Міністерство освіти і науки України
Київський національний університет імені Тараса Шевченка

Факультет інформаційних технологій
Кафедра мережевих та інтернет технологій

ЗАТВЕРДЖУЮ
завідувач кафедри
мережевих та інтернет технологій
_ Юрій КРАВЧЕНКО
«_____»_____2022 року

КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

галузі знань 17 «Електроніка та телекомунікації»
за спеціальністю 172 «Телекомунікації та радіотехніка»
освітньо-професійна програма «Мережеві та інтернет технологій»

на тему:

Прогнозування цін на житло за допомогою технологій машинного навчання

Виконала: студентка групи МІТ -41
Баранюк Катерина Іванівна

Керівник:
доцент кафедри к.т.н., доцент Дахно Наталія Борисівна
мережевих та інтернет технологій

Київ - 2022

Міністерство освіти і науки України
«Київський Національний університет імені Тараса Шевченка»

Факультет інформаційних технологій
Кафедра мережевих та інтернет технологій

ЗАТВЕРДЖУЮ
завідувач кафедри
мережевих та інтернет технологій
_____ Ю.В. Кравченко
«_____»_____2022 року

ЗАВДАННЯ
НА ДИПЛОМНУ РОБОТУ

Здобувачу вищої освіти Баранюк Катерина Іваніва

1.Тема роботи:
Прогнозування цін на житло за допомогою технологій машинного навчання
затверджена на засіданні кафедри МІТ, протокол «24» грудня 2021 р. протокол №8
2.Термін здачі закінченої роботи «30» травня 2022 р.
3.Вихідні дані до проекту (роботи)
Мова програмування – Python
Результати дослідження
4.Зміст пояснювальної записки (перелік питань, що їх потрібно розробити, обсяг – 35-40 стор.)
Вступ

1. Теоретичні основи методів машинного навчання
   1.1.Аналіз алгоритмів машинного навчання
   1.2.Навчання з вчителем
   1.3.Методи вирішення задачі
      1.3.1. Лінійна регресія
      1.3.2. Регресія опорних векторів
      1.3.3. Випадковий ліс
      1.3.4. Стохастичний градієнтний спуск
   1.4.Метрики оцінки похибки
      1.4.1. Середня абсолютна похибка
      1.4.2. Середньоквадратична похибка
      1.4.3. Медіана середньої похибки
      1.4.4. Коефіцієнт детермінації
2. Реалізація та порівняння моделей
   2.1.Вибір середовища розробки

2.2.Обробка та аналіз вхідних даних
2.3.Реалізація методів та результат.
2.3.1. Реалізація та результат лінійної регресі
2.3.2. Реалізація та результат регресії опорних векторів
2.3.3. Реалізація та результати методу випадкового ліса
2.3.4. Реалізація та результати методу градієнтного спуску
2.4.Порівняння моделей
Висновки
3. Перелік графічного матеріалу 8-10 слайдів

Дата видачі завдання _____
Керівник роботи доц. Дахно Н.Б.
(підпис) (посада, прізвище, ім'я, по батькові)
Завдання прийняв до виконання Баранюк К.І.

КАЛЕНДАРНИЙ ПЛАН ВИКОНАННЯ РОБОТИ

Номер	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Підготовчий	16.03.2022	
2	Розділ 1	26.03.2022	
3	Розділ 2	05.04.2022	
4	Доповідь та слайди	19.04.2022	
5	Пояснювальна записка	25.05.2022	

Здобувач вищої освіти_____ _____ К.І. Баранюк
(підпис)

Керівник _____ Н.Б. Дахно
(підпис)

РЕФЕРАТ

Пояснювальна записка: 39 с., 25 рис., 6 табл., 20 джерел.

Об’єкт дослідження – порівняння та застосування алгоритмів машинного навчання для прогнозування даних.

Мета роботи – розглянути та порівняти основні алгоритми машинного навчання для прогнозування на основі різних типів ознак.

Предмет дослідження – прогнозування цін на житло за допомогою алгоритмів.

Нерухоме майно займає центральне місце у будь-якій економічній системі. Найбільшу цінність всім учасників ринку житлової нерухомості представляє інформація про ціни: динаміка цін, причини зміни рівня цін, прогнози розвитку цінової ситуації. Для вивчення цінової ситуації на ринку житлової нерухомості останнім часом все частіше застосовуються методи штучного інтелекту. Одним із класів штучного інтелекту є машинне навчання. Розроблено багато алгоритмів машинного навчання, які використовуються для різного типу даних. Тому визначення найефективнішого алгоритму для прогнозування цін на житло є актуальним на сьогоднішній день.

Метод дослідження – аналіз та порівняння точності для різних алгоритмів.

У ході роботи проведено аналіз сучасних технологій, що використовуються при прогнозуванні за допомогою машинного навчання.

Запропоновано застосувати такі алгоритми та засоби як: Лінійна Регресія, регресія з застосуванням метода стохастичного градієнта, Random Forest, Support Vector Regression.

Визначено модель, яка більш точно та швидко прогнозує дані.

Практичне значення роботи полягає у застосуванні швидкої та найбільш точної моделі на даних, де спостерігаються різні типи даних.

Результати здійснених у дипломній роботі досліджень можуть бути використані будь-яким пересічним громадянином.

Ключові слова:

PYTHON, MACHINE LEARNING, LINEAR REGRESSION, DATA ANALYSIS, SVR, RANDOM FOREST, PANDAS, NUMPY, GRADIENT DESCENT.

ЗМІСТ

ВСТУП

В сучасному світі існує великий обсяг даних. Дані з’являються і оновлюються кожної миті, в такій кількості, що людина більше не в змозі обробляти та аналізувати їх самостійно. Виходячи з цього, існує потреба втілення статистичних моделей та алгоритмів машинного навчання за допомогою мов програмування.

