення від провідних IT компаній світу.

Метою створення комп’ютеризованої системи формування ціни на нерухомості є автоматизація процесу надання додаткових послуг щодо оцінювання вартості об’єктів житлової нерухомості, виявлення важливих факторів, які впливають на значення ціни, а також, як наслідок, зростання цільової аудиторії відповідних агенцій.

Досягнення поставленої мети кваліфікаційної роботи можливе шляхом розв’язання наступних завдань:

– аналіз особливостей функціонування агенцій нерухомості та існуючих засобів автоматизації щодо підтримки продажу об’єктів нерухомості;

– аналіз параметрів, якими описуються об’єкти житлової нерухомості та визначити найбільш важливі з них, що впливають на вартість;

– обґрунтувати застосування хмарних сервісів для реалізації інтелектуальної підсистеми формування ціни на нерухомості;

– виконати аналіз і препроцесинг даних при описі властивостей нерухомості;

– розробити програмну модель для прогнозування ціни на нерухомість;

– спроектувати архітектуру інтелектуального сервісу прогнозування ціни на нерухомість та визначити можливі шляхи його інтеграції з існуючими системами;

– забезпечити точність прогнозування ціни та стійкість результатів.

Основними функціями комп’ютеризованої системи формування ціни на нерухомість є збір, аналіз і прогнозування вартості завершеного житлового будівництва, що давало б можливість адекватно приймати рішення щодо купівлі або продажу цих об’єктів.

До основних функцій, які виконує система, належать організація взаємодії сховищ даних з інтелектуальними хмарними сервісами для формування вартості нерухомості з врахуванням тенденцій ринку, а також забезпечення прийнятної точності щодо значень ціни. Комп'ютеризована система може бути організована у вигляді сервісу та інтегруватись з іншими суміжними системами.

Для автоматизації процесу інтелектуального формування ціни на нерухомість доцільно використовувати наступні структурні компоненти:

– сховище даних з характеристиками нерухомості, що продається;

– модуль імпорту/експорту даних;

– сервер передачі та опрацювання даних.

При моделюванні і реалізації інтелектуальної складової комп'ютеризованої системи необхідна наявність таких програмних компонентів:

– файл з даними у форматі csv або іншому, що підтримується мовою програмування Python;

– наявність програмних компонентів та бібліотек з підтримкою алгоритмів для розв'язання задач регресії;

– програмні інструменти для формування контейнерів.

Важливим аспектом при проектуванні комп'ютеризованої системи є залучення засобів моделювання таких як UML, зокрема при побудові use case діаграм, діаграм компонентів і класів.

Комп'ютеризована система формування ціни на нерухомість повинна забезпечити ефективність реалізації процесів характерних для агенцій нерухомості, шляхом надання менеджерам і клієнтам додаткових функцій, і як наслідок призвести до зростання продажів об'єктів нерухомості.

Комп'ютеризована система формування ціни на нерухомість із застосування хмарних сервісів повинна видавати адекватні рекомендації щодо вартості житла із врахуванням доступних факторів, які можна одержати при його метаописі, і тенденцій ринку нерухомості у конкретно взятому регіоні. У загальному випадку, вимоги, які висуваються до проектованої комп'ютеризованої системи можна сформулювати наступним чином:

– можливість зчитування і фільтрації вхідного набору даних щодо пропозицій житла на ринку;

– можливість визначення рівня впливу одних характеристик житла на цільову змінну;

– визначення найбільш важливих характеристик, які впливають на ціну об’єктів нерухомості;

– застосування алгоритмів розв’язання задач регресії з високою точністю;

– видача найбільш релевантних результатів щодо ціни нерухомості;

– продуктивність видачі результатів прогнозування вартості житла;

– можливість налаштування гіперпараметрів моделі при прогнозуванні ціни.

До структури комп’ютеризованої системи формування ціни на нерухомість висуваються такі основні вимоги:

– наявність компоненти одержання даних з відповідного джерела за заданим URL;

– наявність інтелектуального модуля прогнозування вартості нерухомості на основі хмарних сервісів;

– зручний користувацький відображення рекомендації щодо можливої вартості житлового об’єкта;

– можливість інтеграції з електронними ресурсами і платформами управління процесами продажу нерухомості;

– наявність авторизованого доступу бази даних продажу нерухомості.

В загальному випадку, вимоги до комп’ютеризованої системи формування ціни на нерухомість повинні задовольняти висловленим раніше вимогам, а також відображати процеси щодо прогнозування тенденцій на ринку нерухомості. Дана система повинна бути розгорнута у хмарному сховищі і використовувати модель реалізовану мовою програмування Python або інструменти машинного навчання обраної платформи.

До основних функціональних вимог, які повинна реалізовувати проектована комп’ютеризована система належать:

– здатність імпорту/експорту даних із визначеного джерела даних;

– можливість виявляти пропущені та некоректні дані;

– здатність до відновлення або видалення пропущених даних;

– здатність формувати матрицю кореляцій між параметрами об’єктів нерухомості;

– здатність реалізовувати алгоритми прогнозування вартості нерухомості на основі кращих моделей розв’язання задач регресії;

– можливість візуалізації важливих залежностей та типів розподілів даних;

– здатність кількісного оцінювання якості прогнозованих значень щодо ціни нерухомості;

– можливість інтеграції із суміжними системами по типу електронних магазинів спеціального призначення (продажу об’єктів нерухомості).

Комунікація та способи зв’язку між структурними компонентами комп’ютеризованої системи формування ціни на нерухомість повинні відповідати вимогам до взаємодії структурних елементів комп’ютерних систем, які функціонують на основі технології клієнт-сервер.

З однієї сторони комп’ютеризована система формування ціни на нерухомість виступає у вигляді клієнта – при зчитуванні даних, а в іншому випадку – серверною частиною – при зверненні зовнішніх систем для формування рекомендацій щодо вартості об’єкту нерухомості.

Усі схеми комунікації між структурними компонентами комп’ютеризованої системи виконуються на основі протоколів комп’ютерних мереж та протоколів HTTP/HTTPS.

Діагностування коректності функціонування комп’ютеризованої системи формування ціни на нерухомість повинна відбуватись у відповідності до графіку профілактичних заходів системи управління продажами об’єктів нерухомості, або при виникненні нештатних ситуацій чи збоїв. Окрім цього, діагностика

системи повинна проводитись у випадку зниження точності прогнозування вартості нерухомості або зміни структури вхідного набору даних.

Перспективами розвитку і застосування комп'ютеризованої системи формування ціни на нерухомість є можливість гнучкого налаштування використовуваних апаратних ресурсів при виконанні операцій з прогнозування вартості житлових об'єктів, а також універсальність взаємодії з іншими системами.

Модернізація комп'ютеризованої системи можлива у випадку зміни структури вхідного набору даних, що впливатиме на додаткові фактори ціноутворення, а також при необхідності міграції на інші платформи машинного навчання, які відносяться до класу Low/No code.

Модернізація або утилізація комп'ютеризованої системи виконується у випадку морального старіння технологій та не відповідності стандартам новоутворених технологій і засобів управління бізнес-процесами на ринку нерухомості.

Вимоги, що висуваються до надійності функціонування програмного і апаратного забезпечення комп'ютеризованої системи формування ціни на нерухомість належать:

Безвідмовність роботи протягом часу, визначеного надійністю платформи на якій функціонує хмарний сервіс (зазвичай становить 99,999%) та зовнішньої системи управління бізнес-процесами ринку нерухомості;

Здатність системи відновлювати свою працездатність після виникнення непередбачуваних та нештатних ситуацій

Функції захищеності комп'ютеризованої системи від несанкціонованого втручання повинні виконуватись на інфраструктурному рівні платформ або хостів, де розміщені дані та сама система прогнозування ціни на нерухомість.

Основними вимогами до функцій, які виконує комп'ютеризована система формування ціни на нерухомість належать :

– здатність виконувати читання файлів у форматі csv та можливості експорту з реляційних баз даних;

– можливість віддаленого управління та фільтрації інформації щодо критеріїв, які формують ціну об’єктів нерухомості;

– можливість керувати гіперпараметрами моделі прогнозування при моделювання роботи комп’ютеризованої системи;

– здатність видавати результати прогнозування ціни на нерухомість;

– можливість забезпечувати точність та адекватність формування ціни з врахуванням тенденцій ринку нерухомості конкретного регіону;

– здатність використовувати метрики якості при оцінюванні результатів прогнозування;

– наявність механізмів захисту на різних рівнях використання системи;

– забезпечення простоти і зручності експлуатації комп’ютеризованої системи;

– здатність до співіснування із визначеними класами систем;

– можливість використання та налаштування параметрів хмарних сервісів.

Мінімальні вимоги, які висуваються до апаратних характеристик пристроїв для нормального режиму функціонування комп’ютеризованої системи формування ціни на нерухомість:

– тактова частота процесора – 2,0 ГГц з 8-ма паралельними потоками;

– об’єм оперативної пам’яті – 32 ГБ;

– об’єм жорсткого диску – 4Тб.

Вимоги до апаратного забезпечення клієнтських станцій:

– тактова частота процесора – 2,0 ГГц з 4-ма паралельними потоками;

– об’єм оперативної пам’яті – 8 ГБ;

– об’єм жорсткого диску – 1Тб.

Вимоги до програмного забезпечення клієнтських станцій – операційна система будь-якого типу, наявність браузера.

Вимоги до програмного забезпечення сервера – визначаються програмною екосистемою платформи машинного навчання.

1.2 Аналіз хмарних платформ і сервісів при проектуванні комп'ютеризованої системи формування ціни на нерухомість

В загальному випадку, хмарна платформа представляє собою сукупність сервісів і повноважень, які надають їхні розробники. Серед видів послуг, які пропонуються інженерам з проектування комп'ютерних систем, можна виділити такі як: доступ до обчислювальних ресурсів і аналітичних інструментів, використання сховища даних, серверів, програмного забезпечення і т.д.

Перш, ніж проводити порівняння існуючих лідерів ринку хмарних провайдерів (рис. 1.1), потрібно ознайомитись з описом кожного з них окремо.

Рисунок 1.1 – Провайдери хмарних сервісів

Amazon Web Services заснований у 2006 році і станом на зараз надає такі послуги, як IaaS, PaaS, SaaS і ін. Він також пропонує більше 70 ресурсів з розширеною зоною покриття в чотирнадцяти регіонах світу.

Azure – це продукт Microsoft, випущений у 2010 році. Сьогодні платформа пропонує широкий вибір різних допоміжних інструментів, мов програмування і фреймворків. Він працює на основі платформ операційних систем Microsoft Windows і Linux. У даний час на платформі наявні близько 60 сервісів і центрів обробки даних в більш ніж 38 точках світу. Серед клієнтів Azure такі відомі компанії, як Johnson Controls, Fujifilm, HP, Apple та ін.

Google Cloud Platform – наймолодша хмарна платформа із згаданих вище – вона була запущена у 2011 році і пропонує багато послуг, включаючи IaaS, PaaS і Serverless, а також підтримує Big data і IoT. У розпорядженні провайдерів більше 50 ресурсів і 6 глобальних центрів обробки даних.

Проведемо порівняльний аналіз технологій, які використовують на своїх платформах наведені вище провайдери для організації безпеки зберігання даних.

AWS використовує Simple Storage Service (S3) для зберігання інформації і сервіс Amazon Glacier для архівування. Платформи Azure і Google Cloud пропонують сховища з високою продуктивністю і надійним захистом. Компанія Azure активно працює над впровадженням і поліпшенням функцій резервного копіювання та відновлення файлів. Сервіс StorSimple, який використовується в якості гібридного сервісу зберігання даних для корпоративних клієнтів, значно підвищив ефективність компанії. Це дозволяє економити до 60% від звичайної вартості.

Гібридний підхід – одна з останніх тенденцій, яка досить швидко розвивається у світі хмарних платформ. Microsoft вже давно визнана кращою опцією, якщо обирати цей метод серед конкурентів AWS, Google Cloud і Microsoft Azure, завдяки своєму програмному рішенню Azure Stack. Платформа надає своїм клієнтам можливість розміщувати хмарні сервіси Azure на локальних сервісах обробки даних з відкритим порталом, кодом і API-інтерфейсами з метою забезпечення простої інтеграції та сумісності.

Платформа AWS показала свій перехід до гібридного розгортання ще в 2018 році. Пізніше, в 2019 році, менеджери Google зробили крок в тому ж напрямі, випустивши свою платформу Anthos. Anthos – це, по суті, ребрендинг платформи хмарних сервісів Google, що включає Google Kubernetes Engine (GKE), GKE On-Prem і Anthos Config Management.

Основні характеристики платформи Amazon наведена у таблиці 1.1.

Таблиця 1.1 – Характеристики PaaS-рішення на основі Amazon EMR

PaaS-рішення	Склад екосистеми	Надійність і безпека	Вартість
Amazon EMR, інтеграція зі всіма веб-сервісами Amazon	Apache Hadoop 2.x, Hive, Pig, HBase, Impala, Spark, Tez, Oozie, Flink, Zeppelin, Hue, Presto, HCatalog, Machout, MXNet, Sqoop, підтримка мов програмування Scala, Pig, R, Python, SQL, HiveQL.	Політики доступу, ведення журналів і аудиту на рівні аккаунта і об'єктів. Захищені протоколи HTTPS і SSH, аутентифікація і шифрування даних. Безперервний моніторинг операцій щодо доступу до даних, виявлення відхилень за допомогою алгоритмів машинного навчання і генерація сповіщень при виявленні ризиків несанкціонованого доступу. Зберігання даних у датацентрах на території США.	Плата нараховується на основі щосекундного тарифу, мінімальний рівень оплати складає одну хвилину. Вартість Amazon EMR складає 0,015$ за годину на один вузол додатково до вартості використання Amazon EC2.

Наявність датацентрів компанії Amazon охоплює доволі широкий спектр локацій, які показано на рис. 1.1.

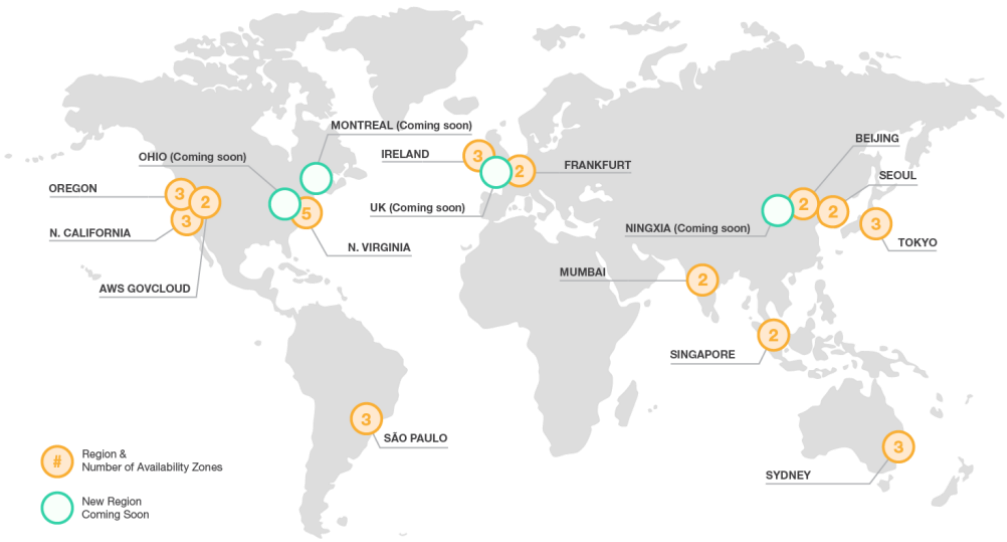


Рисунок 1.1 – Локації датацентрів компанії Amazon

Особливості використання, технічні і вартісні характеристики платформи HDInsight (Azure) наведено у таблиці 1.2.

Таблиця 1.2 – Характеристики платформи HDInsight

PaaS-рішення	Склад екосистеми	Надійність і безпека	Вартість
HDInsight (Microsoft), інтеграція з усіма службами Microsoft Azure, Active Directory і Apache Ranger з пакетом безпеки корпоративного рівня	Apache Hadoop 2.x, Spark, Kafka, HBase, Storm. Інтерактивний запит, служби машинного навчання HDInsight	Підтримка захищеного шлюза у віртуальній мережі Azure. Управління доступом на основі ролей і шифрування у службі сховища. Зберігання даних у датацентрах на території США, Європи і Азії.	Плата за використання кластерів нараховується щохвилинно. Вузли розрізняються в залежності від групи, кількості і типу екземпляра.

При використанні HDInsight вартість використання одного вузла інфраструктури за годину коливається від 0,14$ до 4,09$. На рис. 1.2 показано присутність датацентрів Azure у світі.

Рисунок 1.2 – Розташування датацентрів Azure

Ще однією платформою, що містить сервіси для побудови інтелектуальних сервісів є платформа, розроблена компанією Google. На рис. 1.3. показано розташування її датацентрів, а у табл. 1.3 – основні технічні характеристики і особливості її застосування.

Рисунок 1.3 – Розташування датацентрів компанії Google

Таблиця 1.3 – Платформа Google Cloud Platform

PaaS-рішення	Структура екосистеми	Надійність і безпека	Вартість
Dataproc (Google Cloud Platform), інтеграція зі всіми веб-сервісами Google	Apache Hadoop 2.x, Spark, Hive, Pig, ZooKeeper, Zeppelin, Presto	Аутентифікація користувачів, ізоляція і шифрування даних з використанням протоколу Kerberos. Керування доступом на основі ролей і груп.	Погодинна тарифікація від $0.01 до $1,640 за вузол, в залежності від його конфігурації.

До провайдерів PaaS рішень належить також компанія IBM, характеристики платформи якої наведено у табл. 1.4.

Таблиця 1.4 – Платформа Analytics Engine (IBM)

PaaS-рішення	Структура екосистеми	Надійність і безпека	Вартість
Analytics Engine (IBM), інтеграція зі сховищем даних IBM Cloud Object Storage та іншими сервісами IBM, зокрема, Watson™ Studio і Machine Learning	Apache Hadoop 2.x, Livy, Knox, Spark, JEG, Ambari, Anaconda Py, Hive, HBase, Phoenix, Oozie	Аутентифікація користувачів, ізоляція і шифрування даних (SSL, REST). Зберігання інформації у датацентрах США, Європи і Азії.	Погодинна тарифікація від $0,7 до $2,640 за вузол, в залежності від його конфігурації.

Для Analytics Engine характерним є наявність безкоштовної версії з обмеженнями кількості годин та вузлів.

Вище наведені таблиці дозволяють зробити наступні висновки:

– найбільш широкий набір вбудованих засобів для Big Data і Machine Learning містять рішення від Amazon Web Services (AWS);

– кожен провайдер оголошує безперебійну доступність кластера і веб-сервісів - вище 99% за угодою про рівень надання послуги за рахунок захищених протоколів обміну даними, резервування каналів передачі інформації, шифрування SSH, ізоляції даних, аутентифікації і гнучких налаштувань політики безпеки на основі ролей;

– ЦОДи більшості провайдерів розташовані у США, Європі та Азії.

– практично всі провайдери перейшли на ціноутворення за спожиті ресурси, тарифікація їх використання відбувається на основі щосекундних, щохвилинних або погодинних тарифів, однак, наприклад, при використанні AWS до цих витрат додається вартість самого продукту (Amazon EC2);

– серед проаналізованих рішень AWS і Microsoft Azure вважаються найбільш затребуваними хмарними платформами в корпоративному секторі.

Таким чином, є доволі широкий вибір щодо реалізації комп’ютеризованої системи формування ціни на нерухомість із використанням хмарних сервісів, однак у роботі пропонується використати Microsoft Azure Learning Service.

2 АРХІТЕКТУРА КОМП'ЮТЕРИЗОВАНОЇ СИСТЕМИ ТА МОДЕЛЬ ПРОГНОЗУВАННЯ ВАРТОСТІ ОБ'ЄКТІВ НЕРУХОМОСТІ

2.1 Проектування структури комп'ютеризованої системи формування ціни на нерухомість з використанням хмарних сервісів

Для проектування структури комп'ютеризованої системи формування ціни на нерухомість із застосуванням хмарних сервісів, в першу чергу необхідно провести аналіз функціональності обраного хмарного сервісу та спроектувати його архітектуру.

При використанні Microsoft Azure Learning Service, необхідно врахувати загальний процес та його особливості при побудові інтелектуальних рішень. В цілому, процес побудови комп'ютерної системи формування ціни на нерухомості у вигляді веб-сервісу показано на рис. 2.1.

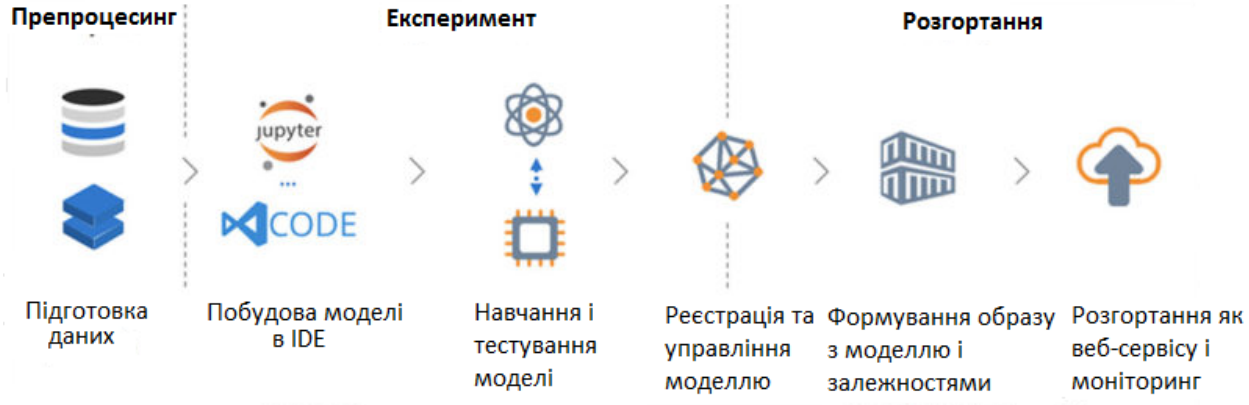


Рисунок 2.1 – Загальний процес побудови комп'ютеризованої системи формування ціни на нерухомість з використанням Azure Machine Learning

Більш детальна схема архітектури комп'ютеризованої системи формування ціни на нерухомість з використанням хмарного сервісу показана на рис. 2.2.

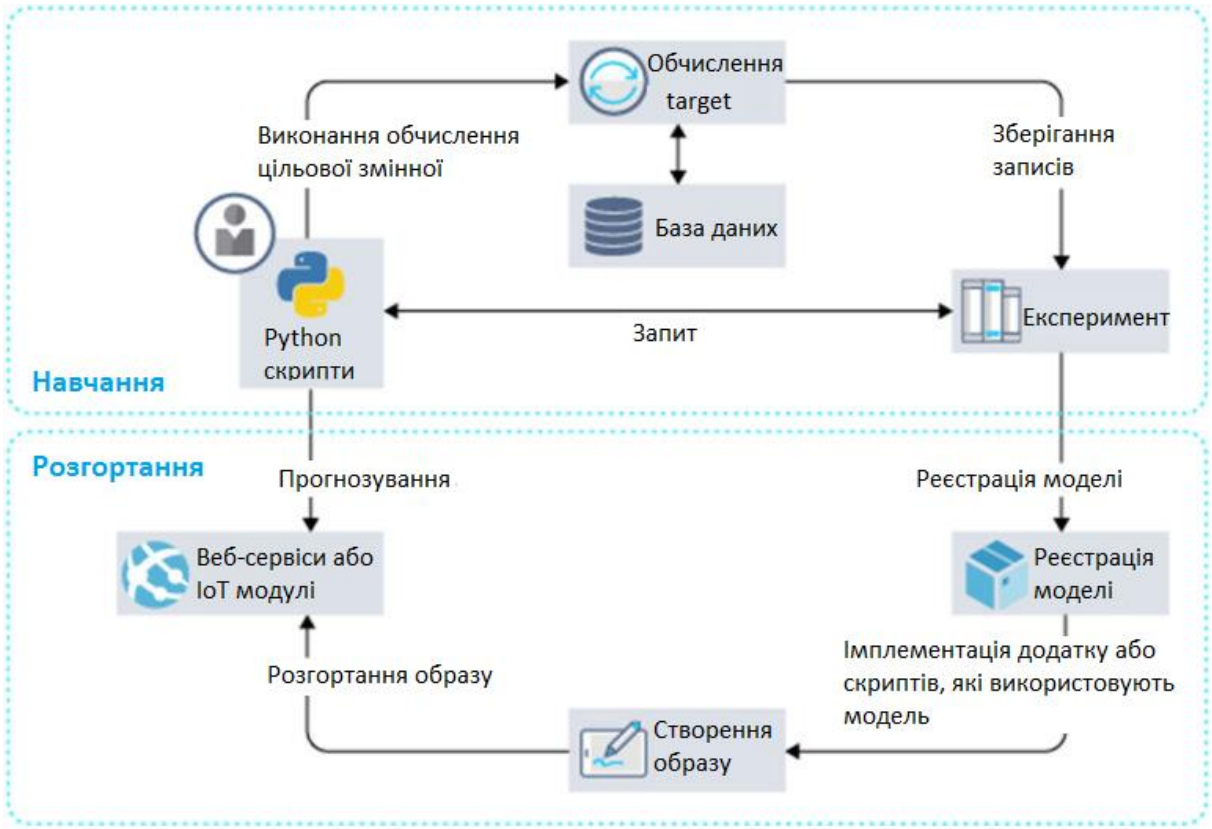


Рисунок 2.2 – Детальна архітектура комп'ютеризованої системи формування ціни на нерухомість

Як видно з рис. 2.2 платформа машинного навчання Azure підтримує реалізацію моделі прогнозування ціни на нерухомість засобами мови програмування Python. Це означає, що моделювання та гіпероптимізацію параметрів моделі прогнозування ціни можна виконувати на локальній машині, а після цього розгорнути у хмарному сховищі.