Задача аналізу даних полягає в тому, щоб отримати максимум інформації з набору даних. Метою застосування статистичних моделей та алгоритмів машинного навчання є отримання тенденції, розділення на класи, прогнозування даних, тощо.

Для цього використовуються статистичні моделі та алгоритми машинного навчання. Перш за все, є необхідність з’ясувати яка модель буде найбільш точною для поставленої мети та для даного набору даних.

Сьогодні, найбільш популярною метою є застосування технологій машинного навчання для отримання прогнозу кількісних неперервних даних: вартість, урожайність, попит, тощо.

Наразі існує велика кількість досліджень щодо використання різних моделей, але все ще є потреба в порівнянні найпопулярніших моделей за критеріями точності та часу, що потрібне для обробки та прогнозування.

Також, сьогодні часто дані складаються не лише з одного типу даних. Часто набір даних містить в собі якісні та кількісні типи даних, які поділяються на дискретні та неперервні, порядкові та номінальні відповідно. Що треба враховувати при виборі моделі для поставленої мети.

В цій роботі було порівняно чотири моделі для прогнозування кількісних дискретних даних на основі даних з відкритого джерела, які складаються з номінальних, дискретних та порядкових даних. Та було знайдено найточнішу та найменш часово витратну модель.

Результати цієї роботи можливо застосовувати не лише для прогнозування вартості житла, а й для прогнозування інших кількісних дискретних даних.

1 АНАЛІЗ АЛГОРИТМІВ МАШИННОГО НАВЧАННЯ

1.1 Алгоритми машинного навчання

Модель машинного навчання визначається як математичне представлення результату навчального процесу. Алгоритм машинного навчання – математичний метод знаходження зв’язку та шаблонів в наборі даних [1]. Часто алгоритми машинного навчання походять зі статистики та лінійної алгебри.

В загальному, більшість моделей машинного навчання класифікуються в навчання з вчителем (контрольоване), навчання без вчителя та навчання з підкріпленням.

В цій роботі було порівняно чотири алгоритми навчання з вчителем: Лінійна Регресія, SVR (Support Vector Regression), Random Forest, SGD-Regression з метою дослідження та виявлення найточнішого в прогнозуванні цін на житло.

1.2 Навчання з вчителем

Мета навчання з вчителем – створити функцію, що може бути натренована використовуючи тренувальний набір даних, яка потім застосовується до тестових даних та показує результат прогнозування. [1]

Модель навчається, доки не почне розпізнавати основні шаблони та співвідношення поміж вхідними даними та вихідними лейблами.

Навчання вчителем добре підходить для вирішення задач класифікації та регресії [2].

Регресія використовується для того, щоб розуміти співвідношення між залежними та незалежними даними.

1.3 Лінійна Регресія

Лінійна Регресія – це один з найбільш відомих підходів як в статистиці, так і в машинному навчанні. Лінійна Регресія може бути розглянута як алгоритм машинного навчання, що дозволяє зіставляти чисельні вхідні дані до чисельних вихідних даних. Іншими словами, Лінійна Регресія – це спосіб змоделювати відношення між змінними [3]. На прикладі лінійної регресії на рисунку 1.1 показано приклад лінійної регресії та відношення змінних.

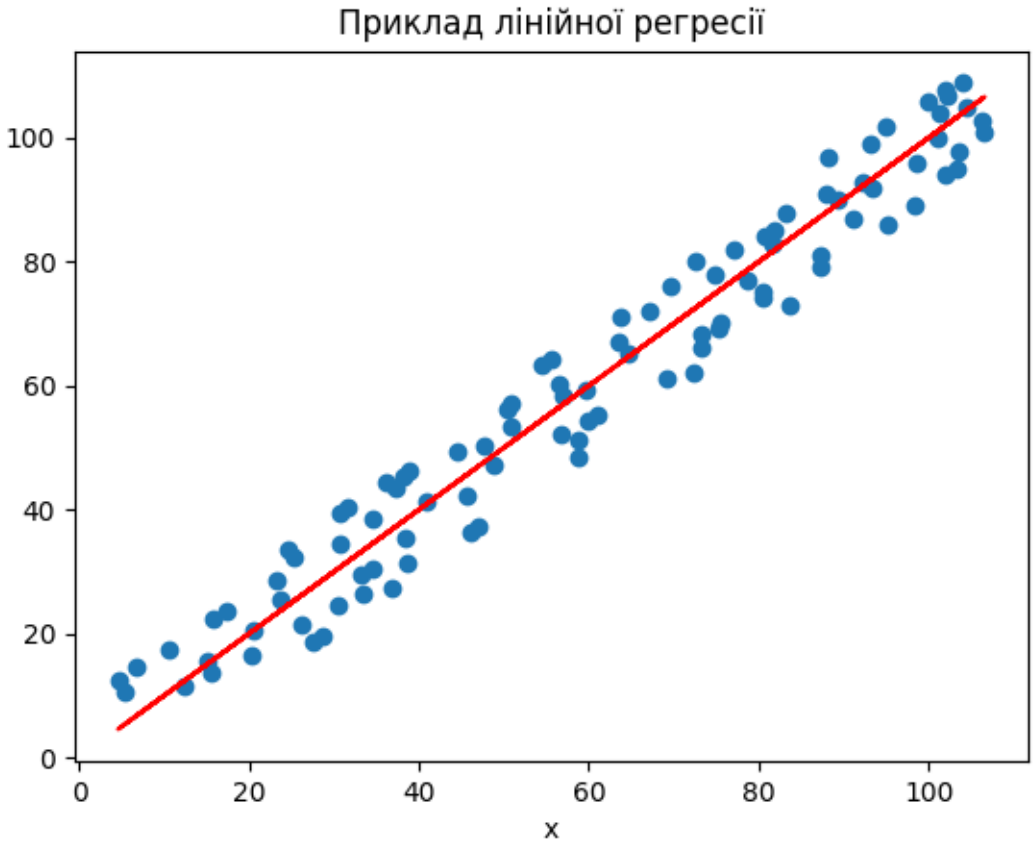

Рисунок 1.1 Лінійна Регресія

Лінійна Регресія описується формулою прямої [4]:

$$y = wx + b \quad (1.1)$$

Де y – залежна змінна,

w – кутовий коефіцієнт регресії,

x – незалежна змінна,

b – вільний член.

Кутовий коефіцієнт регресії знаходяться з використанням метода найменших квадратів [5].

Метод найменших квадратів – це підхід в статистиці для пошуку найкращої відповідності. Головна мета – зменшити суму квадратів похибки, настільки, наскільки це можливо [6]. Навіть незважаючи на те, що цей метод вважається найкращим методом знаходження найбільш відповідної прямої, він має декілька обмежень:

– цей метод чутливий до викидів у даних. Великі викиди можуть відхилити результати аналізу найменших квадратів;

– метод демонструє відношення лише між двома змінними;

– метод нерівномірний, коли дані розподілені нерівномірно.

Формула методу для багатомірного випадку має вигляд :

$$w = (X^T X)^{-1} X^T y$$

w – кутовий коефіцієнт;

X – матриця незалежних змінних;

y – залежна змінна.

1.4 SVR (Support Vector Regression)

Support Vector Regression – це тип методу опорних векторів, що опирається на лінійну та нелінійну регресію. Особливою властивістю метода є безперервне зменшення емпіричної похибки класифікації та збільшення зазору [7].

Основна концепція метода – перевести вхідні вектори в простір більш високої розмірності та пошук поділяючої гіперплощини с найбільшим зазором в цьому просторі.

Support Vector Regression схожа на лінійну регресію в однаковій формулі прямої (1.1). Але в методі опорних векторів пряма лінія відноситься до гіперплощини. Точки даних по обидва боки від гіперплощини, які є найближчими до іншої гіперплощини, називаються опорними векторами, які використовуються для побудови граничної лінії [7].

На відміну від інших моделей регресії, що мінімізують похибку поміж реальними та прогнозованими значеннями, SVR вміщає найкращу лінію в порогове значення (відстань між гіперплощиною та граничною лінією). Виходячи з цього можна стверджувати, що Support Vector Regression намагається задовільнити умову [7]:

$$-a < y - wx + b < a$$

Де a – порогове значення,

y – залежна змінна ,

w – кутовий коефіцієнт,

x – незалежна змінна,

b – вільний член.

1.5 Random Forest Regressor

Random Forest – алгоритм, що може використовуватись і для вирішення задачі класифікації, і для вирішення задач регресії. Моделі Random Forest побудовано на основі використання набору дерев рішень, що базуються на основі навчальних даних [8]. Приклад на рисунку 1.2

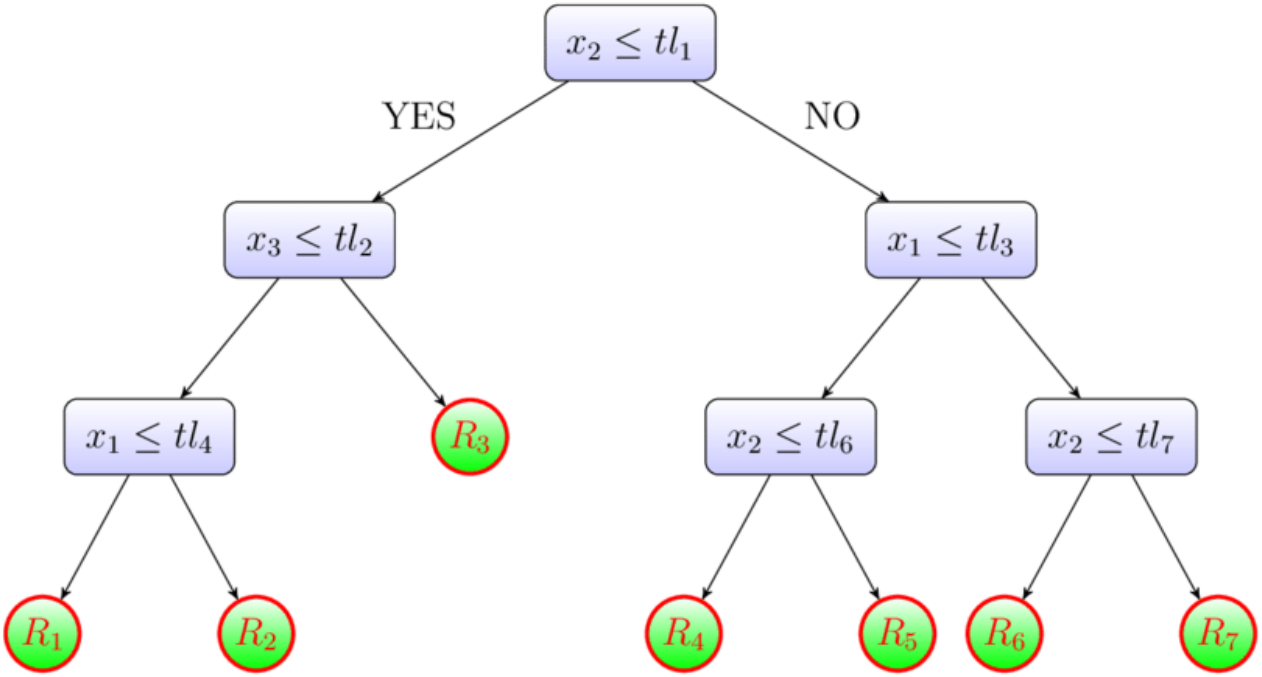

Рисунок 1.2 Приклад Дерева Рішень

Замість того, щоб брати вихідне значення з одного дерева, алгоритм Random Forest робить прогнозування на основі середнього прогнозу набору дерев рішень. Цей підхід дозволяє зменшити вірогідність оверфітінгу моделі [9].