Реалізацію архітектури комп'ютерної системи формування ціни на нерухомість з використанням хмарних сервісів умовно можна поділити на дві частини:

– навчання (тренування) моделі – ітераційний процес, що передбачає використання вхідного набору даних та алгоритмів машинного навчання для

визначення оптимальних параметрів моделі прогнозування вартості нерухомості;

– розгортання інтелектуального модуля – сукупність процесів, які ведуть до створення веб-сервісу для вирішення регресійної задачі прогнозування ціни на нерухомість.

Структурно організацію інфраструктури платформи Azure Machine Learning показано на рис. 2.3.

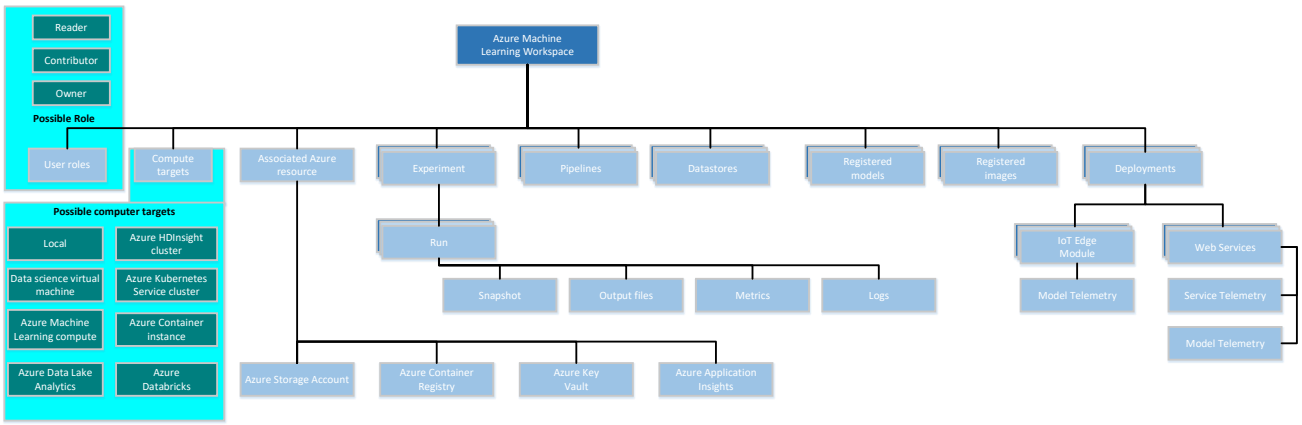


Рисунок 2.3 – Інфраструктура при побудові комп'ютеризованої системи формування ціни на нерухомість з використанням Azure Machine Learning

Ефективність застосування комп'ютеризованої системи може бути досягнута шляхом її інтеграції із зовнішніми сервісами або веб-сайтами. Так, наприклад, агенції нерухомості або відповідні платформи для розширення бази даних клієнтів інтенсивно використовують веб-простір.

Реалізація об'єктів нерухомості здійснюється на спеціалізованих веб-додатках, які по суті, створені на основі готових шаблонів і платформ електронної комерції.

Найбільш широко популярною платформою з продажу житлової і не житлової нерухомості в Україні є https://dom.ria.com/. В інших країнах наявні свої платформи, які дозволяють також проводити операції з нерухомістю.

Для кожної з цих платформ важливим є їхній розвиток, який можна реалізувати шляхом інтеграції рекомендаційних сервісів. Власне комп'ютеризована система формування ціни на нерухомість є тією системою

побудови рекомендацій, що допомагає продавцю зорієнтуватися у цінах на ринку і сформувати її конкуренто здатну вартість не затрачаючи часових і відповідно фінансових ресурсів. На рис. 2.5 показано загальний вигляд інтерфейсу платформи dom.ria для подачі і пошуку оголошень щодо купівлі-продажу нерухомості.

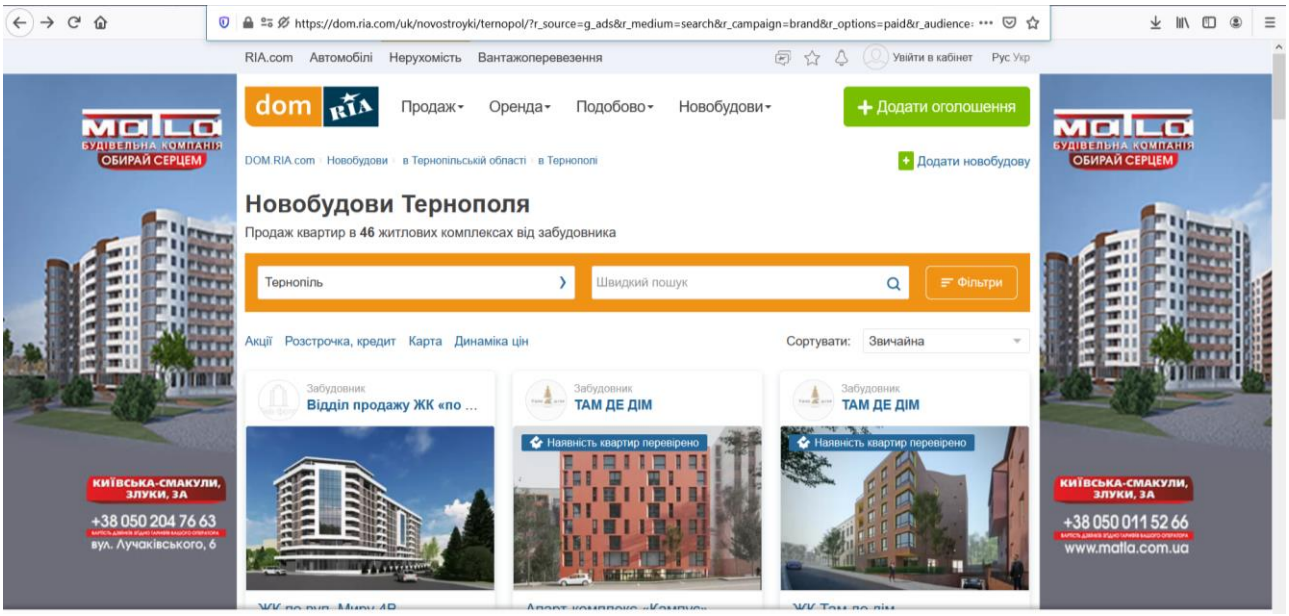


Рисунок 2.5 – Інтерфейс платформи з продажу об’єктів нерухомості

На даному сайті можна розрахувати орієнтовну вартість об’єкту нерухомості на основі критеріїв, які показано на рис. 2.6.

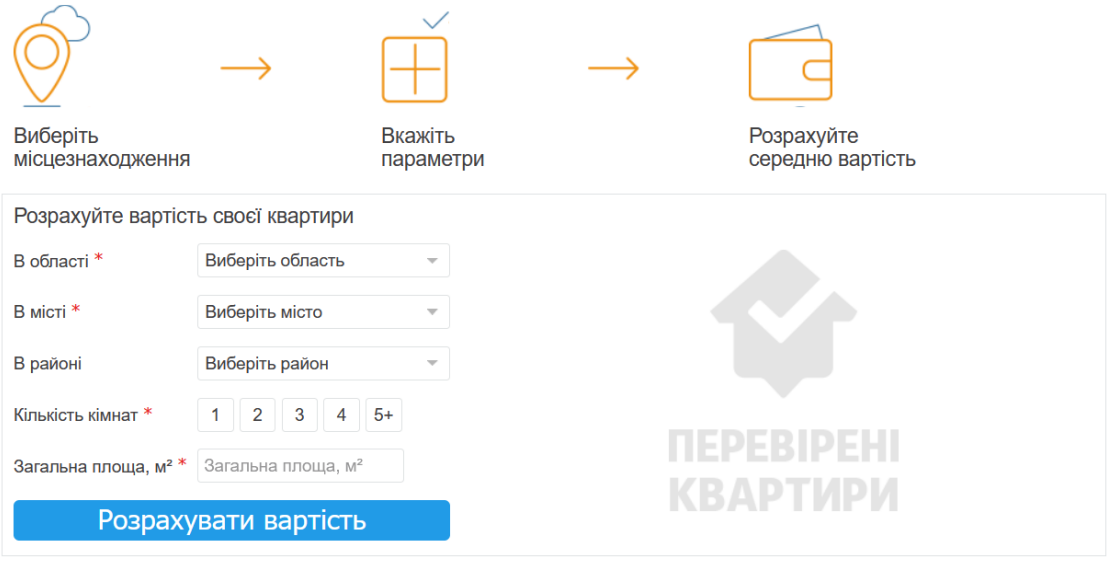


Рисунок 2.6 – Формування ціни нерухомості на сайті dom.ria

Як видно з рис. 2.6, при формуванні рекомендованої ціни на нерухомість враховується лише 5 параметрів:

– область;
– місто;
– район;
– кількість кімнат;
– загальна площа.

Наведені вище параметри недостатньо в повній мірі описують фактори, які є визначальними при формуванні рекомендації ціни, хоча при додаванні оголошення (рис. 2.7) можна вказати значно більше критеріїв опису об’єкту нерухомості.

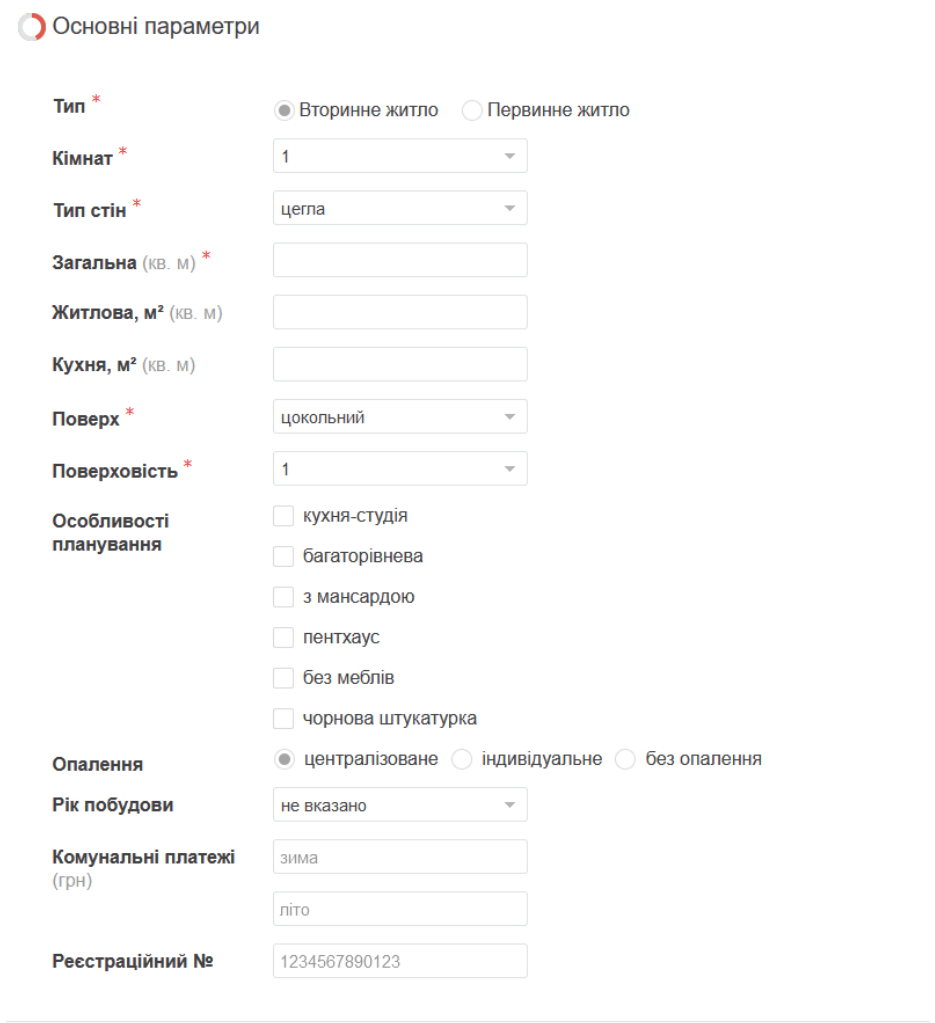


Рисунок 2.7 – Форма для створення оголошення про продаж нерухомості

Таким чином, для таких платформ з продажу нерухомості доцільним є формування інтелектуального модуля, або цілої комп'ютеризованої системи, яка б давала змогу враховувати кореляцію між характеристиками об'єкту нерухомості та видавати більш точні результати щодо рекомендованої ціни.

В загальному випадку, при проектуванні архітектури комп'ютеризованої системи прогнозування ціни на нерухомість, структуру компонентів і зв'язків між ними пропонується побудувати, як показано на рис. 2.8.

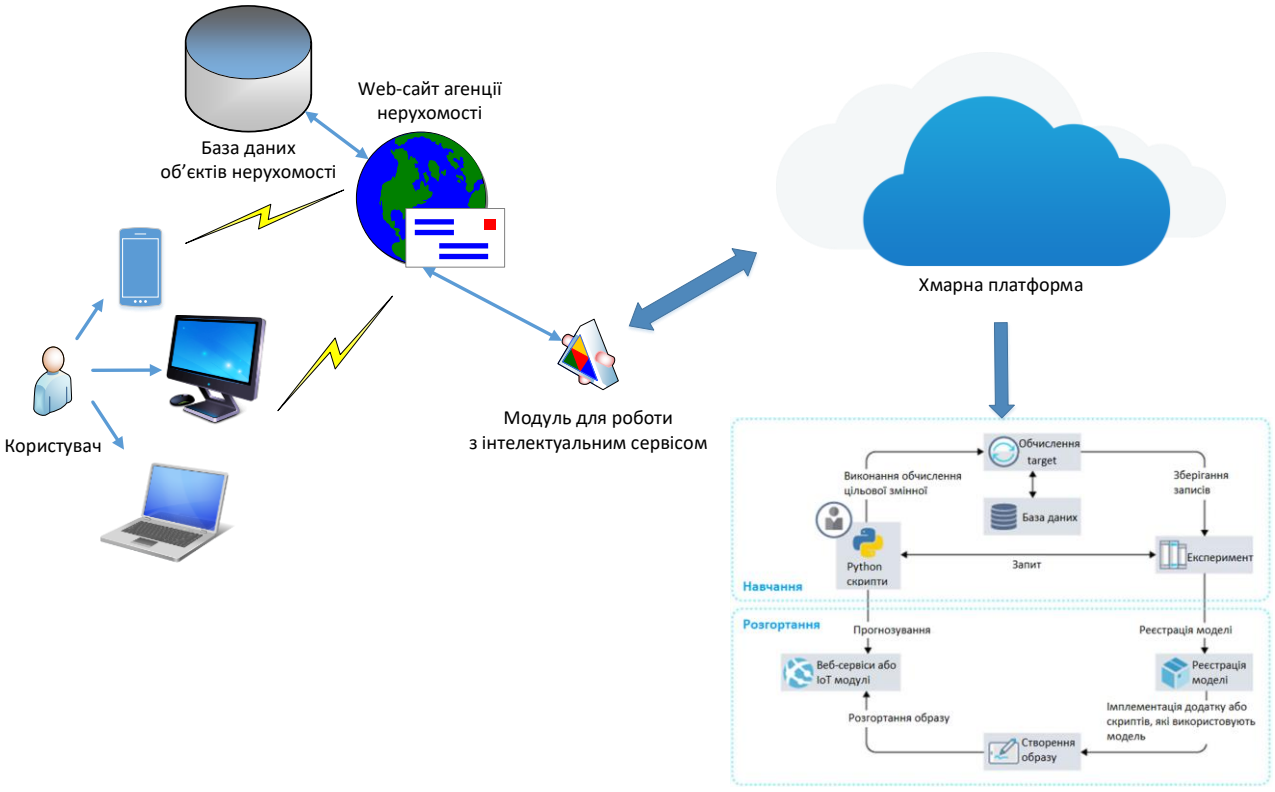


Рисунок 2.8 – Архітектура комп'ютеризованої системи формування ціни на нерухомість з використанням хмарних сервісів

До складу комп'ютеризованої системи входять два основних компоненти:

– веб-сайт або платформа з продажу нерухомості – забезпечує функціональність щодо збору та зберігання даних про об'єкти нерухомості.

– хмарний сервіс – програмний комплекс, що забезпечує функціонування інтелектуальної складової формування ціни на нерухомість.

Веб сайт є точкою входу для користувачів системи і надає інтерфейс для формування оголошення, виконання пошуку і фільтрації наявної у базі даних

інформації. Окрім, цього дані користувача та характеристика об’єктів продажу нерухомості зберігаються у базі даних з можливістю експорту у різні формати, зокрема, XML, Excel, csv формат.

Ввівши параметри об’єкту нерухомості, інформація потрапляє у базу даних. Після цього, користувачу при виборі функції рекомендованої ціни, повинен бути виданий результат щодо ціни за яку він може продати або придбати нерухомість. За цей функціонал відповідає модуль для роботи з інтелектуальним сервісом, що забезпечує прямий і зворотній трансфер рекомендацій щодо вартості нерухомості. Безпосередньо формування ціни відбувається на основі алгоритмів і моделі прогнозування. Далі потрібно провести аналіз оптимальних моделей і алгоритмів, що вирішують регресійну задачу прогнозування ціни.

2.2 Моделі та алгоритми розв’язку регресійних задач

При розв’язані задач регресії за допомогою підходів машинного навчання використовується ряд моделей та алгоритмів. Оскільки, задача формування ціни на нерухомість належить саме цього класу (прогнозується значення цільової змінної «ціна», область визначення якої належить деякому інтервалу), тому потрібно розглянути та проаналізувати найбільш поширені та ефективні підходи.

2.2.1 Підхід до прогнозування ціни на основі лінійної регресії

Перед тим, як перейти до можливостей безпосереднього застосування лінійної регресії потрібно провести аналіз таких понять, як: інтерполяція, апроксимація та регресія. Характерною особливістю цих понять є те, що у них загальна спільна мета: із сімейства функцій вибрати ту, яка володіє певним набором властивостей.

Суть інтерполяції полягає у здатності вибрати з сімейства функцій ту, яка проходить через задані точки. Часто функція потім використовується для обчислення значення у проміжних точках, через які проходить функція. Для

прикладу, задано колір кількох точок і потрібно зробити так, щоб кольори інших точок утворили плавні переходи між заданими.

До класичних прикладів застосування інтерполяції належать:

– інтерполяція із застосуванням поліномів Лагранжа,

– сплайн-інтерполяція,

– багатовимірна інтерполяція (білінійна, метод найближчих сусідів та ін.).

Візуалізований приклад застосування інтерполяції показано на рис. 2.9.

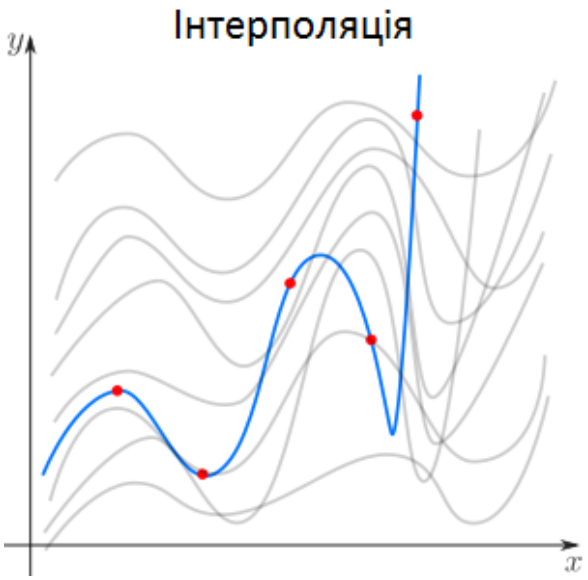


Рисунок 2.9 – Приклад інтерполяції

Суть екстраполяції полягає у прогнозуванні поведінки функції поза межами заданого інтервалу. До практичних прикладів застосування екстраполяції належать прогнозування курсу долара на основі попередніх його значень.

Апроксимація передбачає дослідження способу вибрати з сімейства «простих» функцій найбільш наближену для представлення «складної» функції на деякому інтервалі області визначення. При цьому помилка не повинна перевищувати деякої встановленої границі. Апроксимацію використовують, у тих випадках, коли потрібно отримати функцію, схожу на задану, але більш зручну для виконання обчислень і маніпуляцій (диференціювання, інтегрування

і т.п). При оптимізації критичних ділянок програмного коду часто використовують апроксимацію: якщо значення функції обчислюється багато разів в секунду і не потрібна абсолютна точність, то можна обійтися простішим апроксимантом з меншою «вартістю» обчислення. Класичні приклади апрокстмації: ряд Тейлора на інтервалі, апроксимація ортогональними многочленами, апроксимацію Паде, апроксимація синуса Бхаськара і т.п. На рис. 2.10 показано приклад застосування екстраполяції.

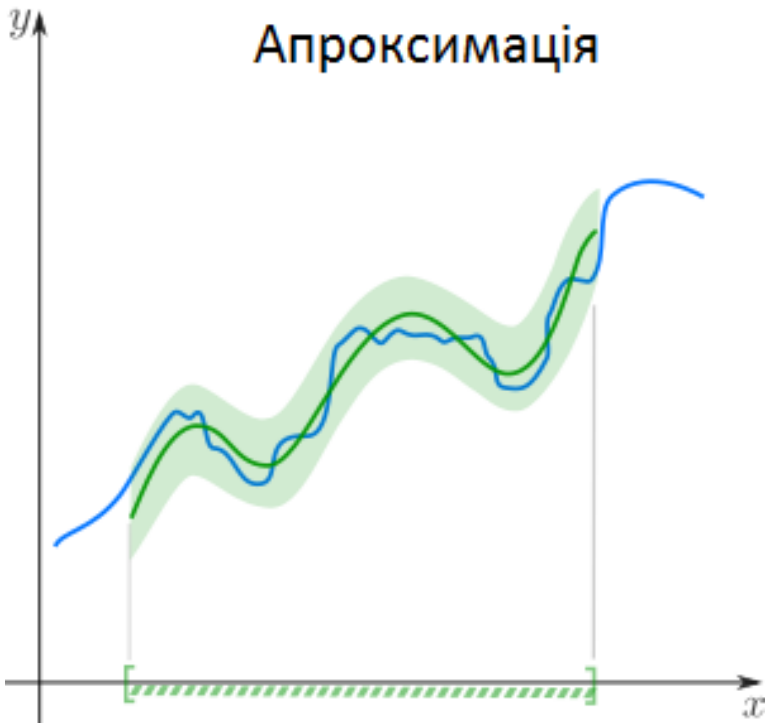


Рисунок 2.10 – Приклад апроксимації

Регресія представляє собою підхід до вибору з сімейства функцій тієї, яка мінімізує функцію втрат. Функція втрат характеризує наскільки сильно модельована функція відхиляється від значень у заданих точках. Якщо точки одержані у результаті експерименту, обов'язково містять помилку вимірювань, шум, тому більш доцільно, щоб функція відображала загальний тренд, а не точно проходила через всі них. В якомусь сенсі регресія – це «інтерполююча апроксимація»: потрібно провести криву якомога ближче до точок і при цьому зберегти її максимально простою, щоб визначити загальний тренд. За баланс між цими суперечливими бажаннями якраз відповідає функція втрат (в англійській

літературі «loss function» або «cost function»). Приклад функції на основі лінійної регресії показано на рис. 2.11.

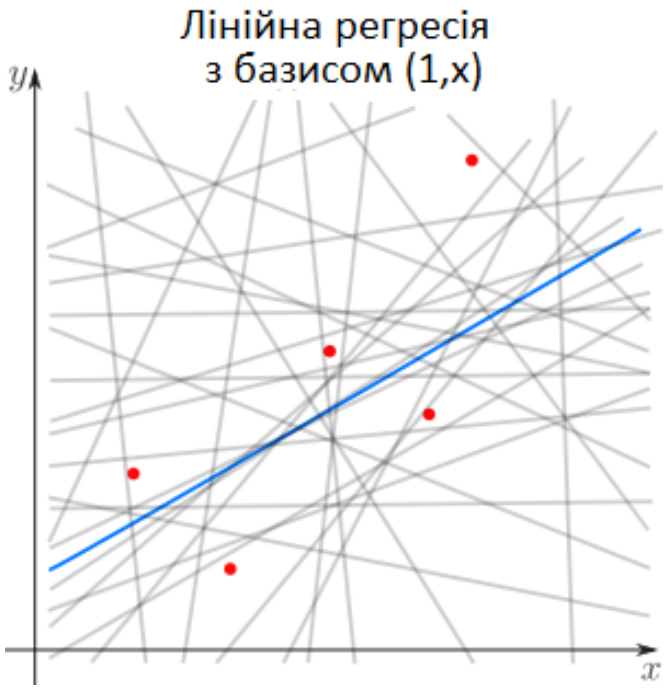


Рисунок 2.11 – Візуалізація лінійної регресії

Основна мета при застосуванні лінійної регресії полягає у знаходженні коефіцієнтів лінійної комбінації (коефіцієнтів рівняння прямої), і тим самим визначити регресійну функцію (яку також називають моделлю). Лінійну регресію називають лінійною саме через лінійну комбінацію базисних функцій - це не пов'язано з самими базовими функціями (вони можуть бути лінійними або ні). Існує багато варіантів і узагальнень лінійної регресії: LAD, метод найменших квадратів, Ridge регресія, Lasso регресія, ElasticNet та багато інших.

До переваг лінійної регресії належить:

– швидке моделювання (особливо на невеликих наборах даних);

– лінійну регресію легко зрозуміти, що є цінним для різних бізнес-рішень.

До недоліків лінійної регресії належать:

– у разі нелінійних даних, поліноміальну регресію важко спроектувати.

– необхідно мати інформацію про структуру даних і взаємозв'язку між змінними.

Виходячи з викладених вище фактів, лінійна регресія не є ефективним інструментом, коли мова потрібно побудувати модуль прогнозування на складних даних і великих їх обсягах.

2.2.2 Підхід на основі дерев прийняття рішень

Дерева прийняття рішень і випадковий ліс (Random Forest) є представниками деревовидного методу у машинному навчанні. Дерева рішень представляють собою моделі, які використовуються для розв’язку задач прогнозування шляхом циклічного перегляду кожної функції в наборі даних послідовно одна за одною.

Випадковий ліс – це ансамбль (комітет) дерев рішення, які використовують випадкові порядки об’єктів у наборі даних. Щ

Розглянемо приклад, коли спортсмен грає у баскетбол щопонеділка. Кожен раз він запрошує одного і того ж друга. Іноді друг дійсно приходить, іноді ні. Рішення приходити чи ні залежить від сукупності факторів: погодні умови, температура, вітер і втома. Спортсмен починає помічати ці фактори і відстежувати їх разом з рішенням друга грати чи ні. Ці дані можна використовувати для прогнозування рішення друга. При цьому можна скористатись технікою дерев прийняття рішень (рис. 2.12).

У кожного дерева рішень є 2 типи елементів:

– вузли (Nodes) – сегменти, де дерево розділяється залежно від значення певного параметра;

– ребра (Edges) – результат поділу, що веде до наступного вузла.

Як видно з рис.2.12, на схемі є вузли для прогнозу (outlook), вологості (humidity) і вітру (Windy). І також межі для кожного потенційного значення кожного з цих параметрів.

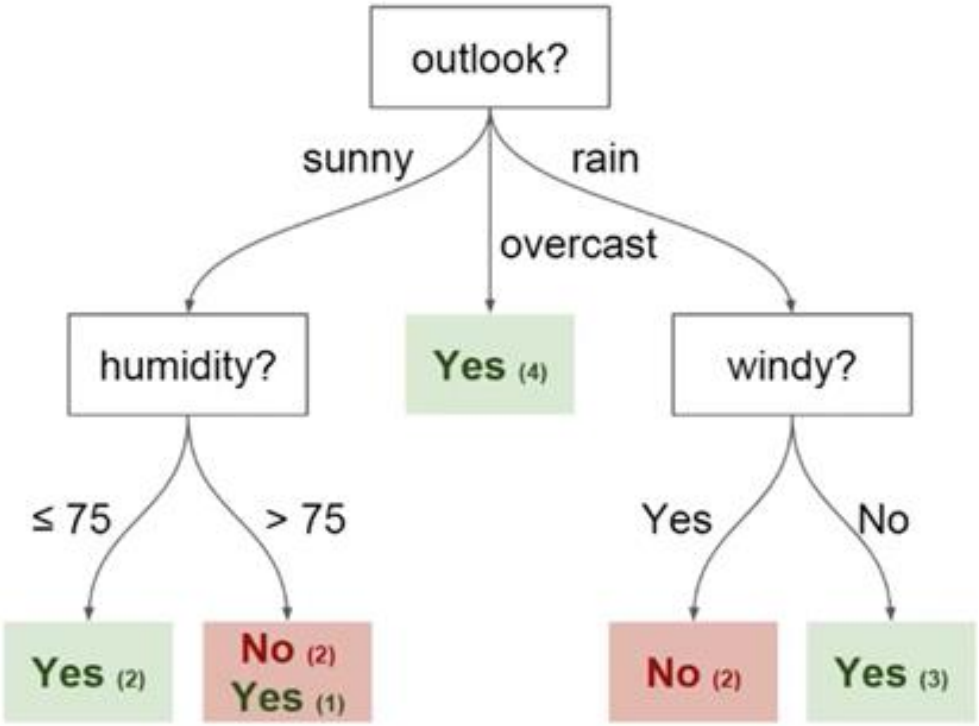


Рисунок 2.12 – Приклад дерева прийняття рішень

При побудові дерев прийняття рішень використовують наступну термінологію:

– корінь (Root) – вузол, з якого починається поділ дерева;

– листкові сегменти (Leaves) – заключні вузли, які прогнозують остаточний результат.

Побудова дерева прийняття рішень складніша, ніж може здатися з першого погляду. Для розв’язання цієї задачі фахівці в області машинного навчання, зазвичай, використовують багато дерев прийняття рішення, застосовуючи випадкові набори характеристик, вибраних для поділу дерева на них. Іншими словами, нові випадкові набори характеристик вибираються для кожного окремого дерева, на кожному окремому поділі. Ця техніка називається випадкові ліси.

У загальному випадку, фахівці зазвичай вибирають розмір випадкового набору характеристик (позначається m) так, щоб він був квадратним коренем загальної кількості характеристик в наборі даних (позначається p).

Якщо коротко, то m – це квадратний корінь з p і тоді конкретна характеристика випадковим чином вибирається з m.

Перевагами використання випадкового лісу є те, що можна опрацьовувати великий набір даних у якого є одна «сильна» характеристика. Іншими словами, в цій множині даних є характеристика, яка набагато більш передбачувана щодо кінцевого результату, ніж інші характеристики цієї множини.

Якщо дерево прийняття рішень будується вручну, то має сенс використовувати цю характеристику для самого «верхнього» поділу у дереві. Це означає, що буде існувати кілька дерев, прогнози яких сильно корелюють. Для хочемо цього уникнути, тому що використання середньої від сильно корелюють змінних не знижує значно дисперсію.

Використовуючи випадкові набори характеристик для кожного дерева у випадковому лісі, виконується декореляція дерева і дисперсія отриманої моделі зменшується. Зменшення кореляцій – головна перевага у використанні випадкових лісів у порівнянні з деревами прийняття рішень.

Перевагами використання підходів дерева прийняття рішень та випадкових лісів є наступні:

– ефективність при дослідженні нелінійних залежностей у наборі даних з доволі високою продуктивністю – краще, ніж у поліноміальної регресії та приблизно однакова з нейромережевим підходом.

– основні алгоритми прості в розумінні і реалізації, межі, які будуються при навчанні моделі також легко зрозуміти.

Недоліки підходу дерев прийняття рішень:

– здатність до перенавчання та складність структури дерев, що зберігає не потрібну інформацію;

– при використанні випадкових лісів для досягнення високої продуктивності використовуються значні апаратні ресурси, зокрема оперативна пам’ять і як наслідок час.

Через специфічну і високодисперсну природу регресії, регресори дерева прийняття рішень потрібно ретельно обрізати. Проте, підхід до регресії нерегулярний – замість того, щоб обчислювати значення безперервно, він приходить до заданим кінцевих вузлів. Якщо регресор обрізаний занадто сильно,

у нього занадто мало кінцевих вузлів, щоб належним чином виконати своє завдання.

Отже, дерево прийняття рішень повинно бути обрізане так, щоб воно мало найбільшу свободу (можливі вихідні значення регресії – кількість кінцевих вузлів), але недостатньо, щоб воно було занадто глибоким. Якщо його не обрізати, то і без того високодисперсний алгоритм стане надмірно складним через природу регресії.

2.2.3 Підхід до прогнозування ціни нерухомості на основі нейронних мереж

Нейронні мережі є досить потужним інструментом у галузі машинного навчання, що може ефективно використовуватись при розв'язанні задач класифікації і регресії. Сигнали проходять через шари нейронів і узагальнюються в один з декількох класів. Однак їх можна дуже швидко адаптувати у регресійні моделі, якщо змінити останню функцію активації. Кожен нейрон передає значення з попереднього зв'язку через функцію активації, що служить для досягнення мети узагальнення і нелінійності. Зазвичай функція активації – це щось на зразок сигмоїдної функції (рис. 2.13) або функції ReLU (рис. 2.14).

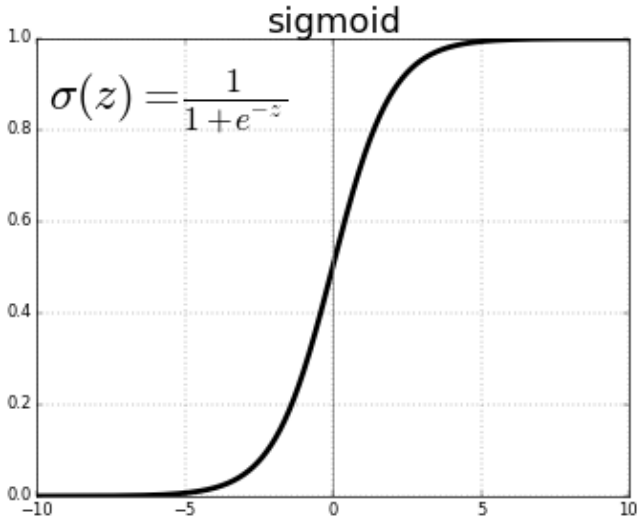


Рисунок 2.13 – Функція активації на основі сигмоїдної функції

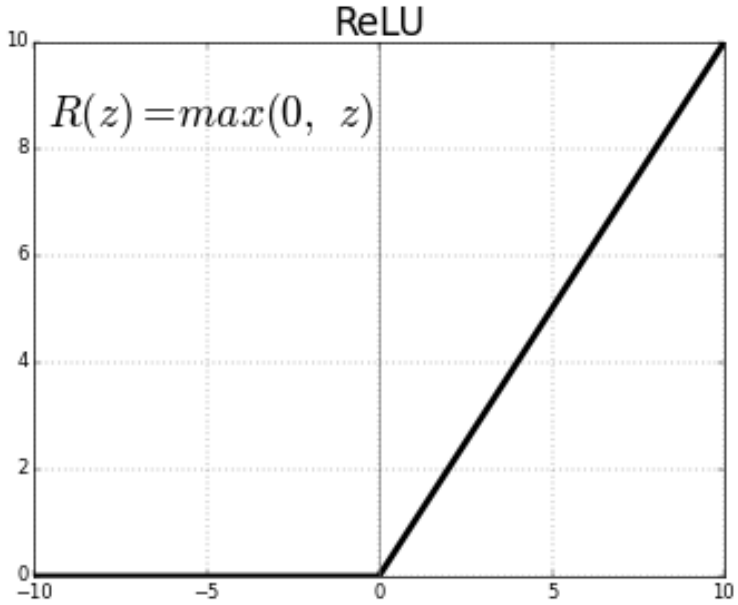


Рисунок 2.14 – Функція активації на основі RELU

Замінивши останню функцію активації (вихідний нейрон) лінійною функцією активації, вихідний сигнал можна відобразити на множину значень, що виходять за межі фіксованих класів. Таким чином, на виході буде не ймовірність приналежності вхідного сигналу до якого-небудь одного класу, а безперервне значення, на якому фіксує свої спостереження нейронна мережа.

У цьому випадку можна сказати, що нейронна мережа як би доповнює лінійну регресію. Нейромережева регресія має перевагу нелінійності, яку можна ввести за допомогою сігмоїду або іншими нелінійними функціями активації раніше у нейронній мережі. Однак надмірне використання ReLU як функції активації може означати, що модель має тенденцію уникати виведення від'ємних значень, оскільки ReLU ігнорує відносні відмінності між негативними значеннями. Це можна вирішити або обмеженням використання ReLU і додаванням більшої кількості негативних значень відповідних функцій активації, або нормалізацією даних до строго позитивного діапазону перед навчанням.

Проблема нейронних мереж завжди полягала в їх високій дисперсії і схильності до перенавчання. Якщо нейронна мережа добре справляється з навчальними даними з чисто лінійною структурою, можливо, краще

використовувати регресію на основі дерев прийняття рішень, яка емулює лінійну і високодисперсних нейронну мережу, але дозволяє краще контролювати глибину, ширину та інші атрибути для контролю перенавчання. Приклад структури нейронної мережі показано на рис. 2.15.

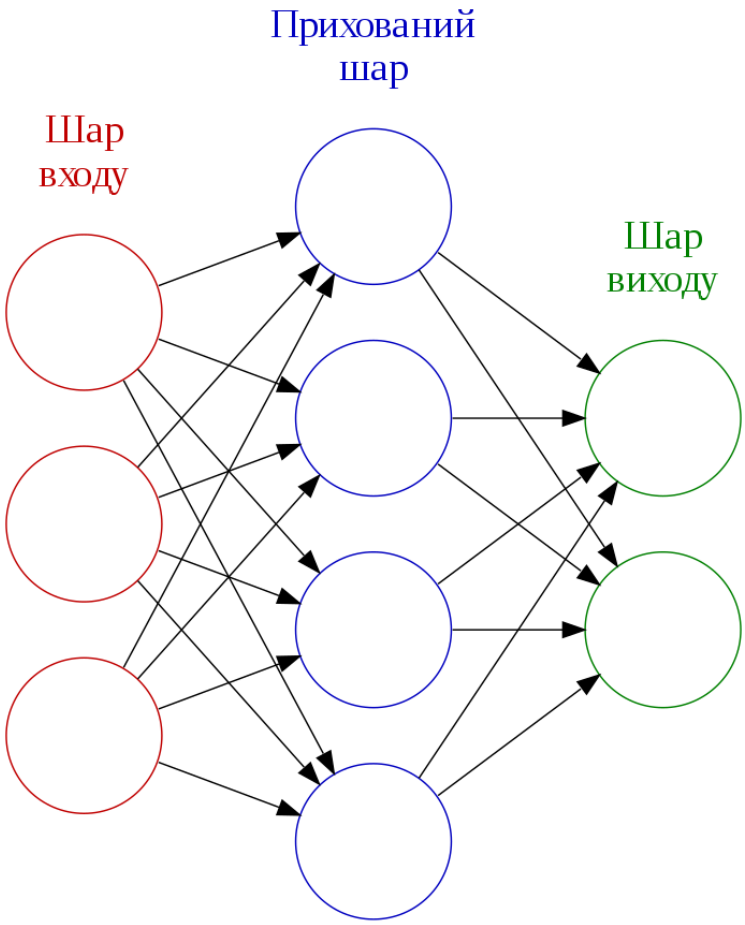


Рисунок 2.15 – Загальний приклад структури нейронної мережі

Нейронна мережа складається із взаємозв'язаних груп вузлів, що називають нейронами. Вхідні дані передаються у ці нейрони у вигляді лінійної комбінації з множиною змінних. Значення кожної змінної помножене на кожну функціональну змінну, називається вагою. Потім до цієї лінійної комбінації застосовується нелінійність, що дає нейронній мережі можливість моделювати складні нелінійні відношення.

Найчастіше нейромережі бувають багатошаровими: вихід одного шару передається наступному. На виході нелінійність не застосовується. Нейронні

мережі тренуються за допомогою методу стохастичного градієнта і алгоритму зворотного поширення помилки.

До переваг застосування нейронних мереж належить:

– нейронні мережі можуть бути багатошаровими, що забезпечує високу ефективність при моделюванні складних нелінійних відношень;

– для нейронних мереж не важлива природа і структура даних, що забезпечує гнучкість при дослідженні майже будь-якого типу змінних ознак об’єкту;

– чим більше навчальних даних для навчання нейронної мережі, тим продуктивніше вона стає.

Через складність моделей нейронних мереж, її складно зрозуміти і реалізувати. Нейронні мережі вимагають ретельного налаштування гіперпараметрів і швидкості навчання. Для досягнення високої продуктивності нейронних мереж необхідна величезна кількість даних, і в результаті, як правило, нейромережі поступаються іншим ML алгоритмам в тих випадках, коли даних мало.

2.3 Процедура побудови інтелектуального модуля прогнозування вартості нерухомості

Формування прогнозів за допомогою підходів машинного навчання передбачає не лише зчитування даних та їх передачу і опрацювання деяким алгоритмом. В свою чергу, алгоритм може видати якийсь результат прогнозу, але він не завжди задовольняє кінцевого користувача.

В загальному випадку, процес побудови інтелектуальних сервісів передбачає виконання наступних кроків:

– розуміння проблеми;

– побудова гіпотези;

– одержання даних;

– дослідження даних;

– препроцесинг даних;

– розробка функцій;
– навчання моделі;
– оцінка моделі;
– тестування моделі.

Перш ніж отримувати дані з деякого джерела інформації, потрібно зрозуміти проблему, яку необхідно вирішити. Якщо фахівець з інтелектуального аналізу даних володіє знаннями про домен чи предметну область необхідно визначити, які фактори можуть відігравати найбільш важливу роль і впливати на результат. В іншому випадку, необхідно залучати експертів або фахівців конкретної предметної області для того, щоб транслювати особливості домену у зрозумілій для розробника формі.

Побудова гіпотез відноситься до класу задач, що передбачає формування набору ознак, які могли б впливати на цільову змінну з урахуванням довірчого інтервалу (весь час приймається як 95%). Таку процедуру можна провести перед тим, як виконується аналіз даних. Це дозволяє уникнути необ'єктивних думок та допомагає у створенні нових функцій.

Одержання даних передбачає завантаження даних із визначеного джерела та їх аналізу. При виконанні цього етапу потрібно визначити, які функції є доступними, а які ні, скільки об'єктів, які створені під час формування гіпотез відображають мету дослідження, а які дані можна додаткового згенерувати.

Дослідження даних допомагає зрозуміти природу змінних (зміщення, відсутність, наявність та величину дисперсії). Ці фактори впливають на правильне опрацювання даних. У результаті виконання цього кроку передбачається створення діаграм, графіків (одновимірний та двовимірний аналіз) та перехресних таблиць для розуміння поведінки ознак.

Під час препроцесингу даних виконується відновлення відсутніх значень та здійснюється їх очищення, зокрема, видалення зайвих пробілів, заміна неправильних розділювачів, представлення єдиного формату часу і т.д. Зазвичай цей крок виконується разом із етапом дослідження даних.

На кроці розробки функцій створюються та імплементуються нові фактори та ознаки об’єктів до існуючого набору даних. Більшість ідей щодо цих особливостей виникають на етапі формування гіпотези.

На етапі навчання моделі виконується пошук і прогнозування значення цільової змінної. Крім цього, визначається які змінні є важливими і найбільше впливають на цільову змінну. У результаті цього можна відібрати найкращі змінні та знову навчити модель.

На етапі оцінювання моделі виконується перевірка моделі на тестовому наборі даних, що відрізняється від навчального дата сету

Після того, як модель навчена, виконується тестування або оцінювання продуктивності моделі із застосуванням відповідних метрик.

3 ПОБУДОВА І ПРОГРАМНА РЕАЛІЗАЦІЯ МОДЕЛІ ПРОГНОЗУВАННЯ ЦІНИ НА НЕРУХОМІСТЬ

3.1 Аналіз вхідного набору даних з характеристиками об’єктів нерухомості

Загальний алгоритм побудови інтелектуальних сервісів на основі інструментів машинного навчання наведено у попередньому розділі. Першим кроком є розуміння проблеми та аналіз вхідних даних при формуванні ціни на нерухомість.

Для формування ціни на нерухомість у даній кваліфікаційній роботі взято набір даних, який описує нерухомість з платформи Kaggle [17]. Даний дата сет описує дані та орієнтований на прогнозування цін на житло в Еймісі, штат Айова, США.

Наступний крок полягає у визначенні гіпотез, що зазвичай складаються з двох частин: нульової гіпотези (H_o) та альтернативної гіпотези (H_1). Їх можна інтерпретувати наступним чином:

– гіпотеза H_o – не існує впливу певної функції на цільову змінну;

– гіпотеза H_1 – існує пряма залежність певної ознаки на цільову змінну.

Виходячи з критерію прийняття рішення (скажімо, рівень значущості 5%), ми завжди `` відхиляємо " або `` не відкидаємо " нульову гіпотезу статистичною мовою.

Практично, будуючи модель, виконується пошук значення ймовірності (p). Якщо значення p <0,05, то відкидається нульова гіпотеза і приймається гіпотеза H_1. Якщо p> 0,05, приймається нульова гіпотеза. Деякі фактори на які повинен звертати увагу розробник, полягають у висуненні гіпотези про те, які безпосередньо з них впливають на ціни на житло. У даному випадку на вартість

нерухомості повинні впливати такі фактори як:

– площа об’єкту нерухомості;
– вік будинку;
– розташування будинку;
– відстань до ринку;
– зручність громадського транспорту;
– поверховість будинку;
– матеріал з якого побудовано об’єкт нерухомості;
– наявність комунікацій і їх види;
– наявність ігрових зон/парків для дітей;
– наявність тераси;
– наявність автостоянки;
– наявність засобів безпеки.

Для кожного конкретного споживача можна формувати значно більше ознак об’єктів нерухомості, ніж наведені вище. Однак перейдемо до аналізу наявного відкритого набору даних про об’єкти нерухомості.

Набір даних містить 81 характеристику (ознаку) об’єктів нерухомості, а цільовою змінною є SalePrice. До складу дата сету входять числові, категоріальні та порядкові змінні. Фрагмент набору даних на основі якого будується інтелектуальний сервіс формування ціни на нерухомість показано на рис. 3.1.