Алгоритм приймає три важливі гіперпараметри:

– кількість дерев: чим більша кількість дерев, тем менше оверфітінг моделі;

– метод оцінки похибки: визначає яку метрику похибки використовувати для вимірювання яксоті розколів у деревах у випадковому лісі. Це може бути або середньоквадратична похибка, або середня абсолютна похибка;

– максимальна кількість ознак: контролює кількість ознак, які впливають на побудову дерев.

1.6 Стохастичний градієнтний спуск

Градієнт – вектор, який вказує на найбільше збільшення функції. З цього випливає, що від'ємний градієнт – вектор, що вказує на найбільше зменшення функції. Логіка градієнтного спуску полягає в тому, щоб рухатись в напрямку від'ємного градієнта. Градієнт функції f в точці (x, y) виражається формулою [10]:

$$\nabla f(x,y) = \frac{df}{dx} i + \frac{df}{dy} j$$

Графічний приклад градієнтного спуску (рисунок 1.3)

Стохастичний градієнтний спуск – це ітеративний метод для оптимізації цільової функції з відповідним властивостями гладкості. Стохастичним він називається через особливість алгоритму. Коли, градієнтний спуск використовує в розрахунку увесь набір даних, даний метод використовує випадкову підмножину з набору даних [11]. Виходячи з цього, цей метод можна вважати стохастичною апроксимацією градієнтного спуску, оскільки він рахує середнє серед всіх підмножин [12].

Стохастичний градієнт виражається формулою:

$$\omega^{(t+1)} = \omega^{t} - n_t \cdot \sum_{i=1}^{l} \nabla L_i(\omega^{(t)}, x_i)$$

Де L – функція втрат(лосс-функція).

Loss-функція – це функція, що рахує відстань між поточним результатом та очікуваним результатом [13].

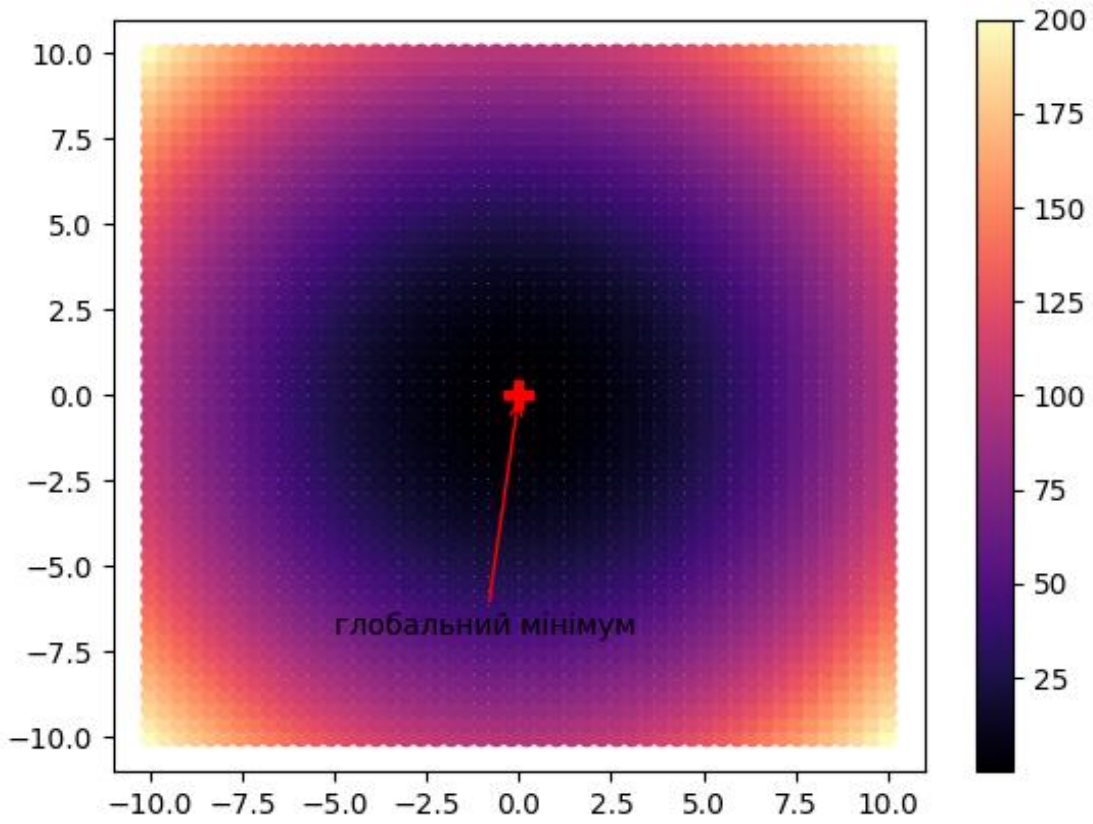

Рисунок 1.3 Графічний приклад градієнтного спуску

1.7 Метрики оцінки похибки

Для того, щоб оцінити якість прогнозу моделей, було використано чотири метрики. Середня абсолютна похибка, середньоквадратична похибка, медіана середньої похибки та коефіцієнт детермінації.

1.7.1 Середня абсолютна похибка

Середня абсолютна похибка (Mean Absolute Error, MAE) – оцінює похибку прогнозу рахуючи середнє всіх абсолютних значень похибок [14]:

$$MAE = \frac{\sum_{i=0}^{n} |y_i - \hat{y}_i|}{n} \quad (1.5)$$

Де y – цільові значення;

$\hat{y}$ – прогнозоване значення;

n – кількість значень.

Модель вважається точнішою зі зменшенням значення середньої абсолютної похибки.