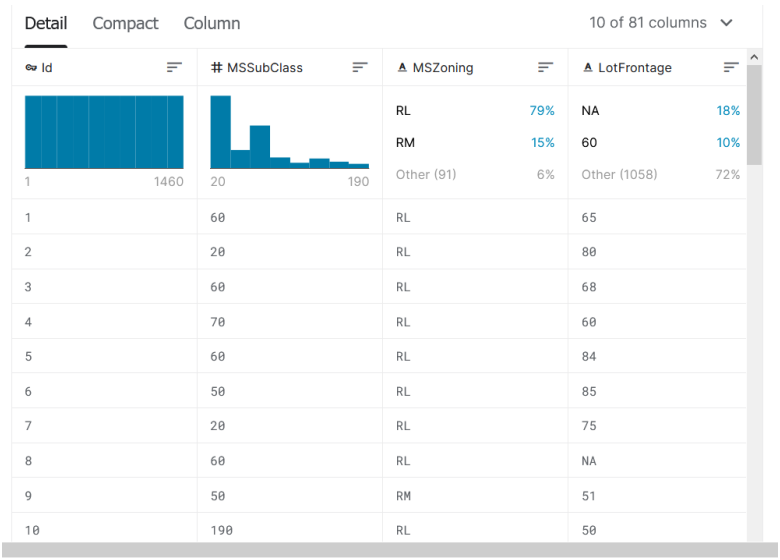


Рисунок 3.1 – Фрагмент вхідного набору даних

Дослідження даних є ключем для одержання статистичних даних. З точки зору практики, хороша стратегія дослідження інформації може вирішити навіть складні проблеми за кілька годин.

Стратегія дослідження даних включає в себе:

– одновимірний аналіз даних –використовується для візуалізації однієї змінної в одному графіку, наприклад, у вигляді гістограми, графіку щільності розподілу тощо;

– двовимірний аналіз – застосовується для візуалізації двох змінних (вісь x та y) на одному графіку (гістограма, лінійна діаграма, діаграма області тощо);

– багатофакторний аналіз – використовується для візуалізації більше двох змінних одночасно (стовпчаста діаграма з накопиченням, гістограма відхилень і т.п.);

– перехресні таблиці – застосовують для порівняння поведінки двох категоріальних змінних (використовуються також у зведених таблицях).

Для побудови моделі формування ціни на нерухомість, виконання препроцесингу, навчання та оцінювання результатів прогнозування у кваліфікаційній роботі пропонується скористатись засобами мови програмування Python та відкритими бібліотеками, оскільки хмарна платформа Azure Machine Learning підтримує саме їх.

Для виконання поставлених задач необхідно завантажити відповідні бібліотеки машинного навчання, як показано у лістингу 3.1.

Лістинг 3.1 – Імпорт потрібних бібліотек

```
#Імпорт бібліотек
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
plt.rcParams['figure.figsize'] = (10.0, 8.0)
import seaborn as sns
from scipy import stats
from scipy.stats import norm
```

Завантаживши бібліотеки з лістингу 3.1, далі потрібно зчитати дані вхідного набору даних, як показано у лістингу 3.2.

Лістинг 3.2 – Зчитування вхідного набору даних

```
#зчитування даних
train = pd.read_csv("/data/Housing/train.csv")
test = pd.read_csv("/data/Housing/test.csv")
```

За допомогою функції head() можна візуалізувати зчитані дані. На рис. 3.2 продемонстровано фрагмент зчитаних даних.

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

Рисунок 3.2 – Результат зчитування вхідного набору даних

Для виявлення пропущених даних (незаповнених комірок дата сету) та їх візуалізації використовується лістинг 3.3.

Лістинг 3.3 – Виявлення та візуалізація пропущених даних

```
train.columns[train.isnull().any()]
#пропущені значення за колонками таблиці
miss = train.isnull().sum()/len(train)
miss = miss[miss > 0]
miss.sort_values(inplace=True)
miss
```

У результаті виконання програмного коду лістингу 3.3, одержано результат, як показано на рис. 3.3.

Electrical	0.000685
MasVnrType	0.005479
MasVnrArea	0.005479
BsmtQual	0.025342
BsmtCond	0.025342
BsmtFinType1	0.025342
BsmtExposure	0.026027
BsmtFinType2	0.026027
GarageCond	0.055479
GarageQual	0.055479
GarageFinish	0.055479
GarageType	0.055479
GarageYrBlt	0.055479
LotFrontage	0.177397
FireplaceQu	0.472603
Fence	0.807534
Alley	0.937671
MiscFeature	0.963014
PoolQC	0.995205
dtype: float64	

Рисунок 3.3 – Характеристики нерухомості, що містять пропущені дані

Аналізуючи результат, показаний на рис. 3.3, можна зробити висновок про те, що у змінної PoolQC (басейн) відсутні 99,5% значень, а близькими до неї є фактори MiscFeature (інші характеристики), Alley та Fence (паркан). Для наочності та більшого розуміння того, з якими даними і факторами потрібно буде працювати в подальшому варто візуалізувати 19 характеристики нерухомості, які містять пропущені дані. Для цього необхідно виконати програмний код, який представлений у лістингу 3.4.

Лістинг 3.4 – Візуалізація пропущених даних

```
#Візуалізація пропущених значень
miss = miss.to_frame()
miss.columns = ['count']
miss.index.names = ['Name']
miss['Name'] = miss.index

#відображення кількості пропущених значень
sns.set(style="whitegrid", color_codes=True)
sns.barplot(x = 'Name', y = 'count', data=miss)
plt.xticks(rotation = 90)
sns.plt.show()
```

На рис. 3.4 представлено результат візуалізації кількості відсутніх значень у вхідному наборі даних.

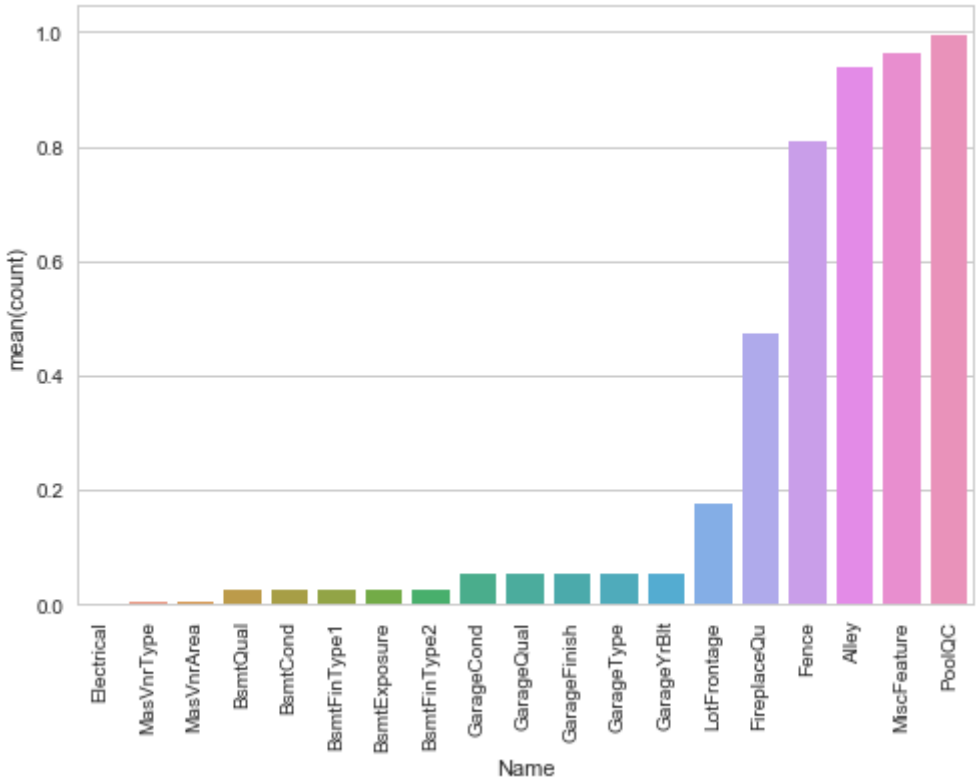


Рисунок 3.4 – Візуалізація кількості пропущених даних у вхідному наборі даних

Далі необхідно визначити тип закону розподілу за цільовою змінною, що інтерпретує вартість нерухомості. Для цього виконується команда: `sns.distplot(train['SalePrice'])`

Результат графічного представлення закону розподілу показано на рис. 3.5.

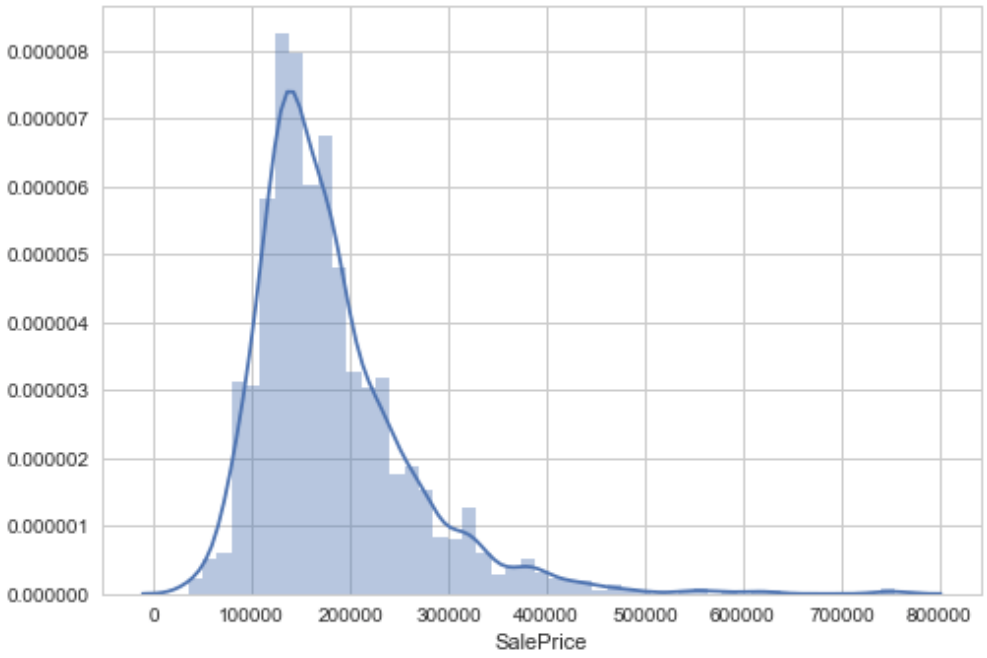


Рисунок 3.5 – Розподіл за цільовою змінною SalePrice

Як видно з рис. 3.5, нормальний розподіл є зміщений вправо. Нормально розподілена (або близька до нормальної) цільова змінна допомагає краще моделювати взаємозв'язок між цільовою та незалежними змінними. Крім того, лінійні алгоритми передбачають постійну дисперсію помилки, тому доцільно привести цю змінну до нормального розподілу без зміщення. Для цього потрібно реалізувати лістинг 3.5.

Лістинг 3.5 – Приведення розподілу до нормального закону

```
#трансформація цільової змінної
target = np.log(train['SalePrice'])
print ('Skewness is', target.skew())
sns.distplot(target)
```

У результаті проведеної трансформації, розподіл набуде вигляду, як показано на рис. 3.6.

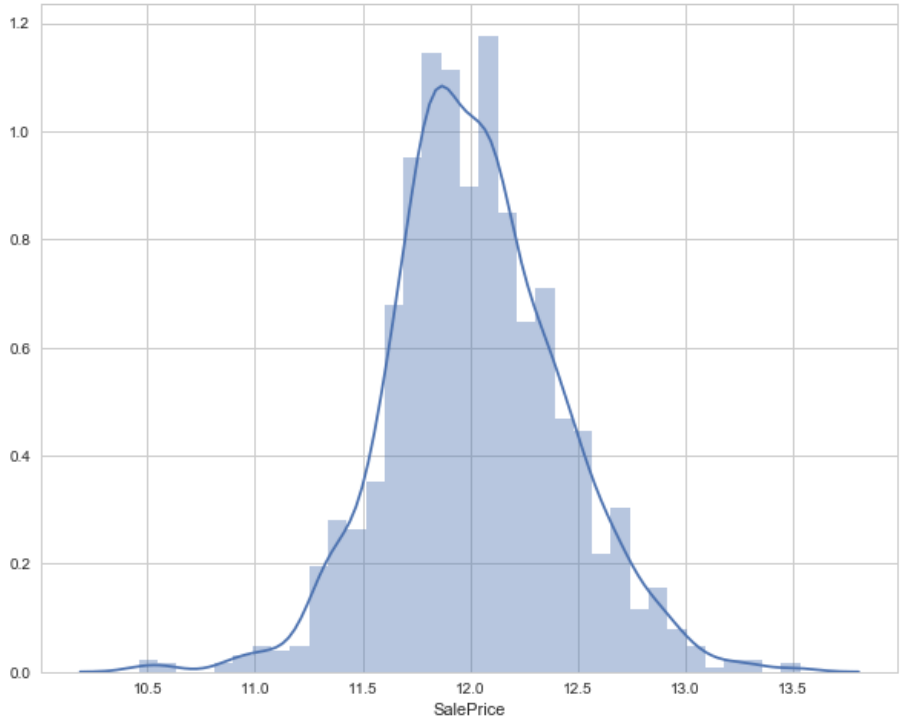


Рисунок 3.6 – Приведена до нормального розподілу змінна SalePrice

Перетворення цільової змінної дозволило виправити її зміщений розподіл, і новий розподіл виглядає ближчим до нормального. Оскільки у наборі даних є 80 змінних, візуалізація кожної з них не буде розумним підходом. Тому доцільним є аналіз деяких змінних на основі їх кореляції з цільовою змінною. Для цього потрібно сформувати на основі існуючого набору даних новий дата сет, з якого варто видалити поле-ідентифікатор (стовбець ID) (лістинг 3.6).

Лістинг 3.6 – Створення нового набору даних

```
#separate variables into new data frames
numeric_data = train.select_dtypes(include=[np.number])
cat_data = train.select_dtypes(exclude=[np.number])
print ("There are {} numeric and {} categorical columns in
train data".format(numeric_data.shape[1],cat_data.shape[1]))`
del numeric_data['Id']
```

Наступний крок полягає у визначенні кореляційної залежності між числовими змінними. З 38 змінних можливим є випадок їх взаємозалежності, які в подальшому доцільно видалити, оскільки вони не надаватимуть корисної інформації для моделі. Пошук корельованих змінних реалізується лістингом 3.7.

Лістинг 3.7 – Побудова кореляційної матриці залежностей

```
corr = numeric_data.corr()
sns.heatmap(corr)
```

Кореляційна матриця залежностей змінних, в тому числі залежності цільової змінної, показана на рис. 3.7.

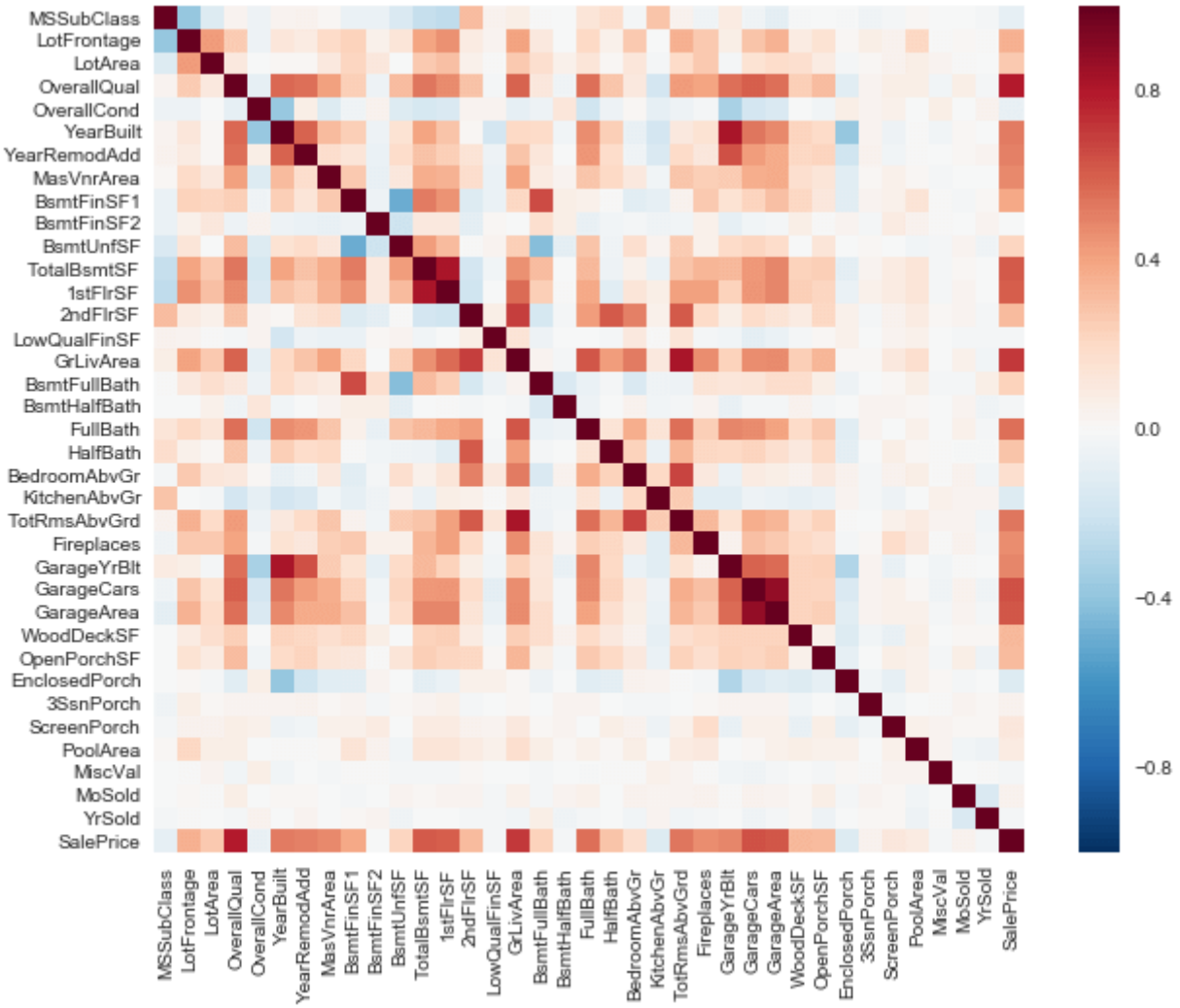


Рисунок 3.7 – Кореляційна матриця

В останньому рядку матриці-карти кореляції (рис. 3.7) можна побачити вплив всіх змінних на цільову (SalePrice). Деякі змінні дуже сильно корелюють з вартістю нерухомістю, тому для більш адекватної оцінки краще їх представити у числовому вигляді, як показано на рис. 3.8.

```
SalePrice        1.000000
OverallQual      0.790982
GrLivArea        0.708624
GarageCars       0.640409
GarageArea       0.623431
TotalBsmtSF      0.613581
1stFlrSF         0.605852
FullBath         0.560664
TotRmsAbvGrd     0.533723
YearBuilt        0.522897
YearRemodAdd     0.507101
GarageYrBlt      0.486362
MasVnrArea       0.477493
Fireplaces       0.466929
BsmtFinSF1       0.386420
Name: SalePrice, dtype: float64, '\n')
----------------------
YrSold           -0.028923
OverallCond      -0.077856
MSSubClass       -0.084284
EnclosedPorch    -0.128578
KitchenAbvGr     -0.135907
Name: SalePrice, dtype: float64</pre>
```

Рисунок 3. 8 – Числове представлення кореляції змінних

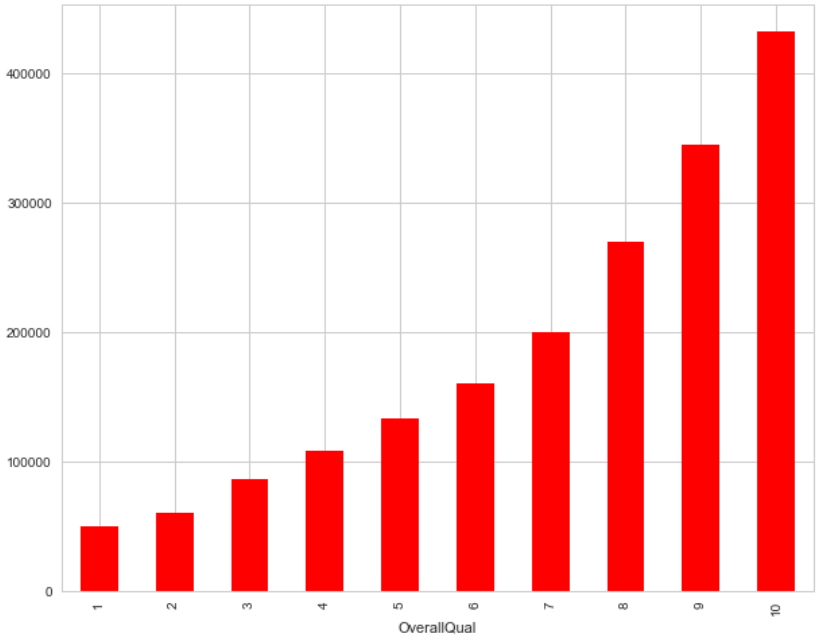
З результатів кореляційного аналізу видно, що змінна, яка інтерпретує загальну якістю об’єкту нерухомості (OverallQual) на 79% впливає на вартість. Дана характеристика стосується типу матеріалу та його якості. Це є справедливим, оскільки люди зазвичай враховують такі параметри при купівлі будинку своєї мрії. Крім того, GrLivArea на 70% корелює з цільовою змінною. GrLivArea відноситься до житлової площі (у квадратних футах). Наступні змінні показують, що люди також дбають про те, чи є в будинку гараж, площа цього гаража, розмір підвалу тощо. Проведемо більш детальний аналіз змінних, які найбільше впливають на ціну нерухомості. Область визначення змінної OverallQual належить інтервалу від 1 до 10 і важливим при цьому є порядок

значень, тобто застосовується порядкова шкала для оцінювання якості матеріалів.

Перш за все цікаво як залежить середня ціна продажу будинків від загальної якості самого будинку. В якості показника середнього у даному випадку потрібно використовувати медіану, оскільки цільова змінна має зміщення. Асиметрична змінна має відхилення, а медіана стійка до них. Програмний код, наведений у лістингу 3.8 дає змогу одержати значення медіани у числовому і графічному вигляді (рис. 3.9).

Лістинг 3.9 – Програмний код для знаходження медіани

```
pivot = train.pivot_table(index='OverallQual',
values='SalePrice', aggfunc=np.median)
pivot.sort
pivot.plot(kind='bar', color='red')
```



1	50150
2	60000
3	86250
4	108000
5	133000
6	160000
7	200141
8	269750
9	345000
10	432390
Name: SalePrice, dtype: int64	

Рисунок 3.9 – Медіана за властивістю OverallQual

Поведінка щодо зміни вартості житла цілком нормальна. Зі збільшенням загальної якості будинку зростає і ціна його продажу. Далі аналогічним чином візуалізуємо наступну корельовану змінну GrLivArea (рис. 3.10).

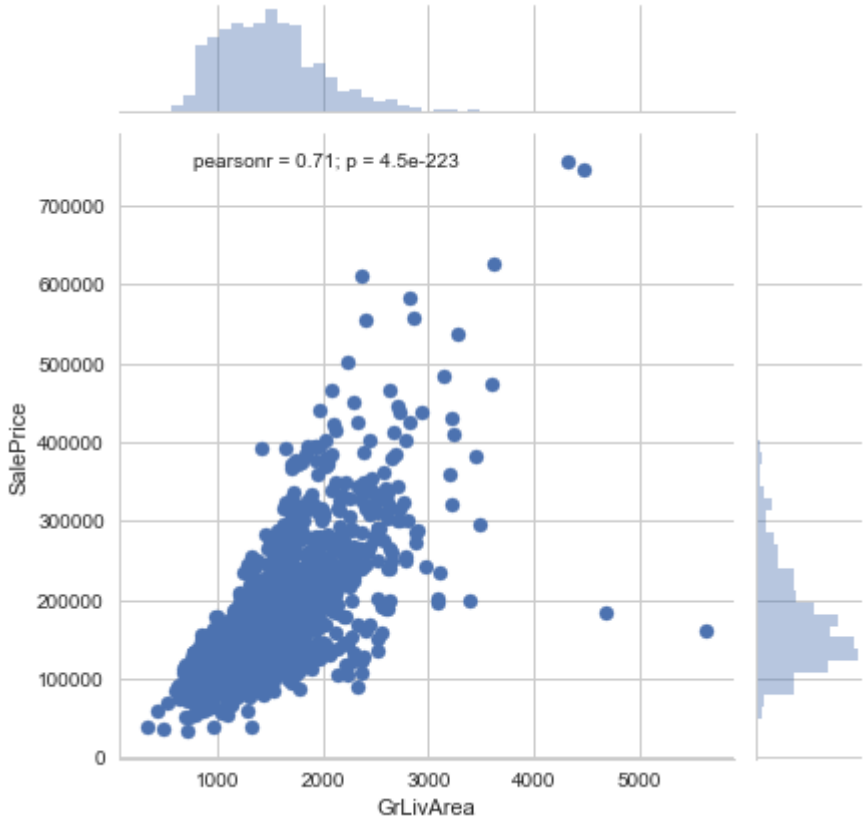


Рисунок 3.10 – Вплив величини житлової площі на вартість нерухомості

Як видно з рис. 3.10, існує пряма кореляція між житловою площею та вартістю нерухомості. Однак, тут можна помітити і величину відхилення GrLivArea> 5000, що є викидом. Викиди погано впливають на продуктивність моделі, тому їх потрібно позбуватись.