1.7.2 Середньоквадратична похибка

Середньоквадратична похибка схожа до середньої абсолютної похибки. Цей метод вимірює помилку прогнозу беручи середнє значення всіх квадратів абсолютних значень похибок [14]:

$$MSE = \frac{\sum_{i=0}^{n} (y_i - \hat{y}_i)^2}{n} \quad (1.6)$$

Де y – цільові значення;

$\hat{y}$ – прогнозоване значення;

n – кількість значень.

Відповідно до середньої абсолютної похибки модель вважається точнішою з меншим значенням MSE.

1.7.3 Медіана середньої похибки

Медіана середньої похибки – медіана всіх абсолютних різниць поміж прогнозованим значенням та цільовим. На відміну від двох попередніх метрик,

середня асболютна похибка є більш стійкою до викидів за рахунок використання медіани замість середнього [14]:

$$\mathrm{M}edianAbsError = median(|y_1 - \widehat{y_i}| \dots |y_n - \widehat{y_n}| \quad (1.7)$$

Низьке значення даної метрики означає хороший прогноз модель.

1.7.4 Коефіцієнт детермінації

Коефіцієнт детермінації R^2 – це співвідношення між дисперсію, що пояснена аналізованої моделі та загальною дисперсією. Ця метрика часто використовується для того, щоб дізнатись як добре регресійна модель прогнозує дані. Значення коефіцієнта існує на проміжку [0,1], де значення близьке до 1 вважається кращим. Тобто, 1 вважається ідеальною детермінацією та 0 вважається відсутністю детермінації [14].

Коефіцієнт детермінації рахується за формулою

$$R^2(y, \hat{y}) = 1 - \frac{\sum_{i=0}^{n-1}(y_i - \widehat{y_i})^2}{\sum_{i=0}^{n-1}(y_i - \bar{y})^2}$$

Де y – цільові значення;

$\hat{y}$ – прогнозоване значення;

n – кількість значень;

$\bar{y}$ – середнє арифметичне

$$\bar{y} = \frac{\sum_{i=0}^{n-1} y_i}{n}$$

2. РЕАЛІЗАЦІЯ ТА ПОРІВНЯННЯ МОДЕЛЕЙ НА ВІДКРИТИХ ДАНИХ

2.1 Вибір середовища розробки

На сьогоднішній день існує безліч мов програмування та середовищ для обробки даних. Однак, найчастіше з них використовуються Python, R або MatLab. Для виконання цієї роботи було обрано Python. Оскільки ця мова входить в 5 найпопулярніших мов програмування (Рисунок 3) [15]. В практиці частіше використовують саме цю мову для аналізу, прогнозування даних та побудови алгоритмів машинного навчання.

Dec 2021	Programming language	Share
1	Python	29.69 %
2	Java	14.98 %
3	JavaScript	7.85 %
4	R	6.95 %

Рисунок 2.1 Рейтинг мов програмування станом на грудень 2021

Також, основною перевагою Python є безліч існуючих бібліотек. В ході виконання цієї роботи були використанні такі бібліотеки як: Scikit-learn, Pandas, Numpy, Matplotlib.

– Scikit-learn: використання алгоритмів машинного навчання та нормалізації даних.

– Pandas: обробка даних, групування та конвертація типів.

– NumPy: робота з числами.

– Matplotlib: візуалізація даних.

2.2 Обробка та аналіз вхідних даних

Для виконання даної дослідницької роботи було використано дані з відкритого джерела. Дані було попередньо розділені на тестову і тренувальну групи.

Перш за все, було визначено тип даних та конвертовано будь-які нечисельні ознаки в чисельні (таблиця 2.1) за допомогою вбудованої функції Pandas.

Таблиця 2.1 Кількість нечисельних даних

Кількість нечисельних даних перед конвертацією	Кількість нечисельних даних після конвертації
43	0

Всього ознак: 81. Та залежна зміна, яка є метою прогнозування.

Було проведено попередній аналіз даних та виявлено, як в середньому змінюється ціна в залежності від року побудови житла (Рисунок 2.2) та від року продажі (Рисунок 2.3)

З цих графіків видно, що ціна зростає в залежності від року побудови та немає сильних викидів, однак ціна дуже залежна від року продажі. Ми можемо спостерігати різке зниження цін після 2008 року. Що може бути пояснено глобальною світовою кризою 2008 року [16].

Було визначено ступінь кореляції даних між собою, це важливий показник чистоти даних [17]:

$$r_{xy} = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2 \sum(y_i - \bar{y})^2}}$$

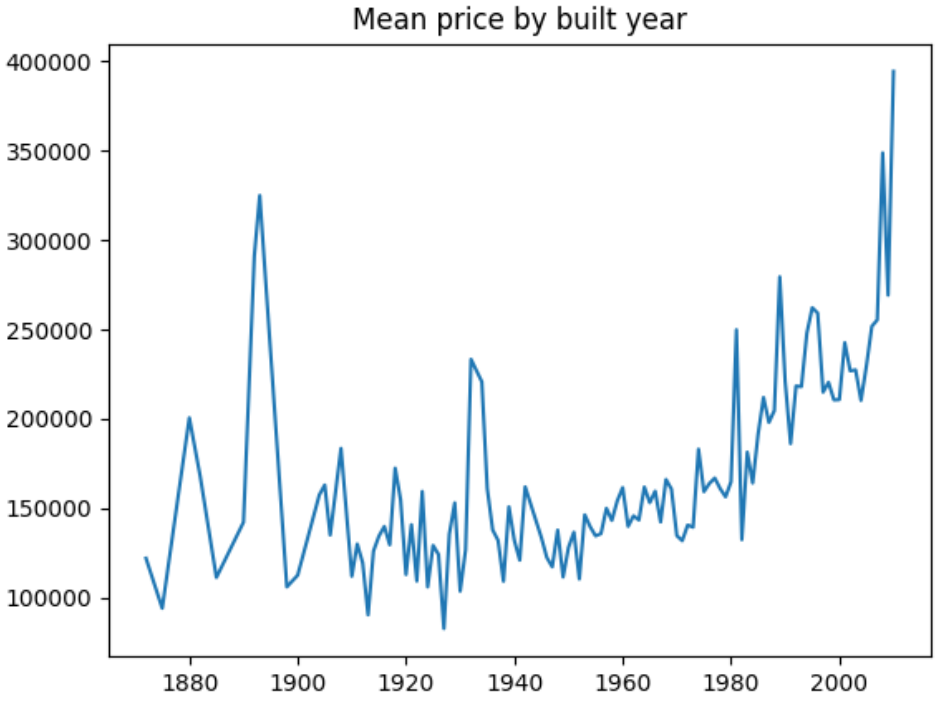

Рисунок 2.2 Залежність ціни від року побудови майна

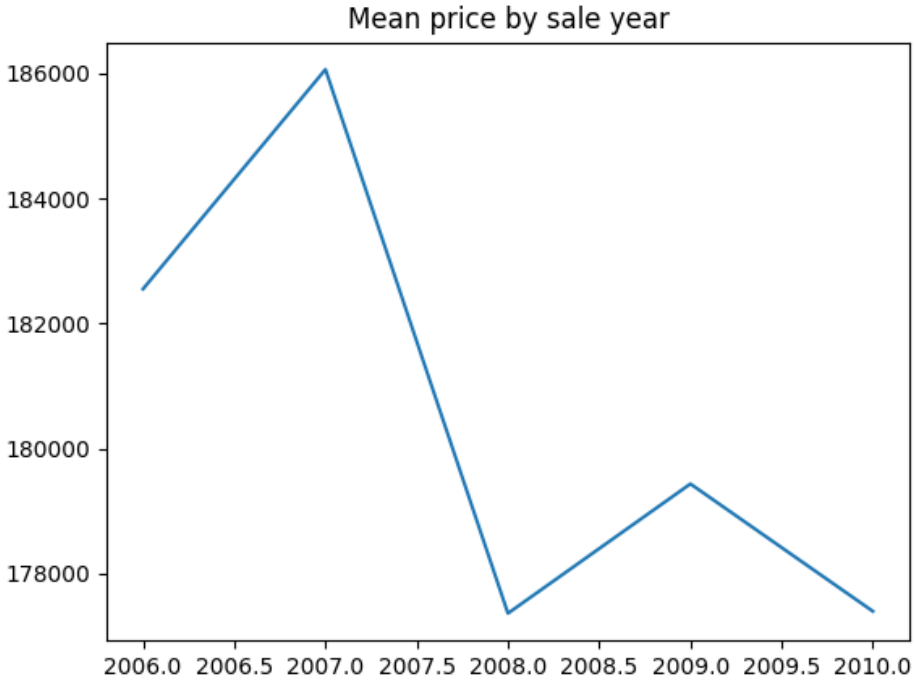

Рисунок **Помилка! У документі відсутній текст указаного стилю.**.3 Залежність ціни від року продажі майна

Кореляцію було візуалізовано за допомогою хітмап (Рисунок 2.4)

За рисунком показаним нижче видно, що більшість ознак мають високу кореляцію з залежною змінною. Це значить, що ціна залежить від цих ознак та їх треба враховувати в моделях.