Наведені вище кореляції між змінними стосувались лише числових типів даних. Проте у наборі існує ще 41 категоріальна змінна, тому доцільним є також проведення їхнього аналізу. Для прикладу можна перевірити медіану ціни продажу будинку на основі його SaleCondition. SaleCondition пояснює умову продажу, але про інтерпретацію цієї категорії наявно мало інформації. Для визначення медіани та візуалізації даних залежностей між вартістю нерухомості та умовою продажу використовується програмний код, який приведено у лістингу 3.10, а результат кореляції показано на рис. 3.11.

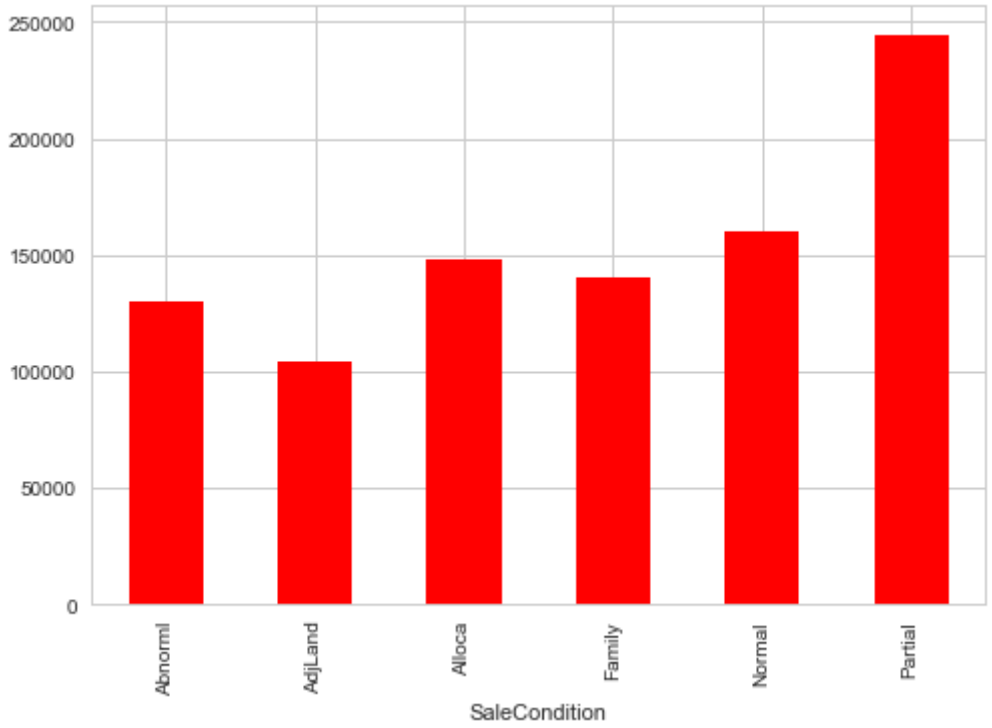


Рисунок 3.11 – Кореляція між SaleCondition та SalePrice

Звідси видно, що SaleCondition Partial має найвищу середню ціну продажу. Хоча через брак інформації не можливо отримати багато інформації з цих даних. Рухаючись вперед, потрібно визначити кореляцію впливу інших ознак на SalePrice. По аналогії до числових змінних використовується тест ANOVA, щоб зрозуміти кореляцію між категоріальними змінними та SalePrice. Тест ANOVA – це статистичний прийом, який застосовують для визначення того, чи існує суттєва різниця середнього значення груп.

Наприклад, є дві змінні A і B. Кожна з цих змінних має 3 рівні (a1, a2, a3 і b1, b2, b3). Якщо середнє значення цих рівнів щодо цільової змінної однакове, тест ANOVA фіксує цю поведінку, і як результат можна безпечно їх видалити. Використовуючи ANOVA, формуються такі гіпотези: X_o – суттєвої різниці між групами не існує та X_1 – між групами існує суттєва різниця. Тепер потрібно визначити функцію, яка обчислює значення p. Із значень p визначається оцінка невідповідності. Чим вище показник невідповідності, тим краща характеристика при прогнозуванні ціни продажу.

Програмний код для виявлення кореляції між категорійними змінними та цільовою змінною наведено у лістингу 3.10.

Лістинг 3.10 – Кореляція між категорійними змінними і цільовою

```
cat = [f for f in train.columns if train.dtypes[f] ==
'object']
def anova(frame):
    anv = pd.DataFrame()
    anv['features'] = cat
    pvals = []
    for c in cat:
            samples = []
            for cls in frame[c].unique():
                    s = frame[frame[c] ==
cls]['SalePrice'].values
                    samples.append(s)
            pval = stats.f_oneway(*samples)[1]
            pvals.append(pval)
    anv['pval'] = pvals
    return anv.sort_values('pval')

cat_data['SalePrice'] = train.SalePrice.values
k = anova(cat_data)
k['disparity'] = np.log(1./k['pval'].values)
sns.barplot(data=k, x = 'features', y='disparity')
plt.xticks(rotation=90)
plt
```

На рис. 3.12 показано кореляцію між категорійними типами змінних і вартістю нерухомості.

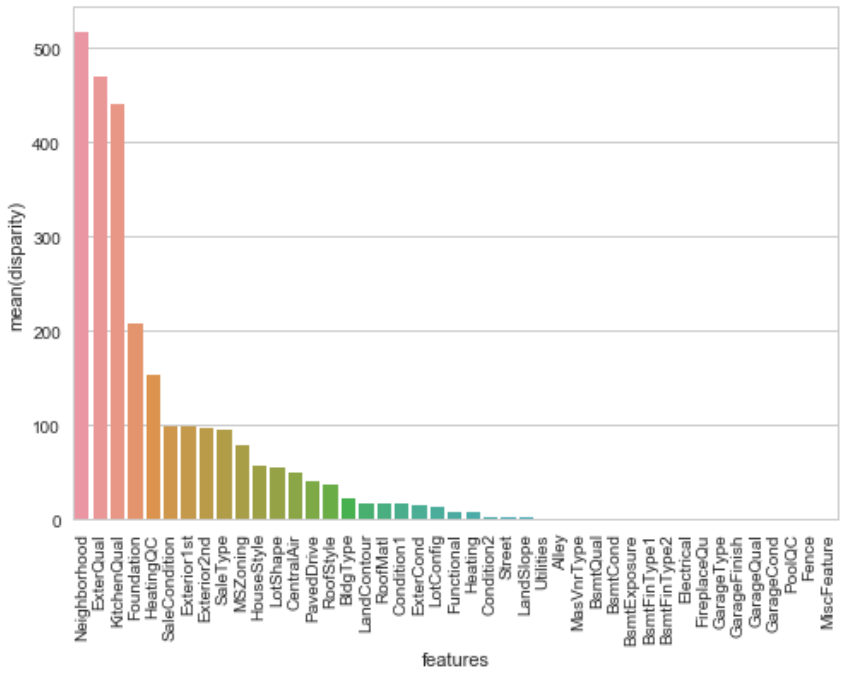


Рисунок 3.12 – Вплив категорійних змінних на цільову

Графічне представлення типів розподілу за числовими та категорійними змінними приведено у додатку Б. Враховуючи той факт, що існують значні відхилення від нормального розподілу за кожною із змінних, доцільним є застосування алгоритмів стійких до викидів, зокрема тих, що базуються на деревах прийняття рішень.

3.2 Препроцесинг даних при побудові рекомендацій ціни на нерухомість

Препроцесинг даних передбачає роботу зі значеннями, які є викидами, кодування змінних, додавання відсутніх значень та вживання усіх можливих ініціатив, які можуть усунути невідповідності у наборі вхідних даних. Видалення викидів зі стовпця GrLivArea та подібних до нього полів дата сету виконується за допомогою лістингу 3.11.

Лістинг 3.11 – Видалення викидів

```
train.drop(train[train['GrLivArea'] > 4000].index,
inplace=True)
train.shape (1456, 81)
```

У рядку 666 в тестовому наборі даних було виявлено, що інформація у змінних, пов’язаних із „Garage” (GarageQual, GarageCond, GarageFinish, GarageYrBlt), відсутня. Для їх заповнення бажано використати середнє значення (моду), як показано у лістингу 3.12.

Лістинг 3.12 – Відновлення даних

```
#imputing using mode
test.loc[666, 'GarageQual'] = "TA"
#stats.mode(test['GarageQual']).mode
test.loc[666, 'GarageCond'] = "TA"
#stats.mode(test['GarageCond']).mode
test.loc[666, 'GarageFinish'] = "Unf"
#stats.mode(test['GarageFinish']).mode
test.loc[666, 'GarageYrBlt'] = "1980"
#np.nanmedian(test['GarageYrBlt'])
```

Тепер потрібно закодувати усі категоріальні змінні (лістинг 3.13). Це необхідно, оскільки більшість алгоритмів машинного навчання не сприймають категоріальних значень, натомість вони будуть перетворені у числові. Функція LabelEncoder від sklearn використовується для кодування змінних.

Лістинг 3.13 – Кодування категоріальних змінних

```
#importing function
from sklearn.preprocessing import LabelEncoder
le = LabelEncoder()
def factorize(data, var, fill_na = None):
      if fill_na is not None:
            data[var].fillna(fill_na, inplace=True)
      le.fit(data[var])
      data[var] = le.transform(data[var])
      return data
```

Функція factorize() передбачає заповнення значення порожніх комірок значеннями моди. Значення моди слід вводити вручну. Тепер, необхідно обчислити відсутні значення у змінній LotFrontage, використовуючи медіанне значення. Такі стратегії відновлення даних будуються під час дослідження даних. Для забезпечення повноти досліджень і одночасної маніпуляції даними у навчальній і тестовій вибірці необхідно їх об’єднати (лістинг 3.14).

Лістинг 3.14 – Об’єднання навчального і тестового набору даних

```
#combine the data set
alldata = train.append(test)
alldata.shape
(2915, 81)
#impute lotfrontage by median of neighborhood
lot_frontage_by_neighborhood =
train['LotFrontage'].groupby(train['Neighborhood'])

for key, group in lot_frontage_by_neighborhood:
                idx = (alldata['Neighborhood'] == key) &
(alldata['LotFrontage'].isnull())
                alldata.loc[idx, 'LotFrontage'] = group.median()
```

Далі, для інших числових змінних заповнимо відсутні значення нулями (лістинг 3.15).

Лістинг 3.15 – Заповнення числових даних нулями

```
#imputing missing values
alldata["MasVnrArea"].fillna(0, inplace=True)
alldata["BsmtFinSF1"].fillna(0, inplace=True)
alldata["BsmtFinSF2"].fillna(0, inplace=True)
alldata["BsmtUnfSF"].fillna(0, inplace=True)
alldata["TotalBsmtSF"].fillna(0, inplace=True)
alldata["GarageArea"].fillna(0, inplace=True)
alldata["BsmtFullBath"].fillna(0, inplace=True)
alldata["BsmtHalfBath"].fillna(0, inplace=True)
alldata["GarageCars"].fillna(0, inplace=True)
alldata["GarageYrBlt"].fillna(0.0, inplace=True)
alldata["PoolArea"].fillna(0, inplace=True)
```

Тепер потрібно перетворити категоріальні змінні на порядкові. Для цього створюється словник пар ключ-значення та пов’язується з його зі змінною у наборі даних.

У результаті проведених маніпуляцій одержано набір даних, який доступний для проведення наступних досліджень, зокрема відбору важливих характеристик, які впливають на ціну нерухомості та побудови моделі прогнозування.

3.3 Інженерія даних при побудові моделі прогнозування ціни нерухомості

Наразі немає бібліотек або наборів ознак, які можна використовувати для проектування факторів впливу на ціну нерухомості. Дана інженерія вимагає знань у галузі машинного навчання. Ідеї щодо нових факторів впливу, як правило, розвиваються на етапах дослідження даних та формування гіпотез. Мотив інженерії факторів, які впливають на цільову змінну, полягає у створенні нових ознак, які можуть допомогти покращити прогнози.

Із заданого набору можна сформувати нові ознаки, які базуються на списку з 81 фактору. Більшість категоріальних змінних мають майже нульовий розподіл дисперсії. Розподіл дисперсії майже до нуля – це коли одна з категорій у змінній має> 90% значень. Створимо бінарні змінні, що відображають наявність або

відсутність категорії. Нові ознаки набуватимуть значень 0 або 1. Крім того, ми створимо ще кілька змінних, які поясненні у коментарях в лістингу 3.16.

Лістинг 3.16 – Додавання нових ознак до набору даних

```
#creating new variable (1 or 0) based on irregular count levels
#The level with highest count is kept as 1 and rest as 0
alldata["IsRegularLotShape"] = (alldata["LotShape"] == "Reg") *
1
alldata["IsLandLevel"] = (alldata["LandContour"] == "Lvl") * 1
alldata["IsLandSlopeGentle"] = (alldata["LandSlope"] == "Gtl") *
1
alldata["IsElectricalSBrkr"] = (alldata["Electrical"] ==
"SBrkr") * 1
alldata["IsGarageDetached"] = (alldata["GarageType"] ==
"Detchd") * 1
alldata["IsPavedDrive"] = (alldata["PavedDrive"] == "Y") * 1
alldata["HasShed"] = (alldata["MiscFeature"] == "Shed") * 1
alldata["Remodeled"] = (alldata["YearRemodAdd"] !=
alldata["YearBuilt"]) * 1

#Did the modeling happen during the sale year?
alldata["RecentRemodel"] = (alldata["YearRemodAdd"] ==
alldata["YrSold"]) * 1

# Was this house sold in the year it was built?
alldata["VeryNewHouse"] = (alldata["YearBuilt"] ==
alldata["YrSold"]) * 1
alldata["Has2ndFloor"] = (alldata["2ndFlrSF"] == 0) * 1
alldata["HasMasVnr"] = (alldata["MasVnrArea"] == 0) * 1
alldata["HasWoodDeck"] = (alldata["WoodDeckSF"] == 0) * 1
alldata["HasOpenPorch"] = (alldata["OpenPorchSF"] == 0) * 1
alldata["HasEnclosedPorch"] = (alldata["EnclosedPorch"] == 0) *
1
alldata["Has3SsnPorch"] = (alldata["3SsnPorch"] == 0) * 1
alldata["HasScreenPorch"] = (alldata["ScreenPorch"] == 0) * 1

#setting levels with high count as 1 and the rest as 0
#you can check for them using the value_counts function
alldata["HighSeason"] = alldata["MoSold"].replace(` `{1: 0, 2:
0, 3: 0, 4: 1, 5: 1, 6: 1, 7: 1, 8: 0, 9: 0, 10: 0, 11: 0, 12:
0})
alldata["NewerDwelling"] = alldata["MSSubClass"].replace(` `{20:
1, 30: 0, 40: 0, 45: 0,50: 0, 60: 1, 70: 0, 75: 0, 80: 0, 85:
0,` `90: 0, 120: 1, 150: 0, 160: 0, 180: 0, 190: 0})
```

Після виконання лістингу 3.16 загальна кількість факторів, які описують об’єкт нерухомості і впливають на його вартість становить 100. Це означає, що додатково створено ще 19 стовпців.

Результат створення та аналізу даних за фактором «Район розташування» показано на рис. 3.13.

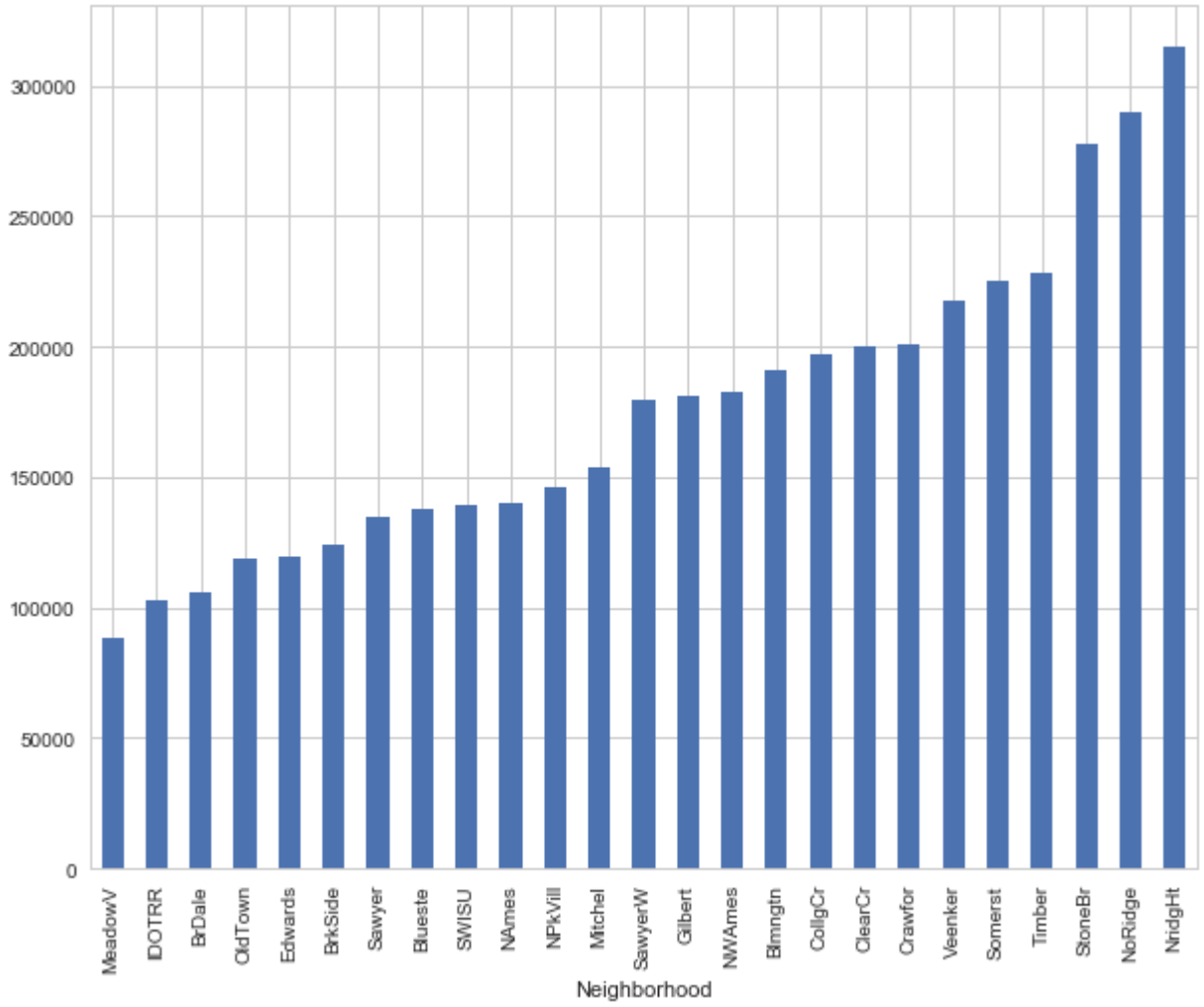


Рисунок 3.13 – Вплив категорії «Розташування району» на ціну нерухомості

Фактори, які належать до однієї категорії та мають приблизно однакове значення варто згрупувати в окремий фактор, що дозволить зменшити розмірність набору даних. Після додавання та об’єднання додаткових ознак нерухомості одержано набір даних, що містить 126 факторів та 2915 записів, з яких 1456 належать навчальній вибірці та 1459 – тестовій.

Далі виконується аналіз та усунення зміщень розподілу за кожною числовою ознакою набору даних та їх стандартизація, а також кодування категоріальних змінних, які наведені у додатку В.

Завершальною фазою роботи з факторами впливу на ціну нерухомості є перетворення цільової змінної та збереження її у новому сформованому масиві даних, як показано у лістингу 3.17.

Лістинг 3.17 – Перетворення цільової змінної

```
#create a label set
label_df = pd.DataFrame(index = train_new.index, columns =
['SalePrice'])
label_df['SalePrice'] = np.log(train['SalePrice'])
print("Training set size:", train_new.shape)
print("Test set size:", test_new.shape)

('Training set size:', (1456, 414))
('Test set size:', (1459, 413))
```

Таким чином проведено так званий «feature engineering», що дало змогу врахувати більше факторів впливу на ціну нерухомості за рахунок нарощування ознак житлових приміщень і перейти до навчання моделей і перевірки результатів прогнозування.

3.4 Навчання і тестування моделі прогнозування вартості нерухомості

Оскільки дані готові до побудови моделі прогнозування ціни на нерухомість, то потрібно реалізувати декілька алгоритмів їх навчання. У роботі пропонується реалізувати 3 алгоритми: XGBoost, підхід нейронних мереж та регресію Лассо. Після цього, ці моделі можна об’єднати в ансамбль для формування остаточних прогнозів. Лістинг 3.17 описує реалізацію алгоритму XGBoost.

Лістинг 3.17 – Реалізація алгоритму XGBoost

```
import xgboost as xgb
regr = xgb.XGBRegressor(colsample_bytree=0.2,
                        gamma=0.0,
                        learning_rate=0.05,
                        max_depth=6,
                        min_child_weight=1.5,
                        n_estimators=7200,
                        reg_alpha=0.9,
                        reg_lambda=0.6,
                        subsample=0.2,
                        seed=42,
                        silent=1)

regr.fit(train_new, label_df)
```

Значення гіперпараметрів можна знайти на основі експериментів із застосуванням методу крос-валідації. Щоб оцінити ефективність моделі, потрібно використати метрику RMSE для оцінювання результатів регресійних задач. Програмна реалізація оцінювання моделі XGBoost наведена у лістингу 3.18.

Лістинг 3.18 – Програмна реалізація оцінювання моделі XGBoost

```
from sklearn.metrics import mean_squared_error
def rmse(y_test,y_pred):
      return np.sqrt(mean_squared_error(y_test,y_pred))

# run prediction on training set to get an idea of how well
it does
y_pred = regr.predict(train_new)
y_test = label_df
print("XGBoost score on training set: ", rmse(y_test,
y_pred))
XGBoost score on training set: ', 0.037633322832013358)

# make prediction on test set
y_pred_xgb = regr.predict(test_new_one)

#submit this prediction and get the score
pred1 = pd.DataFrame({'Id': test['Id'], 'SalePrice':
np.exp(y_pred_xgb)})
pred1.to_csv('xgbnono.csv', header=True, index=False)
```

У результаті виконання лістингу 3.18 одержано значення середньо квадратичного відхилення (RMSE) на рівні 0,12507.

Наступна модель для прогнозування ціни нерухомості регресія на основі моделі Lasso, реалізація якої показана у лістингу 3.19.

Лістинг 3.19 – Модель Lasso

```
from sklearn.linear_model import Lasso

#found this best alpha through cross-validation
best_alpha = 0.00099

regr = Lasso(alpha=best_alpha, max_iter=50000)
regr.fit(train_new, label_df)

# run prediction on the training set to get a rough idea of
how well it does
y_pred = regr.predict(train_new)
y_test = label_df` `print("Lasso score on training set: ",
rmse(y_test, y_pred))
```

Прогнозування на основі моделі для тестового набору даних приведено у лістингу 3.20.

Лістинг 3.20 – Прогнозування ціни на тестовому наборі даних

```
#make prediction on the test set
y_pred_lasso = regr.predict(test_new_one)
lasso_ex = np.exp(y_pred_lasso)
pred1 = pd.DataFrame({'Id': test['Id'], 'SalePrice':
lasso_ex})
pred1.to_csv('lasso_model.csv', header=True, index=False)
```

Значення результату оцінки якості прогнозування на основі метрики RMSE дорівнює 0,11859, тобто похибка менша, ніж при використанні алгоритму XGBoost.

Реалізація моделі прогнозування ціни на основі нейромережевого підходу приведена у лістингу 3.21.

Лістинг 3.21 – Прогнозування ціни нерухомості на основі нейромережевого підходу

```
from keras.models import Sequential
from keras.layers import Dense
from keras.wrappers.scikit_learn import KerasRegressor
from sklearn.preprocessing import StandardScaler

np.random.seed(10)

#create Model
#define base model
def base_model():
     model = Sequential()
     model.add(Dense(20, input_dim=398, init='normal',
activation='relu'))
     model.add(Dense(10, init='normal', activation='relu'))
     model.add(Dense(1, init='normal'))
     model.compile(loss='mean_squared_error', optimizer =
'adam')
     return model

seed = 7
np.random.seed(seed)

scale = StandardScaler()
X_train = scale.fit_transform(train_new)
X_test = scale.fit_transform(test_new)

keras_label = label_df.as_matrix()
clf = KerasRegressor(build_fn=base_model, nb_epoch=1000,
batch_size=5,verbose=0)
clf.fit(X_train,keras_label)

#make predictions and create the submission file
kpred = clf.predict(X_test)
kpred = np.exp(kpred)
pred_df = pd.DataFrame(kpred, index=test["Id"],
columns=["SalePrice"])
pred_df.to_csv('keras1.csv', header=True, index_label='Id')
```

У результаті виконання лістингу 3.21 одержано значення середньоквадратичного відхилення на рівні 1,35346, що є гіршим показником у порівнянні з двома попередніми моделями.

Для реалізації ансамблевого алгоритму із застосування простого голосування реалізовано програмний код, що наведений у лістингу 3.22.

Лістинг 3.22 – Ансамблевий алгоритм формування ціни на нерухомість

```
#simple average
y_pred = (y_pred_xgb + y_pred_lasso) / 2
y_pred = np.exp(y_pred)
pred_df = pd.DataFrame(y_pred, index=test["Id"],
columns=["SalePrice"])
pred_df.to_csv('ensemble1.csv', header=True,
index_label='Id')
```

Похибка середньоквадратичного відхилення при реалізації ансамблевого алгоритму на основі моделей XGBoost та Lasso становить 0,11792, що є найкращим показником на обраному наборі даних. Узагальнюючи результати прогнозування на основі метрики RMSE, одержано результати у вигляді табл. 3.1.

Таблиця 3.1 – Загальні результати прогнозування

№ з/п	Модель	RMSE
1.	XGBoost	0,12507
2.	Lasso	0,11859
3.	Нейронна мережа	1,35346
4.	XGBoost+ Lasso	0,11792

У результаті виконання кваліфікаційної роботи реалізовано комп’ютеризовану систему формування ціни на нерухомість з використанням мови програмування Python та з подальшим розгортанням у хмарній платформі Azure Machine Learning.

4 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ

4.1 Організація служби охорони праці на підприємстві

Роботодавець зобов'язаний згідно Закону України «Про охорону праці» стаття 13 «Управління охороною праці та обов’язки роботодавця» створити на робочому місці в кожному структурному підрозділі умови праці відповідно до нормативно-правових актів, а також забезпечити додержання вимог законодавства щодо прав працівників у галузі охорони праці.

Із цією метою роботодавець забезпечує функціонування системи управління охороною праці, а саме:

– створює відповідні служби і призначає посадових осіб, які забезпечують вирішення конкретних питань охорони праці, затверджує інструкції про їхні обов'язки, права та відповідальність за виконання покладених на них функцій, а також контролює їх додержання;

– розробляє за участю сторін колективного договору і реалізує комплексні заходи для досягнення встановлених нормативів та підвищення існуючого рівня охорони праці;

– забезпечує виконання необхідних профілактичних заходів відповідно до обставин, що змінюються;

– впроваджує прогресивні технології, досягнення науки і техніки, засоби механізації та автоматизації виробництва, вимоги ергономіки, позитивний досвід з охорони праці тощо;

– забезпечує належне утримання будівель та споруд, виробничого обладнання та устаткування, моніторинг за їх технічним станом;

– забезпечує усунення причин, що призводять до нещасних випадків, професійних захворювань, та здійснення профілактичних заходів, визначених

комісіями за підсумками розслідування цих причин;

– організовує проведення аудиту охорони праці, лабораторних досліджень умов праці, оцінку технічного стану виробничого обладнання та устаткування, атестацій робочих місць на відповідність нормативно-правовим актам з охорони праці в порядку і строки, що визначаються законодавством, та за їх підсумками вживає заходів з усунення небезпечних і шкідливих для здоров'я виробничих факторів;

– розробляє і затверджує положення, інструкції, інші акти з охорони праці, що діють у межах підприємства та встановлюють правила виконання робіт і поведінки працівників на території підприємства, у виробничих приміщеннях, на будівельних майданчиках, робочих місцях відповідно до нормативно-правових актів з охорони праці, забезпечує безоплатно працівників нормативно-правовими актами підприємства з охорони праці;

– здійснює контроль за додержанням працівником технологічних процесів, правил поведінки з машинами, механізмами, устаткуванням та іншими засобами виробництва, використанням засобів колективного та індивідуального захисту, виконанням робіт відповідно до вимог з охорони праці;

– організовує пропаганду безпечних методів праці та співробітництво з працівниками у галузі охорони праці.

Роботодавець несе безпосередню відповідальність за порушення нормативно-правових актів з охорони праці. Служба охорони праці створюється роботодавцем на підприємстві з кількістю працівників 50 і більше. На підприємстві з кількістю працівників менше 50 осіб функції цієї служби можуть виконувати у порядку сумісництва особи, що пройшли перевірку знань з охорони праці відповідними державними службами. Якщо кількість працівників менше 20 осіб, для виконання функцій служби охорони праці можуть залучатися сторонні спеціалісти на договірних засадах. Служба охорони праці підпорядковується безпосередньо роботодавцю і прирівнюється до керівників і спеціалістів основних виробничо-технічних служб.

Спеціалісти служби охорони праці у разі виявлення порушень охорони праці мають право:

– видавати керівникам структурних підрозділів підприємства обов'язкові для виконання приписи щодо усунення наявних недоліків, одержувати від них необхідні відомості, документацію і пояснення з питань охорони праці;

– вимагати відсторонення від роботи осіб, які не пройшли передбачених законодавством медичного огляду, навчання, інструктажу, перевірки знань і не мають допуску до відповідних робіт або не виконують вимог нормативно-правових актів з охорони праці;

– зупиняти роботу виробництва, дільниці, машин, механізмів, устаткування та інших засобів виробництва у разі порушень, які створюють загрозу життю або здоров'ю працівників;

– надсилати роботодавцю подання про притягнення до відповідальності працівників, які порушують вимоги щодо охорони праці.

Ліквідація служби охорони праці допускається тільки у разі ліквідації підприємства чи припинення використання найманої праці фізичною особою.

Законодавство про охорону праці передбачає і обов'язки працівників. Зокрема вони зобов'язані:

– дбати про особисту безпеку і здоров'я, а також про безпеку і здоров'я оточуючих людей у процесі виконання будь-яких робіт під час перебування на території підприємства;

– знати і виконувати вимоги нормативно-правових актів з охорони праці, правила поведінки з машинами, механізмами, устаткуванням та іншими засобами виробництва, користуватися засобами колективного та індивідуального захисту;

– проходити у встановленому законодавством порядку попередні та періодичні медичні огляди.

Працівник несе безпосередню відповідальність за порушення зазначених вимог. Дотримання правил безпеки і виробничої санітарії залежить не тільки

від виконання роботодавцем своїх обов'язків, а й від того, наскільки кожен працівник знає і виконує правила під час роботи. Тому всі працівники при прийомі на роботу і в процесі роботи проходять на підприємстві інструктаж з охорони праці, надання першої медичної допомоги потерпілим від нещасних випадків, правил поведінки при виникненні аварій.

Навчання й інструктаж працівників з охорони праці є складовою частиною системи управління охороною праці і проводиться з усіма працівниками в процесі їхньої трудової діяльності. Інструктаж працівників залежно від характеру та часу його проведення буває вступний (при прийомі на роботу); первинний (на робочому місці з усіма працівниками: на роботах із підвищеною небезпекою - один раз на квартал, на інших роботах — один раз на півроку; проводиться або індивідуально, або з групою працівників, що виконують однотипні роботи, за програмою первинного інструктажу); позаплановий (при зміні правил з охорони праці, заміні устаткування чи за інших змін факторів, що впливають на безпеку праці); цільовий (при виконанні разових робіт, не пов'язаних із прямими обов'язками за фахом).

Навчання та інструктаж працівників з охорони праці проводиться у відповідності до Типового положення про порядок проведення навчання і перевірки знань з питань охорони праці від 26.012005 р. № 15 – НПАОП 0.00.4.36 – 05.

4.2 Заходи, які забезпечують створення оптимальних метеорологічних умов у приміщеннях з використанням ПК

Метеорологічні умови визначаються такими параметрами:

– температурою повітря, t (С);
– відносною вологістю, ф (%);
– швидкістю повітря, v ($^{м}/_{с}$).

Крім цих параметрів, що є основними, не слід забувати і про атмосферний тиск (Р, Па), який впливає не тільки на парціальний тиск основних компонентів повітря (кисень та азот), а й на процес дихання.

Життєдіяльність людини проходить в умовах достатньо широкого діапазону тиску 734—1276 гПа. Однак тут треба пам'ятати, що для здоров'я людини є небезпечною швидка зміна тиску, а не сама величина цього тиску. Наприклад, швидке зниження тиску лише на декілька гектопаскалей щодо нормальної величини 1013 гПа спричиняє хворобливі відчуття.

Необхідність урахування основних параметрів метеорологічних умов диктується наслідками в змінах стану людини. Особливо переконливо це можна пояснити під час розглядання теплового балансу між організмом людини і навколишнім середовищем.

Величина тепловиділення (Q) організмом людини залежить від ступеня фізичного напруження у певних метеорологічних умовах і складає від 85 (у стані спокою) до 500 Дж/с (тяжка робота).

Людина постійно перебуває в процесі теплової взаємодії з навколишнім середовищем. Для того, щоб фізіологічні процеси проходили нормально, теплота, що виділяє організм, повинна віддаватись в навколишнє середовище. Співвідношення між кількістю цієї теплоти йохолоджувальною здатністю середовища характеризує умови як комфортні. В умовах комфорту у людини не виникає турбот щодо її температурних відчуттів охолодження чи перегрівання.

Віддача теплоти організмом людини в навколишнє середовище відбувається через теплопровідність крізь одяг (Q_T), конвекцією тіла (Q_K), випромінюванням на навколишні поверхні (Q_B), випаровуванням вологи з поверхні шкіри (Q_{BIffl}). Частина теплоти витрачається на нагрівання повітря, яким дихає людина (Q_r).

Кількість теплоти, яка віддається організмом людини будь-якими шляхами, залежить від того чи іншого параметра мікроклімату. Так, тепловіддача конвекцією залежить від температури навколишнього повітря і швидкості його переміщення. Випромінювання теплоти відбувається у напрямі поверхонь, що оточують людину, мають нижчу температуру поверхні одягу (27—31 °С) і відкритих частин тіла людини (близько 33,4 °С). Під час впливу високих температур навколишньої поверхні (ЗО—35 °С) тепловіддача випромінюванням повністю відсутня, а під час впливу більш високих

температур теплообіг йде у зворотному напрямі — від поверхні до людини. Віддача теплоти за рахунок випаровування залежить від відносної вологості і швидкості переміщення повітря. У стані спокою, коли температура навколишнього середовища 18°С, частка Q_K складає близько ЗО % всієї теплоти, яка віддається людиною, Q_{BIin} = 20 % і Q_n ~ 5 %.

Під час зміни температури повітря, швидкості його руху і вологості, наявності близько людини нагрітої поверхні, в умовах її фізичної праці тощо — це співвідношення змінюється.

Нормальне теплове самопочуття (комфортні умови), відповідно до конкретних видів роботи, забезпечується при дотриманні теплового балансу: $Q = Q_T + Q_K + Q_{Bim} + Q_n$> тому температура внутрішніх органів людини залишається постійною (близько 36,6°С). Ця здатність людського організму до утримання постійної температури під час зміни параметрів мікроклімату та під час виконання роботи будь-якої важкості називається *терморегуляцією.*

Висока температура впливає на людину і сприяє розширенню судин кровообігу. Відповідно має місце підвищений приплив крові до поверхні тіла, і тепловіддача в навколишнє середовище значно підвищується. Однак, коли температура навколишнього середовища і поверхні досягає 30—35°С, віддача теплоти конвекцією і випромінюванням в основному припиняється. Більш висока температура повітря сприяє тому, що більша частина теплоти віддається через випаровування її з поверхні шкіри. В таких умовах організм губить відповідну кількість вологи, а разом з нею і солі, які відіграють важливу роль в життєдіяльності організму.

В умовах зниження температури повітря реакція людського організму на ці зміни інша — судини кровообігу шкіри звужуються, приплив крові до поверхні тіла зменшується, і віддача теплоти конвекцією і випромінюванням зменшується. Таким чином, для теплового самопочуття людини важливим є певне сполучення температури, відносної вологості і швидкості руху повітря.

Вологість повітря значною мірою впливає на терморегулювання організму. Підвищена вологість (ф > 85 %) ускладнює терморегулювання через зниження випару поту, а досить низька

вологість (ф < 20 %) спричиняє сухоту слизових оболонок шляхів дихання. Оптимальні величини відносної вологості складають 40 — 60 %.

Рух повітря в приміщеннях є важливим чинником, який впливає на теплове самопочуття людини. В умовах спекоти рух повітря сприяє підвищенню віддачі теплоти організмом і поліпшує його стан, але в холодну пору року цей вплив не є сприятливим.

Мінімальна швидкість руху повітря, яку відчуває людина, складає 0,2 м/с. Взимку швидкість руху повітря не повинна перевищувати 0,2—0,5 м/с, а влітку 0,2—1,0 м/с.

Швидкість повітря також впливає на розподіл шкідливих речовин у приміщенні. Повітряні потоки можуть розповсюджувати їх по всьому об'єму приміщення, переводити пил з осілого у зважений стан.

Під впливом високої температури повітря, інтенсивного теплового випромінювання виникає загроза перегрівання організму людини, яке характеризується підвищенням температури тіла, рясним потовиділенням, прискореним пульсом і диханням, різкою слабкістю, запамороченням, а в тяжких випадках — появою судом і виникненням теплового удару.

ВИСНОВКИ

У даній роботі спроектовано комп’ютеризовану систему формування ціни на нерухомість із застосуванням хмарних сервісів та реалізовано інтелектуальний модуль, який на основі параметрів і характеристик об’єктів нерухомості дозволяє формувати рекомендації кінцевому користувача.

Для розв’язку поставлених задач у роботі проведено аналіз функціональних особливостей та вартості використання різних хмарних платформ, що підтримують інструменти машинного навчання. У результаті аналізу встановлено, що найбільш ефективним є використання платформи Azure Machine Learning, що володіє власними інструментами розробки інтелектуальних рішень, а також дозволяє інтегрувати зовнішні рішення, реалізовано засобами мови Python.

При розробці комп’ютеризованої системи формування ціни на нерухомість спроектовано її архітектуру, до складу якої входять два основних компоненти:

– веб-сайт або платформа з продажу нерухомості – забезпечує функціональність щодо збору та зберігання даних про об’єкти нерухомості.

– хмарний сервіс – програмний комплекс, що забезпечує функціонування інтелектуальної складової формування ціни на нерухомість.

При проектуванні інтелектуальної складової комп’ютеризованої системи обгрунтовано застосування найбільш ефективних моделей для розв’язку задач регресії, зокрема лінійної регресії та її різновидів, підходу з використанням дерев прийняття рішень і випадкових лісів, а також нейромережевого підходу.

Практична реалізація інтелектуального модуля формування ціни на нерухомість містить аналіз вхідного набору даних, препроцесинг даних, «feature engineering» та реалізацію моделей на основі алгоритму XGBoost, Lasso, нейронної мережі, а також ансамблю XGBoost+Lasso. У результаті експериментальних досліджень досягнуто найкращого результату на основі метрики середньоквадратичного відхилення на рівні 0,11792.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. К. О Нил, Шатт Р. Data Science. Инсайдерская информация для новичков. Включая язык R. Издательский дом "Питер". 2018. 368 с.

2. Фоусет Т., Провост Ф. Data Science для бізнесу: Як збирати, аналізувати і використовувати дані. Наш формат. Київ. 2019. 400 с.

3. Линейная регрессия и методы её восстановления. URL: https://habr.com/ru/post/465743/ (дата звернення 04.05.2021 р.)

4. Алгоритм XGBoost: пусть он царствует долго! URL: https://medium.com/nuances-of-programming/%D0%B0%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC-xgboost-%D0%BF%D1%83%D1%81%D1%82%D1%8C-%D0%BE%D0%BD-%D1%86%D0%B0%D1%80%D1%81%D1%82%D0%B2%D1%83%D0%B5%D1%82-%D0%B4%D0%BE%D0%BB%D0%B3%D0%BE-dc8c4eca3fbc (дата звернення 06.05.2021 р.)

5. Predict the value of your house using Azure Machine Learning. URL: https://channel9.msdn.com/Blogs/Seth-Juarez/Predict-the-value-of-your-house-using-Azure-Machine-Learning (дата звернення 21.04.2021 р.)

6. Building a Regression Model to Predict Real Estate Sales Price. URL: https://gallery.azure.ai/Experiment/Building-a-Regression-Model-to-Predict-Real-Estate-Sales-Price-1 (дата звернення 29.04.2021 р.)

7. 9 ключевых алгоритмов машинного обучения простым языком. URL: https://habr.com/ru/post/509472/ (дата звернення 10.05.2021 р.)

8. Рекомендательные системы. URL: http://www.numberscompany.ru/products/recommenders (дата звернення 20.03.2021 р.)

9. Гомзин А., Коршунов А. Системы рекомендаций: обзор современных подходов. Труды ИСП РАН. 2012. URL: http://cyberleninka.ru/article/n/sistemy-rekomendatsiy-obzorsovremennyh-podhodov (дата звернення 27.03.2021 р).

10. Linden G., Smith B., York J. Amazon.com recommendations: Item-to-item collaborative filtering. IEEE Internet Computing. vol. 7. No. 1. 2003. pp. 76–80.

11. Python-recsys on Github. URL: https://github.com/ocelma/python-recsys (дата звернення 15.04.2021 р).

12. Preprocessing data. URL: https://scikit-learn.org/stable/modules/preprocessing.html#preprocessing (дата звернення 28.04.2021 р.).

13. API reference. URL: https://pandas.pydata.org/docs/reference/index.html (дата звернення 03.05.2021 р.).

14. NumPy Reference. URL: https://numpy.org/doc/stable/reference/index.html (дата звернення 08.05.2021 р.)

15. Барсегян А. Анализ данных и процессов. 3 изд. БХВ-Петербург. 2009. 512 с.

16. Breese J., Heckerman D., Kadie C. Empirical Analysis of Predictive Algorithms for Collaborative Filtering. Proc. 14th Conf. Uncertainty in Artificial Intelligence. 1998. pp. 223-234.

17. Adomavicius G. На пути к новому поколению рекомендационных систем: обзор имеющихся систем и возможные инновации. IEEE Transactions on Knowledge and Data Engineering. Vol. 17. No. 6. 2005. с. 78-86

18. Лексин В.А. Анализ клиентских сред: выявление скрытых профилей и оценивание сходства клиентов и ресурсов. Математические методы распознавания образов-13. М. МАКС Пресс. 2007. С. 488-491

19. Kurucz M., Benczur A., Csalogany K. Methods for large scale SVD with missing values. Proceedings of KDD Cup and Workshop. 2007. pp. 122-129.

20. ДСанПіН 3.3-2.007-98 Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин. - Київ, 1999. - 18с.

21. НПАОП 0.00-1.28-10 «Правила охорони праці під час експлуатації електронно-обчислювальних машин». – Київ, 2010. – 8 с.

Додаток А.

Технічне завдання

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії

Кафедра комп'ютерних систем та мереж

"Затверджую"

Завідувач кафедри КС

_____ Осухівська Г.М.

"___" _____2021 р

КОМП'ЮТЕРИЗОВАНА СИСТЕМА ФОРМУВАННЯ ЦІНИ НА НЕРУХОМІСТЬ З ВИКОРИСТАННЯМ ХМАРНИХ СЕРВІСІВ

ТЕХНІЧНЕ ЗАВДАННЯ

на _*12*_ листках

Вид робіт: Кваліфікаційна робота

На здобуття освітнього ступеня «Бакалавр»

Спеціальність 123 «Комп'ютерна інженерія»

«УЗГОДЖЕНО»	«ВИКОНАВЕЦЬ»
Керівник кваліфікаційної роботи	Студент групи СІс-44
_____ к.т.н., доц. Тиш Є.В.	_____ Горохівський А.В.
«____» _____ 2021 р.	«____» _____ 2021 р.

Тернопіль 2021

1 Загальні відомості

1.1 Повна назва та її умовне позначення

Повна назва теми кваліфікаційної роботи: «Комп’ютеризована система формування ціни на нерухомість з використанням хмарних сервісів».

Умовне позначення кваліфікаційної роботи: КС КРБ 123.164.00.00

1.2 Виконавець

Студент групи СІс-44, факультету комп’ютерно-інформаційних систем і програмної інженерії, кафедри комп’ютерних систем та мереж, Тернопільського національного технічного університету імені Івана Пулюя, Горохівський Анатолій Володимирович.

1.3 Підстава для виконання роботи

Підставою для виконання кваліфікаційної роботи є наказ по університету (№ 4.7-97 від 10.02.2021 р.)

1.4 Планові терміни початку та завершення роботи

Плановий термін початку виконання кваліфікаційної роботи – 10.02.2021 р.

Плановий термін завершення виконання кваліфікаційної роботи – 20.06.2021 р.

1.5 Порядок оформлення та пред’явлення результатів роботи

Порядок оформлення пояснювальної записки та графічного матеріалу здійснюється у відповідності до чинних норм та правил ISO, ЕСКД, ЕСПД та ДСТУ.

Пред’явлення проміжних результатів роботи з виконання кваліфікаційної роботи здійснюється у відповідності до графіку, затвердженого керівником роботи.

Попередній захист кваліфікаційної роботи відбувається при готовності роботи на 90% , наявності пояснювальної записки та графічного матеріалу.

Пред’явлення результатів кваліфікаційної роботи відбувається шляхом захисту на відповідному засіданні ЕК, ілюстрацією основних досягнень за допомогою графічного матеріалу.

2 Призначення і цілі створення системи

2.1 Призначення системи

Комп’ютеризована система формування ціни на нерухомість з використанням хмарних сервісів призначена для прогнозування вартості житла з врахуванням факторів, які впливають на її значення. Для того, щоб спрогнозувати вартість нерухомості доцільно скористатись відкритими даними агенцій нерухомості, які функціонують на ринку в певному регіоні або країні. Система повинна враховувати особливості інфраструктури та місця розташування об’єкту нерухомості, його тип (квартира, котедж або житловий будинок), матеріал з якого виконано стіни, якість ремонту та ряд інших факторів.

Проектована комп’ютеризована система може бути реалізована у вигляді окремого повноцінного сервісу, або як складова підсистема існуючої інформаційної інфраструктури, наприклад, електронного магазину чи сайту агенції нерухомості.

Система повинна реалізовувати функціональність та володіти інтерфейсом для формування запитів щодо параметрів об’єкту нерухомості, а у відповідь на сформований запит генерувати результат щодо можливої його вартості.

До складу комп’ютеризованої системи повинен входити інтелектуальний модуль, що реалізує алгоритми машинного навчання та видає адекватні результати на запити користувачів. Користувачами, які зацікавлені у використанні такої системи є особи, які зацікавлені у купівлі або продажі житлової нерухомості. Оскільки, дані, якими оперують агенції нерухомості доволі громіздкі, то для цього доцільно скористатись хмарними сервісами. Доцільність застосування таких сервісів пов’язана з уникненням необхідності залучення фахівців з налаштування локальної інфраструктури при моделюванні роботи інтелектуальної складової системи, а також сховища для зберігання даних. Окрім цього, наявність готових сервісів машинного навчання в cloud дає змогу використовувати апробовані та найкращі рішення від провідних IT компаній світу.