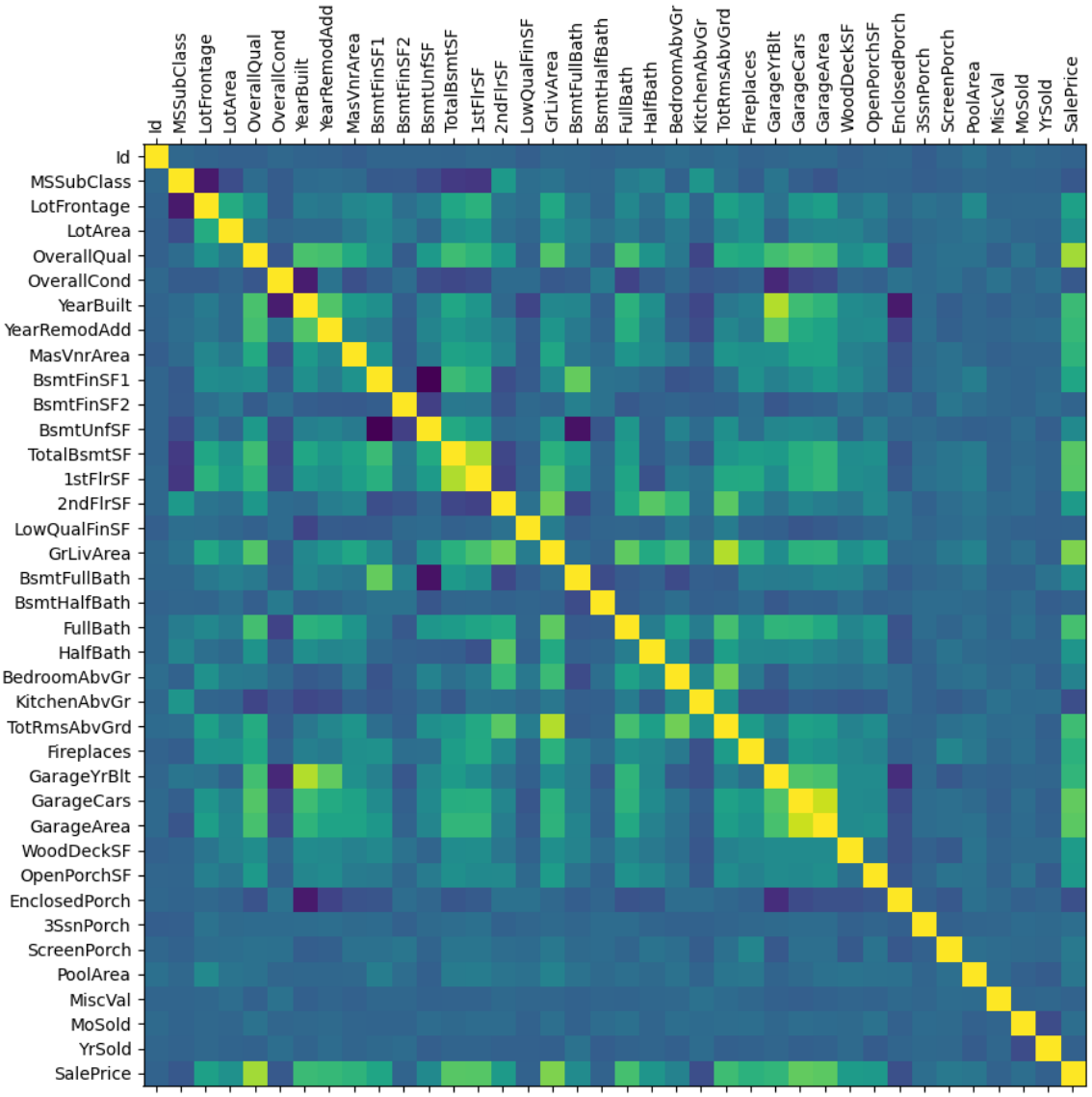

Рисунок 2.4 Кореляція ознак

2.3 Реалізація та результат лінійної регресії

Лінійна Регресія була реалізована за допомогою відкритої бібліотеки Scikit-learn [18]. Ця модель використовує метод найменших квадратів для мінімізації кутового нахилу, як і було зазначено в першому розділі. Лінійна Регресія є класичною моделлю та не приймає додаткових параметрів, що можуть вплинути на результат. За метриками та графічним результатом (табл. 2.2, Рисунок 2.5) видно, що Лінійна Регресія не підходить для вирішення поставленої задачі.

Таблиця 2.2 Метрики результату виконання лінійної регресії

R^2	0.863343
MAE	19368.2
MSE	9.41E+08
Median abs error	0.863343

Лінійна Регресія виконується швидко та не займає багато обчислювальних потужностей (рисунок 2.6)

2.4 Реалізація та результат регресії опорних векторів

Support Vector Regression була реалізована за допомогою відкритої бібліотеки [11].

Ступінь регресії дорівнює 3, тобто регресія поліноміальна. Було порівняно два параметри ядра: linear та rbf (рисунок 2.7, рисунок 2.8, таблиця 2.3)

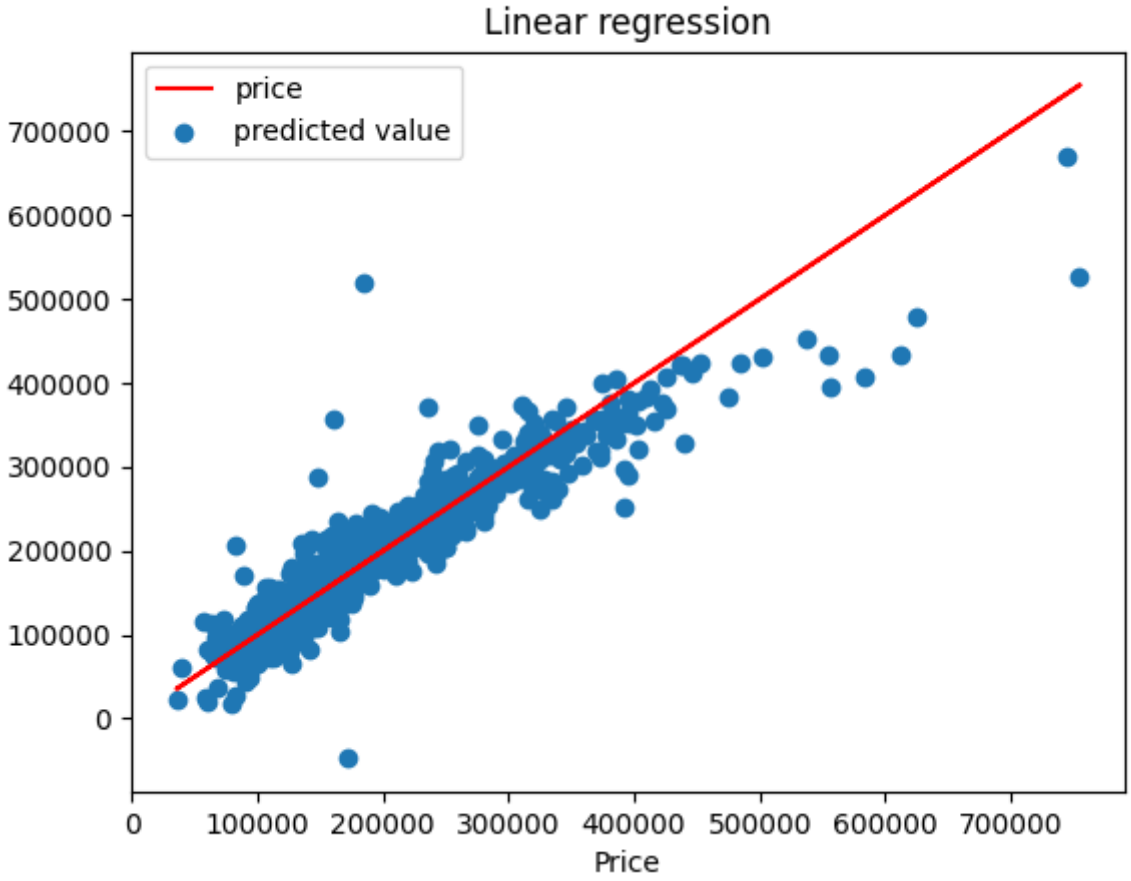

Рисунок **Помилка! У документі відсутній текст указаного стилю.**.5 Результат лінійної регресії

```
Середній час виконання лінійної регресії:0.011615592956542969c
```