2.2 Мета створення системи

Метою створення комп’ютеризованої системи формування ціни на нерухомості є автоматизація процесу надання додаткових послуг щодо оцінювання вартості об’єктів житлової нерухомості, виявлення важливих факторів, які впливають на значення ціни, а також, як наслідок, зростання цільової аудиторії відповідних агенцій.

Досягнення поставленої мети кваліфікаційної роботи можливе шляхом розв’язання наступних завдань:

– аналіз особливостей функціонування агенцій нерухомості та існуючих засобів автоматизації щодо підтримки продажу об’єктів нерухомості;

– аналіз параметрів, якими описуються об’єкти житлової нерухомості та визначити найбільш важливі з них, що впливають на вартість;

– обґрунтувати застосування хмарних сервісів для реалізації інтелектуальної підсистеми формування ціни на нерухомості;

– виконати аналіз і препроцесинг даних при описі властивостей нерухомості;

– розробити програмну модель для прогнозування ціни на нерухомість;

– спроектувати архітектуру інтелектуального сервісу прогнозування ціни на нерухомість та визначити можливі шляхи його інтеграції з існуючими системами;

– забезпечити точність прогнозування ціни та стійкість результатів.

2.3 Характеристика об'єкту

2.3.1 Основні задачі та функції об’єкту

Основними функціями комп’ютеризованої системи формування ціни на нерухомість є збір, аналіз і прогнозування вартості завершеного житлового будівництва, що давало б можливість адекватно приймати рішення щодо купівлі або продажу цих об’єктів.

До основних функцій, які виконує система, належать організація взаємодії сховищ даних з інтелектуальними хмарними сервісами для формування вартості нерухомості з врахуванням тенденцій ринку, а також забезпечення прийнятної точності щодо значень ціни. Комп’ютеризована система може бути організована у вигляді сервісу та інтегруватись з іншими суміжними системами.

Для автоматизації процесу інтелектуального формування ціни на нерухомість доцільно використовувати наступні структурні компоненти:

– сховище даних з характеристиками нерухомості, що продається;

– модуль імпорту/експорту даних;

– сервер передачі та опрацювання даних.

При моделюванні і реалізації інтелектуальної складової комп’ютеризованої системи необхідна наявність таких програмних компонентів:

– файл з даними у форматі csv або іншому, що підтримується мовою програмування Python;

– наявність програмних компонентів та бібліотек з підтримкою алгоритмів для розв’язання задач регресії;

– програмні інструменти для формування контейнерів.

Важливим аспектом при проектуванні комп’ютеризованої системи є залучення засобів моделювання таких як UML, зокрема при побудові use case діаграм, діаграм компонентів і класів.

Комп’ютеризована система формування ціни на нерухомість повинна забезпечити ефективність реалізації процесів характерних для агенцій нерухомості, шляхом надання менеджерам і клієнтам додаткових функцій, і як наслідок призвести до зростання продажів об’єктів нерухомості.

3 Вимоги до системи

3.1 Вимоги до системи в цілому

Комп’ютеризована система формування ціни на нерухомість із застосування хмарних сервісів повинна видавати адекватні рекомендації щодо вартості житла із врахуванням доступних факторів, які можна одержати при його метаописі, і тенденцій ринку нерухомості у конкретно взятому регіоні. У загальному випадку, вимоги, які висуваються до проектованої комп’ютеризованої системи можна сформулювати наступним чином:

– можливість зчитування і фільтрації вхідного набору даних щодо пропозицій житла на ринку;

– можливість визначення рівня впливу одних характеристик житла на цільову змінну;

– визначення найбільш важливих характеристик, які впливають на ціну об’єктів нерухомості;

– застосування алгоритмів розв’язання задач регресії з високою точністю;

– видача найбільш релевантних результатів щодо ціни нерухомості;

– продуктивність видачі результатів прогнозування вартості житла;

– можливість налаштування гіперпараметрів моделі при прогнозуванні ціни .

3.1.1 Вимоги до структури та функціонування системи

До структури комп'ютеризованої системи формування ціни на нерухомість висуваються такі основні вимоги:

– наявність компоненти одержання даних з відповідного джерела за заданим URL;

– наявність інтелектуального модуля прогнозування вартості нерухомості на основі хмарних сервісів;

– зручний користувацький відображення рекомендації щодо можливої вартості житлового об'єкта;

– можливість інтеграції з електронними ресурсами і платформами управління процесами продажу нерухомості;

– наявність авторизованого доступу бази даних продажу нерухомості.

В загальному випадку, вимоги до комп'ютеризованої системи формування ціни на нерухомість повинні задовольняти висловленим раніше вимогам, а також відображати процеси щодо прогнозування тенденцій на ринку нерухомості. Дана система повинна бути розгорнута у хмарному сховищі і використовувати модель реалізовану мовою програмування Python або інструменти машинного навчання обраної платформи.

До основних функціональних вимог, які повинна реалізовувати проектована комп'ютеризована система належать:

– здатність імпорту/експорту даних із визначеного джерела даних;

– можливість виявляти пропущені та некоректні дані;

– здатність до відновлення або видалення пропущених даних;

– здатність формувати матрицю кореляцій між параметрами об'єктів нерухомості;

– здатність реалізовувати алгоритми прогнозування вартості нерухомості на основі кращих моделей розв'язання задач регресії;

– можливість візуалізації важливих залежностей та типів розподілів даних;

– здатність кількісного оцінювання якості прогнозованих значень щодо ціни нерухомості;

– можливість інтеграції із суміжними системами по типу електронних магазинів спеціального призначення (продажу об’єктів нерухомості).

3.1.2 Вимоги до способів та засобів зв’язку між компонентами системи

Комунікація та способи зв’язку між структурними компонентами комп’ютеризованої системи формування ціни на нерухомість повинні відповідати вимогам до взаємодії структурних елементів комп’ютерних систем, які функціонують на основі технології клієнт-сервер.

З однієї сторони комп’ютеризована система формування ціни на нерухомість виступає у вигляді клієнта – при зчитуванні даних, а в іншому випадку – серверною частиною – при зверненні зовнішніх систем для формування рекомендацій щодо вартості об’єкту нерухомості.

Усі схеми комунікації між структурними компонентами комп’ютеризованої системи виконуються на основі протоколів комп’ютерних мереж та протоколів HTTP/HTTPS.

3.1.3 Вимоги по діагностуванню системи

Діагностування коректності функціонування комп’ютеризованої системи формування ціни на нерухомість повинна відбуватись у відповідності до графіку профілактичних заходів системи управління продажами об’єктів нерухомості, або при виникненні нештатних ситуацій чи збоїв. Окрім цього, діагностика системи повинна проводитись у випадку зниження точності прогнозування вартості нерухомості або зміни структури вхідного набору даних.

3.1.4 Перспективи розвитку, модернізація системи

Перспективами розвитку і застосування комп’ютеризованої системи формування ціни на нерухомість є можливість гнучкого налаштування

використовуваних апаратних ресурсів при виконанні операцій з прогнозування вартості житлових об’єктів, а також універсальність взаємодії з іншими системами.

Модернізація комп’ютеризованої системи можлива у випадку зміни структури вхідного набору даних, що впливатиме на додаткові фактори ціноутворення, а також при необхідності міграції на інші платформи машинного навчання, які відносяться до класу Low/No code.

Модернізація або утилізація комп’ютеризованої системи виконується у випадку морального старіння технологій та не відповідності стандартам новоутворених технологій і засобів управління бізнес-процесами на ринку нерухомості.

3.1.5 Вимоги до надійності системи

Вимоги, що висуваються до надійності функціонування програмного і апаратного забезпечення комп’ютеризованої системи формування ціни на нерухомість належать:

Безвідмовність роботи протягом часу, визначеного надійністю платформи на якій функціонує хмарний сервіс (зазвичай становить 99,999%) та зовнішньої системи управління бізнес-процесами ринку нерухомості;

Здатність системи відновлювати свою працездатність після виникнення непередбачуваних та нештатних ситуацій

Функції захищеності комп’ютеризованої системи від несанкціонованого втручання повинні виконуватись на інфраструктурному рівні платформ або хостів, де розміщені дані та сама система прогнозування ціни на нерухомість.

3.1.6 Вимоги до функцій та задач, які виконує система

Основними вимогами до функцій, які виконує комп’ютеризована система формування ціни на нерухомість належать :

– здатність виконувати читання файлів у форматі csv та можливості експорту з реляційних баз даних;

– можливість віддаленого управління та фільтрації інформації щодо критеріїв, які формують ціну об’єктів нерухомості;

– можливість керувати гіперпараметрами моделі прогнозування при моделювання роботи комп’ютеризованої системи;

– здатність видавати результати прогнозування ціни на нерухомість;

– можливість забезпечувати точність та адекватність формування ціни з врахуванням тенденцій ринку нерухомості конкретного регіону;

– здатність використовувати метрики якості при оцінюванні результатів прогнозування;

– наявність механізмів захисту на різних рівнях використання системи;

– забезпечення простоти і зручності експлуатації комп’ютеризованої системи;

– здатність до співіснування із визначеними класами систем;

– можливість використання та налаштування параметрів хмарних сервісів.

3.1.7 Вимоги до апаратного забезпечення

Мінімальні вимоги, які висуваються до апаратних характеристик пристроїв для нормального режиму функціонування комп’ютеризованої системи формування ціни на нерухомість:

– тактова частота процесора – 2,0 ГГц з 8-ма паралельними потоками;

– об’єм оперативної пам’яті – 32 ГБ;

– об’єм жорсткого диску – 4Тб.

Вимоги до апаратного забезпечення клієнтських станцій:

– тактова частота процесора – 2,0 ГГц з 4-ма паралельними потоками;

– об’єм оперативної пам’яті – 8 ГБ;

– об’єм жорсткого диску – 1Тб.

3.1.8 Вимоги до програмного забезпечення

Вимоги до програмного забезпечення клієнтських станцій – операційна система бідь-якого типу, наявність браузера.

Вимоги до програмного забезпечення сервера – визначаються програмною екосистемою платформи машинного навчання.

4 Вимоги до документації

Документація повинна відповідати вимогам ЄСКД та ДСТУ

Комплект документації повинен складатись з:

– пояснювальної записки;

– графічного матеріалу:

1 Загальні процеси та структура при побудові сервісів машинного навчання на основі Azure Machine Learning.

2 Інфраструктура Azure Machine Learning.

3 Архітектура комп’ютеризованої системи формування ціни на нерухомість.

4 Підходи для розв’язку регресійних задач.

5 Фрагмент вхідного набору даних і розподіл пропущених даних.

6 Результати прогнозування ціни на нерухомість.

7 *Примітка: У комплект документації можуть вноситися міни та доповнення в процесі розробки.

5 Стадії та етапи проектування

Таблиця 1 – Стадії та етапи виконання кваліфікаційної роботи бакалавра

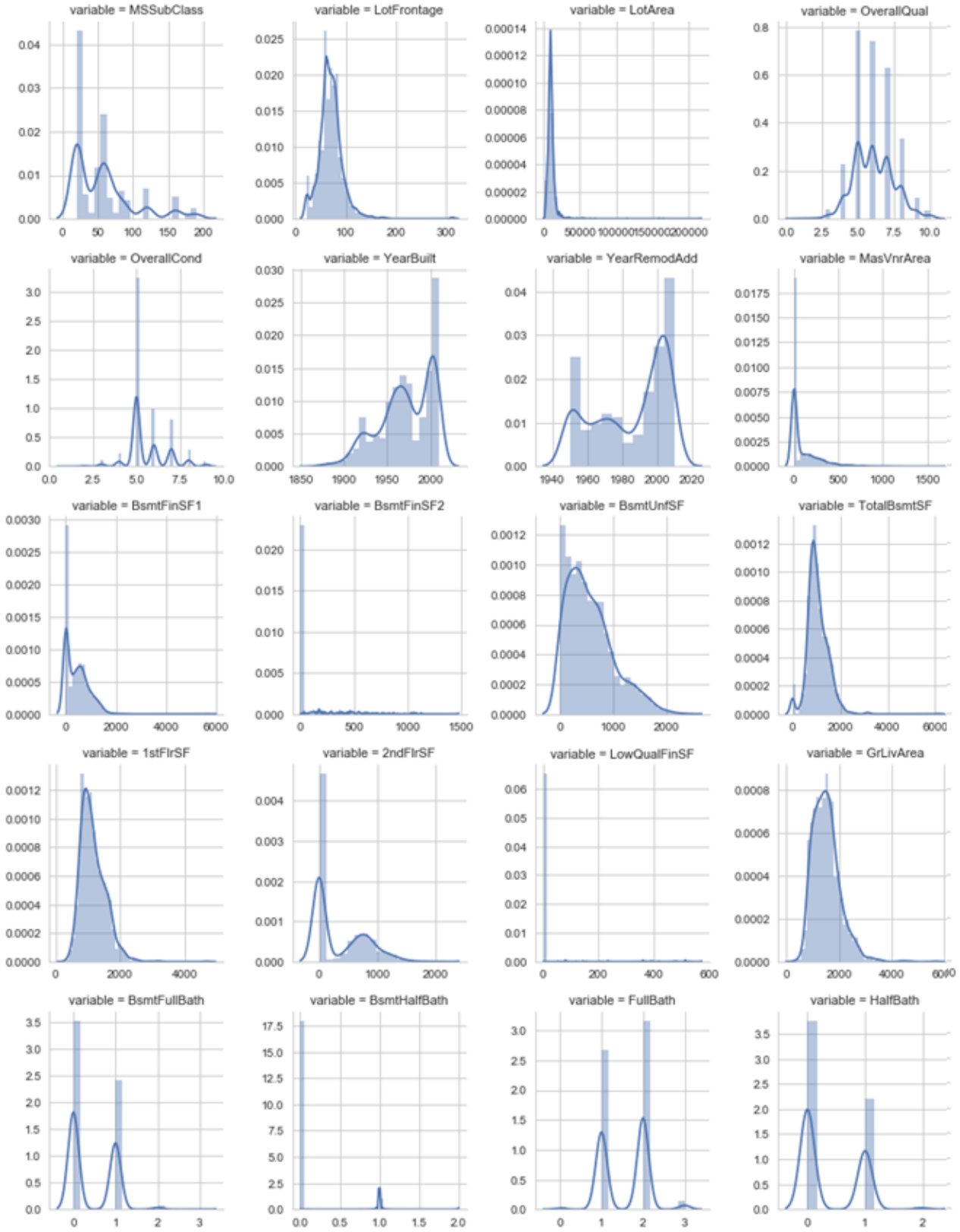
№ етапу	Назва етапу виконання кваліфікаційної роботи	Термін виконання
1	Розробка технічного завдання	10.02-16.02.2021
2	Аналіз технічного завдання	17.02-02.03.2021
3	Аналіз функціональності та вартості хмарних платформ	03.03-18.03.2021
4	Проектування архітектури комп’ютеризованої системи	19.03-04.04.2021
5	Обґрунтування вибору моделі прогнозування ціни на нерухомість	22.03-03.04.2021

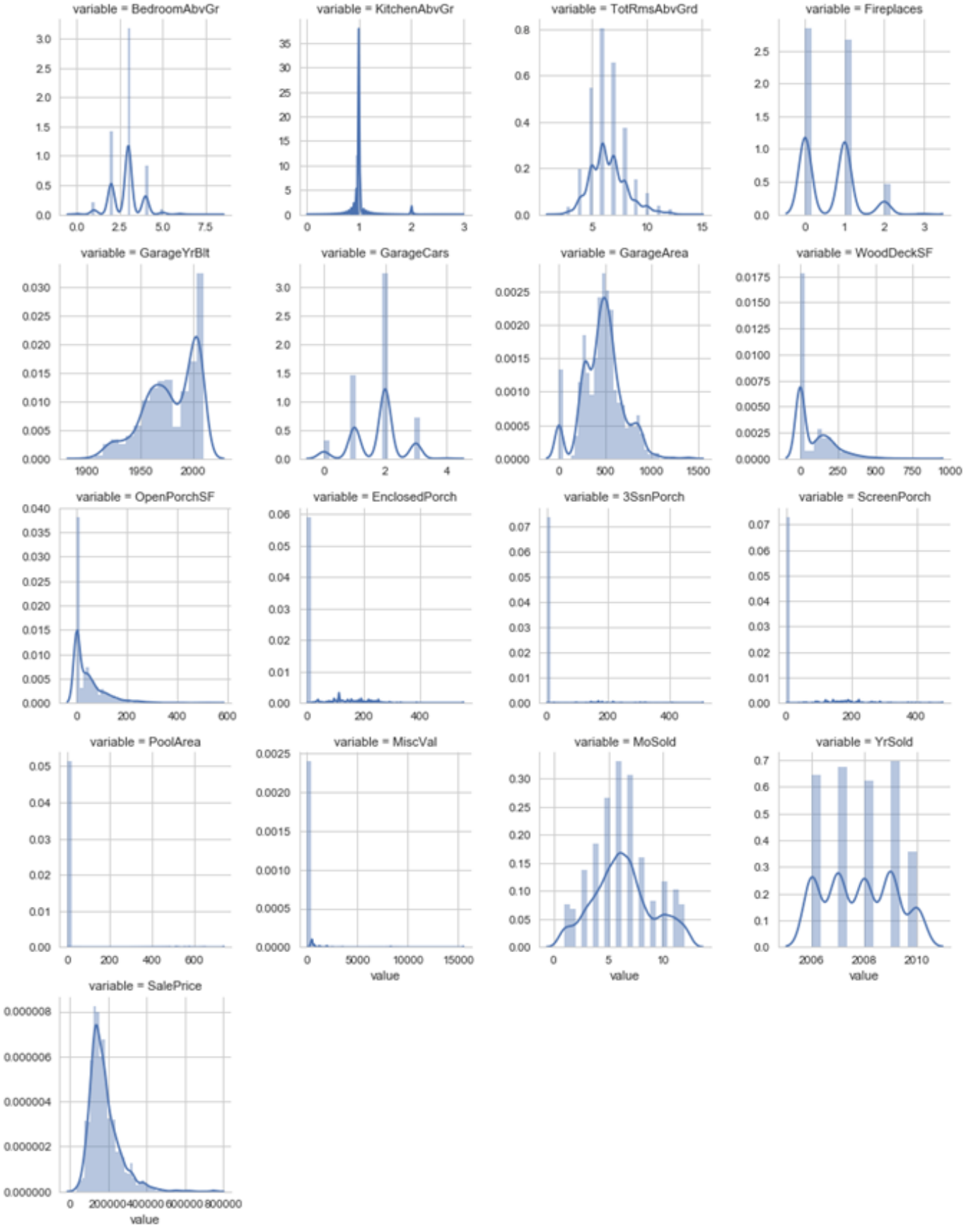
Продовження таблиці 1

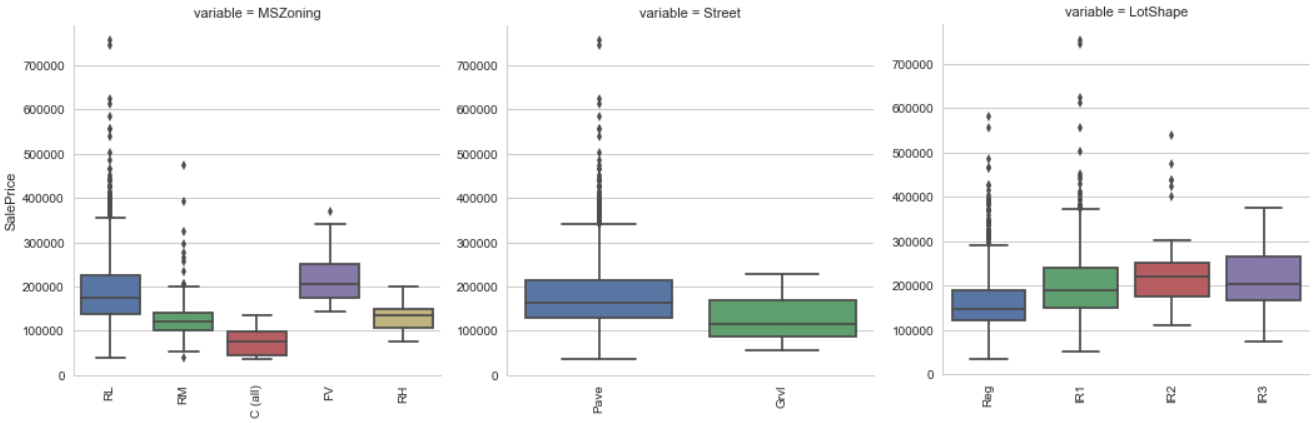
№ етапу	Назва етапу виконання кваліфікаційної роботи	Термін виконання
6	Проектування та реалізація програмної моделі формування ціни на нерухомість	04.04-02.05.2021
7	Розробка інструкцій із встановлення та налаштування параметрів комп’ютеризованої системи	02.05-29.05.2021
8	Безпека життєдіяльності, основи охорони праці	01.06-08.06.2021
9	Оформлення кваліфікаційної роботи	09.06-18.06.2021
10	Попередній захист кваліфікаційної роботи	18.06-22.06.2021
11	Захист кваліфікаційної роботи	22.06-27.06.2021

Додаток Б.

Графіки розподілу за незалежними змінними набору даних

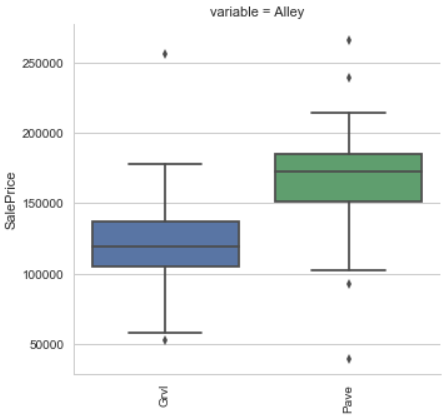


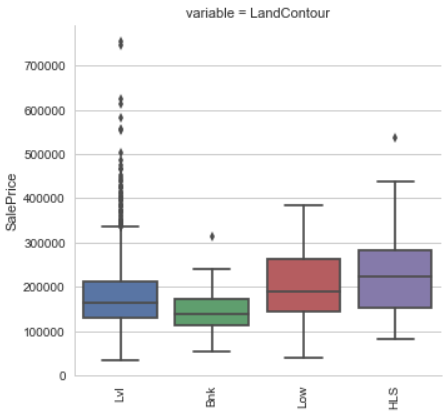

variable = BedroomAbvGr
variable = KitchenAbvGr
variable = TotRmsAbvGrd
variable = Fireplaces
variable = GarageYrBlt
variable = GarageCars
variable = GarageArea
variable = WoodDeckSF
variable = OpenPorchSF
variable = EnclosedPorch
variable = 3SsnPorch
variable = ScreenPorch
variable = PoolArea
variable = MiscVal
variable = MoSold
variable = YrSold
value
variable = SalePrice
value

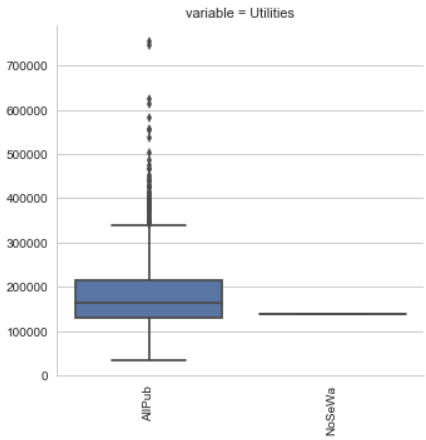

variable = MSZoning
SalePrice
RL
RM
C (all)
FV
RH

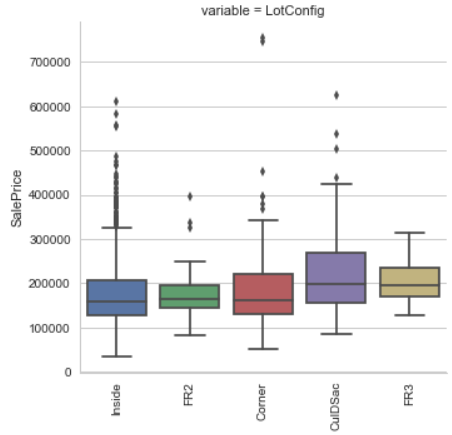
variable = Street
SalePrice
Pave
Grvl

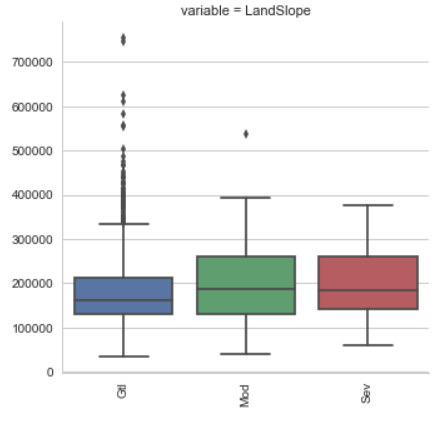
variable = LotShape
Reg
IR1
IR2
IR3

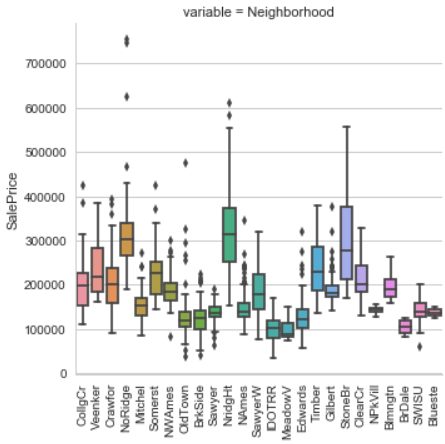

variable = Alley
SalePrice
Grvl
Pave

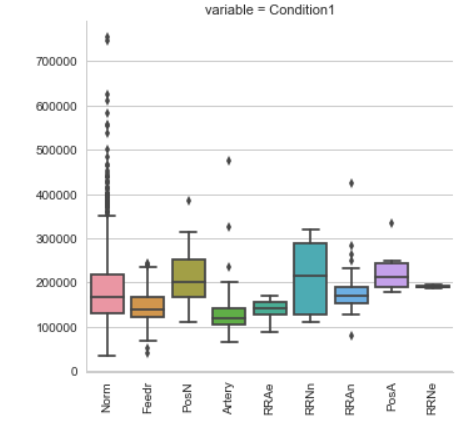

variable = LandContour
SalePrice
Lvl
Bnk
Low
HLS

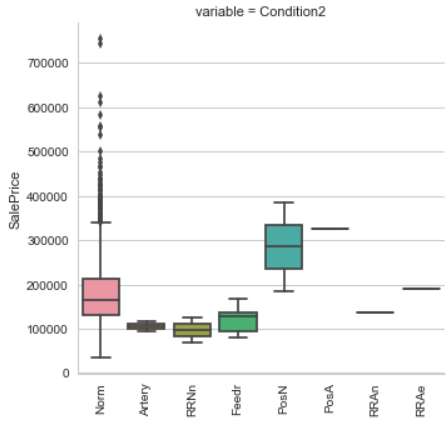

variable = Utilities
AllPub
NoSeWa

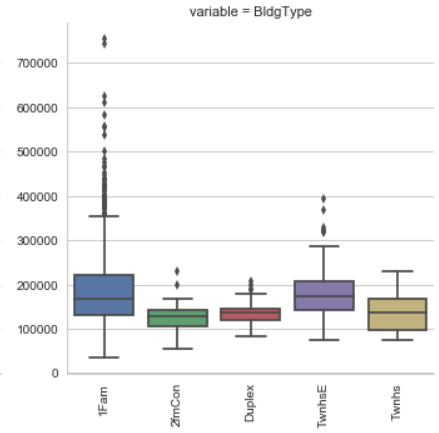

variable = LotConfig
SalePrice
Inside
FR2
Corner
CulDSac
FR3

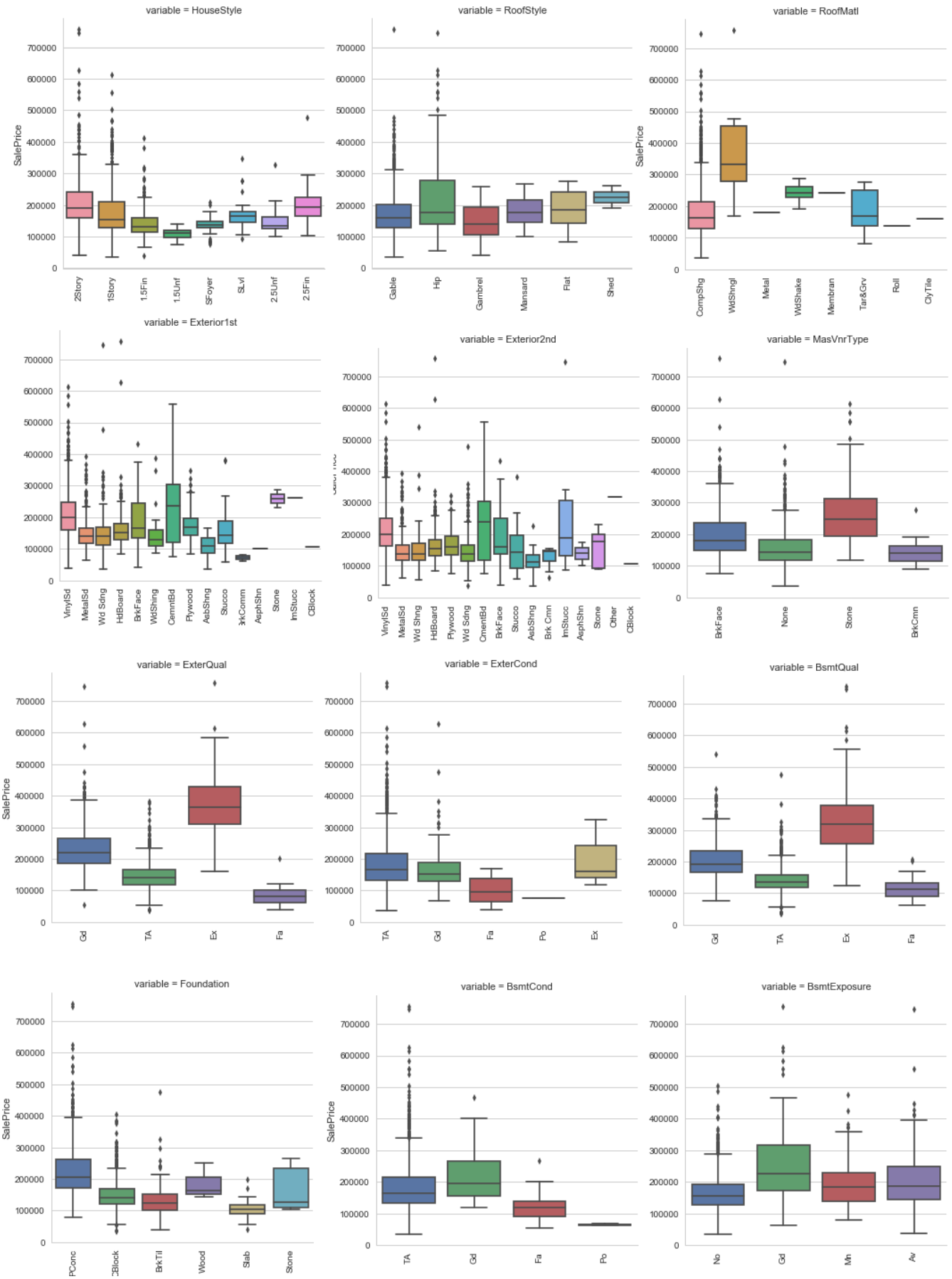

variable = LandSlope
Gtl
Mod
Sev

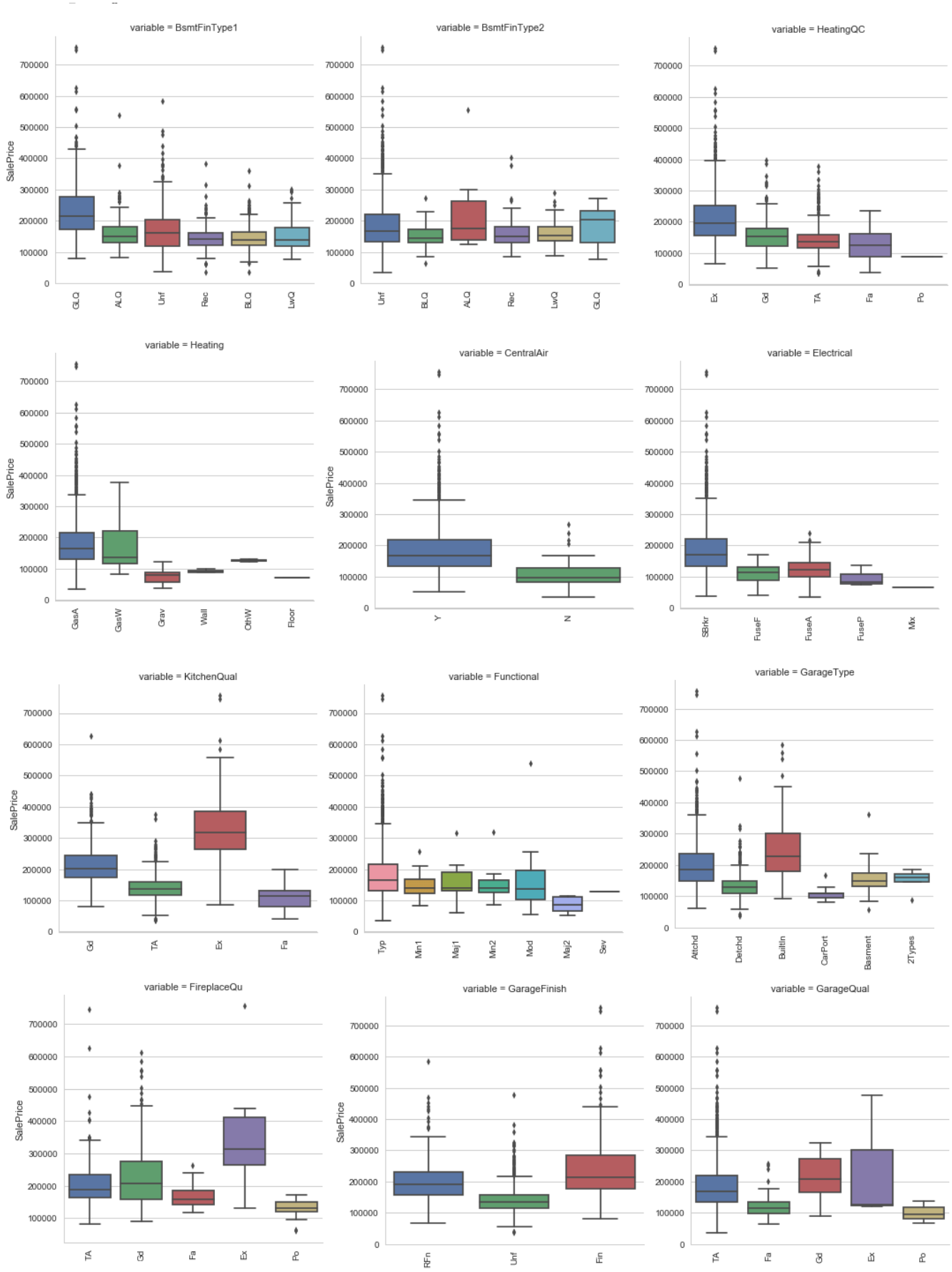

variable = Neighborhood
SalePrice
CollgCr
Veenker
Crawfor
NoRidge
Mitchel
Somerst
NWAmes
OldTown
BrkSide
Sawyer
NridgHt
NAmes
SawyerW
IDOTRR
MeadowV
Edwards
Timber
Gilbert
StoneBr
ClearCr
NPkVill
Blmngtn
BrDale
SWISU
Blueste

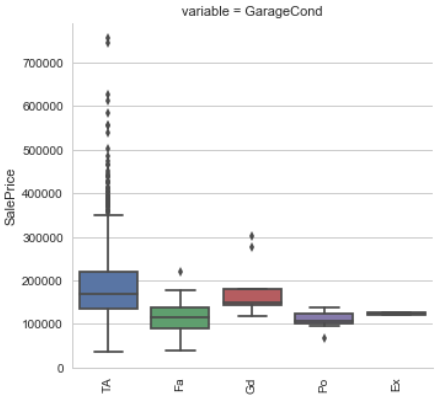

variable = Condition1
Norm
Feedr
PosN
Artery
RRAe
RRNn
RRAn
PosA
RRNe

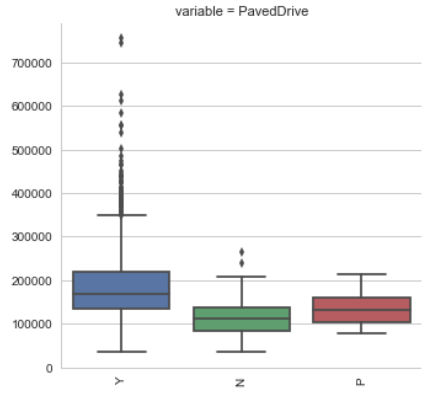

variable = Condition2
SalePrice
Norm
Artery
RRNn
Feedr
PosN
PosA
RRAn
RRAe

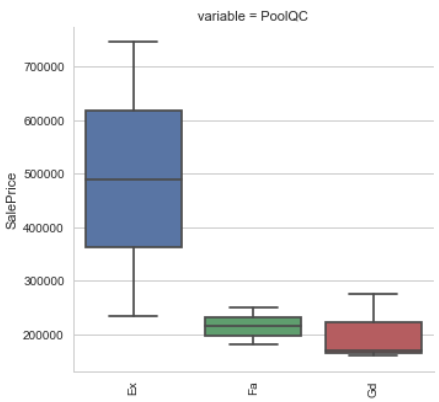

variable = BldgType
1Fam
2fmCon
Duplex
TwnhsE
Twnhs

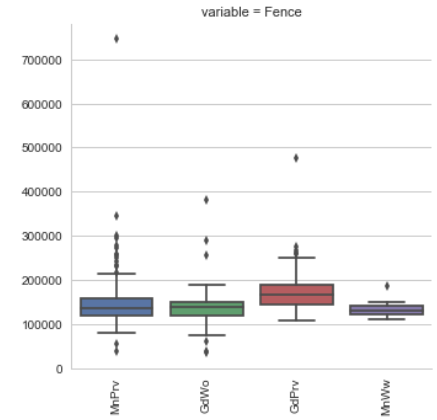

variable = HouseStyle
SalePrice
2Story
1Story
1.5Fin
1.5Unf
SFoyer
SLvl
2.5Unf
2.5Fin
variable = RoofStyle
Gable
Hip
Gambrel
Mansard
Flat
Shed
variable = RoofMatl
CompShg
WdShngl
Metal
WdShake
Membran
Tar&Grv
Roll
ClyTile
variable = Exterior1st
VinylSd
MetalSd
Wd Sdng
HdBoard
BrkFace
WdShing
CemntBd
Plywood
AsbShng
Stucco
Stone
ImStucc
CBlock
variable = Exterior2nd
VinylSd
MetalSd
Wd Shng
HdBoard
Plywood
Wd Sdng
CmentBd
BrkFace
Stucco
AsbShng
Brk Cmn
ImStucc
AsphShn
Stone
Other
CBlock
variable = MasVnrType
BrkFace
None
Stone
BrkCmn
variable = ExterQual
SalePrice
Gd
TA
Ex
Fa
variable = ExterCond
TA
Gd
Fa
Po
Ex
variable = BsmtQual
Gd
TA
Ex
Fa
variable = Foundation
SalePrice
PConc
CBlock
BrkTil
Wood
Slab
Stone
variable = BsmtCond
SalePrice
TA
Gd
Fa
Po
variable = BsmtExposure
No
Gd
Mn
Av

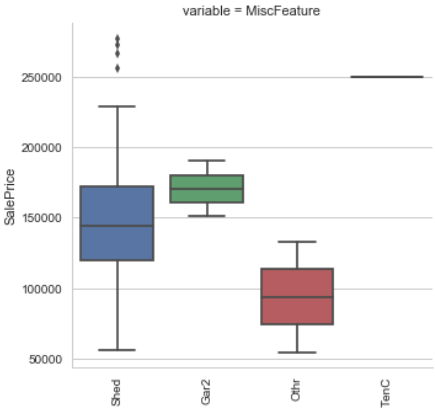

variable = BsmtFinType1
variable = BsmtFinType2
variable = HeatingQC
variable = Heating
variable = CentralAir
variable = Electrical
variable = KitchenQual
variable = Functional
variable = GarageType
variable = FireplaceQu
variable = GarageFinish
variable = GarageQual
SalePrice

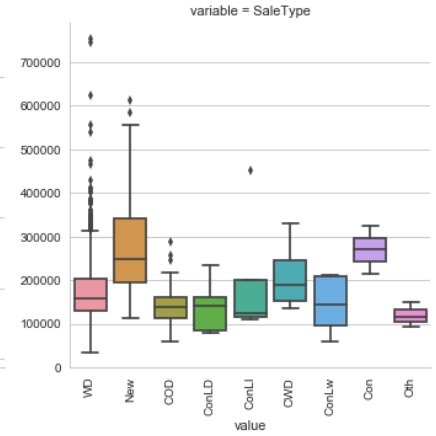

variable = GarageCond
SalePrice
0
100000
200000
300000
400000
500000
600000
700000
TA
Fa
Gd
Po
Ex

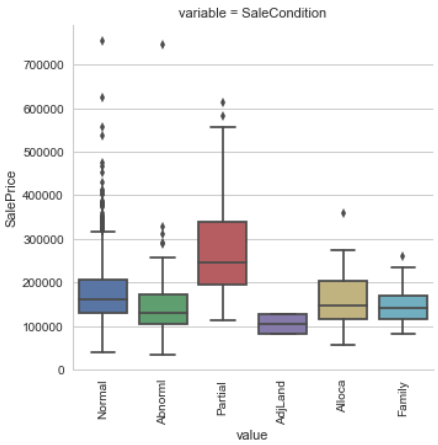

variable = PavedDrive
SalePrice
0
100000
200000
300000
400000
500000
600000
700000
Y
N
P


variable = PoolQC
SalePrice
200000
300000
400000
500000
600000
700000
Ex
Fa
Gd


variable = Fence
0
100000
200000
300000
400000
500000
600000
700000
MnPrv
GdWo
GdPrv
MnWw


variable = MiscFeature
SalePrice
50000
100000
150000
200000
250000
Shed
Gar2
Othr
TenC


variable = SaleType
0
100000
200000
300000
400000
500000
600000
700000
WD
New
COD
ConLD
ConLI
CWD
ConLw
Con
Oth
value


variable = SaleCondition
SalePrice
0
100000
200000
300000
400000
500000
600000
700000
Normal
Abnorml
Partial
AdjLand
Alloca
Family
value

Додаток В.

Кодування категоріальних змінних

```
def onehot(onehot_df, df, column_name, fill_na):
       onehot_df[column_name] = df[column_name]
       if fill_na is not None:
            onehot_df[column_name].fillna(fill_na, inplace=True)

       dummies = pd.get_dummies(onehot_df[column_name],
prefix="_"+column_name)
       onehot_df = onehot_df.join(dummies)
       onehot_df = onehot_df.drop([column_name], axis=1)
       return onehot_df

def munge_onehot(df):
       onehot_df = pd.DataFrame(index = df.index)

       onehot_df = onehot(onehot_df, df, "MSSubClass", None)
       onehot_df = onehot(onehot_df, df, "MSZoning", "RL")
       onehot_df = onehot(onehot_df, df, "LotConfig", None)
       onehot_df = onehot(onehot_df, df, "Neighborhood", None)
       onehot_df = onehot(onehot_df, df, "Condition1", None)
       onehot_df = onehot(onehot_df, df, "BldgType", None)
       onehot_df = onehot(onehot_df, df, "HouseStyle", None)
       onehot_df = onehot(onehot_df, df, "RoofStyle", None)
       onehot_df = onehot(onehot_df, df, "Exterior1st", "VinylSd")
       onehot_df = onehot(onehot_df, df, "Exterior2nd", "VinylSd")
       onehot_df = onehot(onehot_df, df, "Foundation", None)
       onehot_df = onehot(onehot_df, df, "SaleType", "WD")
       onehot_df = onehot(onehot_df, df, "SaleCondition", "Normal")

       #Fill in missing MasVnrType for rows that do have a MasVnrArea.
       temp_df = df[["MasVnrType", "MasVnrArea"]].copy()
       idx = (df["MasVnrArea"] != 0) & ((df["MasVnrType"] == "None") |
(df["MasVnrType"].isnull()))
       temp_df.loc[idx, "MasVnrType"] = "BrkFace"
       onehot_df = onehot(onehot_df, temp_df, "MasVnrType", "None")

       onehot_df = onehot(onehot_df, df, "LotShape", None)
       onehot_df = onehot(onehot_df, df, "LandContour", None)
       onehot_df = onehot(onehot_df, df, "LandSlope", None)
       onehot_df = onehot(onehot_df, df, "Electrical", "SBrkr")
       onehot_df = onehot(onehot_df, df, "GarageType", "None")
       onehot_df = onehot(onehot_df, df, "PavedDrive", None)
       onehot_df = onehot(onehot_df, df, "MiscFeature", "None")
       onehot_df = onehot(onehot_df, df, "Street", None)
       onehot_df = onehot(onehot_df, df, "Alley", "None")
       onehot_df = onehot(onehot_df, df, "Condition2", None)
       onehot_df = onehot(onehot_df, df, "RoofMatl", None)
       onehot_df = onehot(onehot_df, df, "Heating", None)
```

```
        # we'll have these as numerical variables too
        onehot_df = onehot(onehot_df, df, "ExterQual", "None")
        onehot_df = onehot(onehot_df, df, "ExterCond", "None")
        onehot_df = onehot(onehot_df, df, "BsmtQual", "None")
        onehot_df = onehot(onehot_df, df, "BsmtCond", "None")
        onehot_df = onehot(onehot_df, df, "HeatingQC", "None")
        onehot_df = onehot(onehot_df, df, "KitchenQual", "TA")
        onehot_df = onehot(onehot_df, df, "FireplaceQu", "None")
        onehot_df = onehot(onehot_df, df, "GarageQual", "None")
        onehot_df = onehot(onehot_df, df, "GarageCond", "None")
        onehot_df = onehot(onehot_df, df, "PoolQC", "None")
        onehot_df = onehot(onehot_df, df, "BsmtExposure", "None")
        onehot_df = onehot(onehot_df, df, "BsmtFinType1", "None")
        onehot_df = onehot(onehot_df, df, "BsmtFinType2", "None")
        onehot_df = onehot(onehot_df, df, "Functional", "Typ")
        onehot_df = onehot(onehot_df, df, "GarageFinish", "None")
        onehot_df = onehot(onehot_df, df, "Fence", "None")
        onehot_df = onehot(onehot_df, df, "MoSold", None)

        # Divide  the years between 1871 and 2010 into slices of 20
years
        year_map = pd.concat(pd.Series("YearBin" + str(i+1),
index=range(1871+i*20,1891+i*20))  for i in range(0, 7))
        yearbin_df = pd.DataFrame(index = df.index)
        yearbin_df["GarageYrBltBin"] = df.GarageYrBlt.map(year_map)
        yearbin_df["GarageYrBltBin"].fillna("NoGarage", inplace=True)
        yearbin_df["YearBuiltBin"] = df.YearBuilt.map(year_map)
        yearbin_df["YearRemodAddBin"] = df.YearRemodAdd.map(year_map)

        onehot_df = onehot(onehot_df, yearbin_df, "GarageYrBltBin",
None)
        onehot_df = onehot(onehot_df, yearbin_df, "YearBuiltBin", None)
        onehot_df = onehot(onehot_df, yearbin_df, "YearRemodAddBin",
None)
        return onehot_df

#create one-hot features
onehot_df = munge_onehot(train)

neighborhood_train = pd.DataFrame(index=train_new.shape)
neighborhood_train['NeighborhoodBin'] = train_new['NeighborhoodBin']
neighborhood_test = pd.DataFrame(index=test_new.shape)
neighborhood_test['NeighborhoodBin'] = test_new['NeighborhoodBin']

onehot_df = onehot(onehot_df, neighborhood_train, 'NeighborhoodBin',
None)
```