Рисунок 2.6 Середній час виконання лінійної регресії

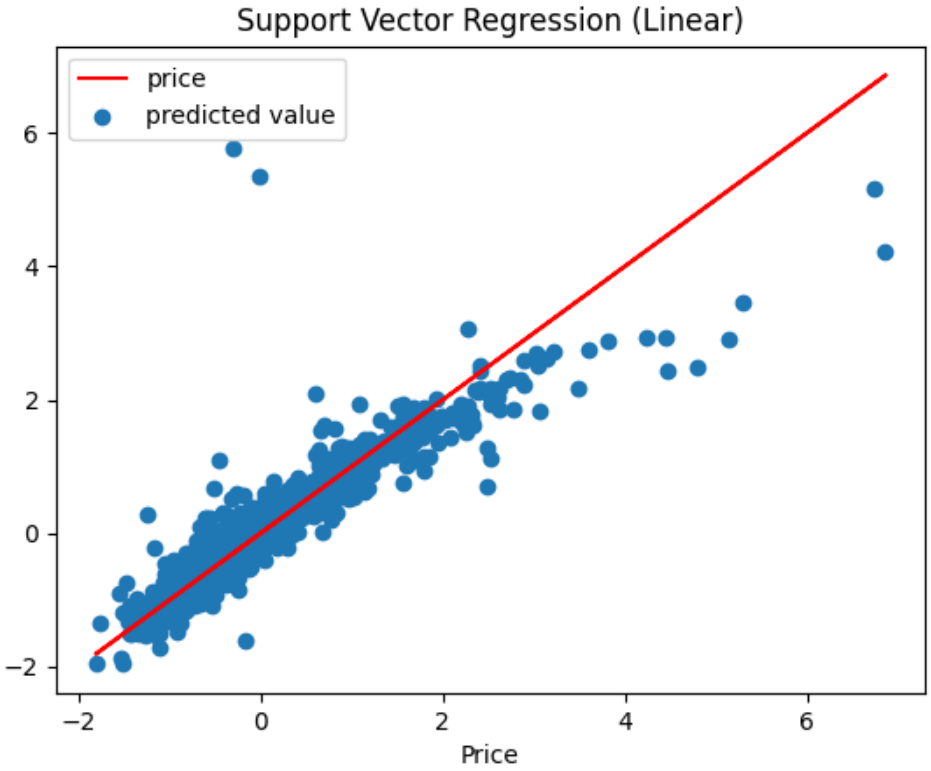

Рисунок 2.7 Результат роботи SVR (Linear)

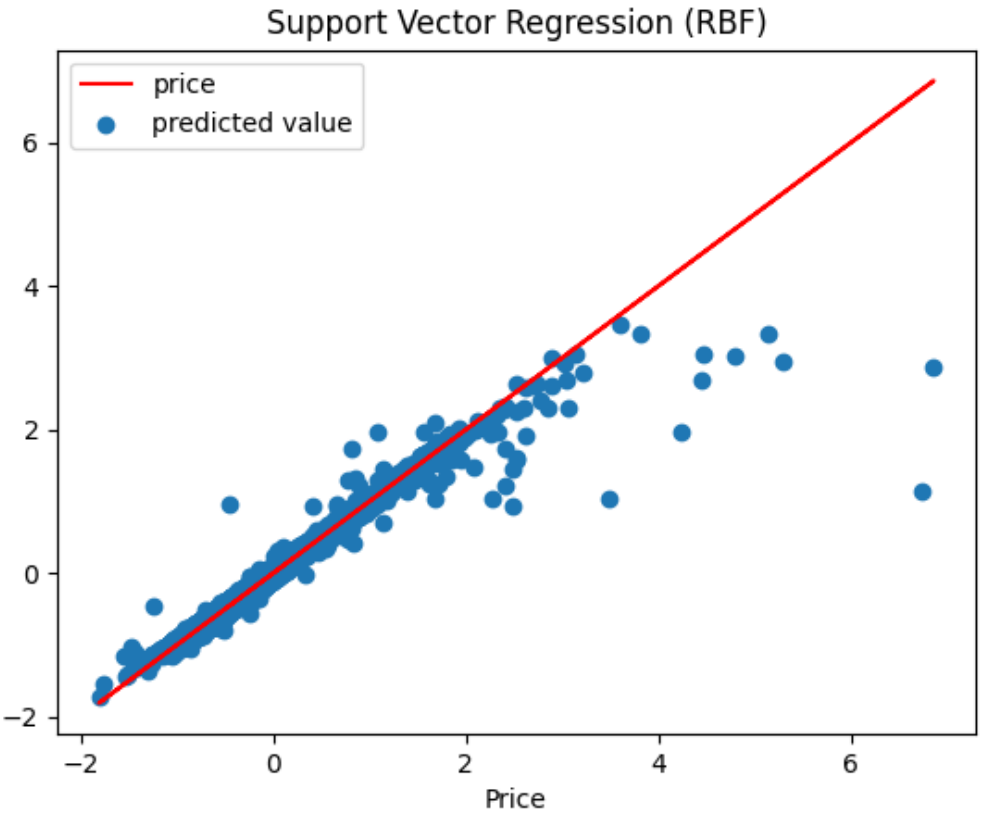

Рисунок 2.8 Результат роботи SVR(RBF)

Таблиця 2.3 Результати роботи SVR

R^2	0.907283
MAE	0.124091
MSE	0.092717
Median abs error	0.907283

За наведеною вище графіками та таблицею не складно помітити, що дана модель вирішую задачу не ідеально, але з відносно непоганими метриками.

Час роботи моделі незначний (рисунок 2.9).

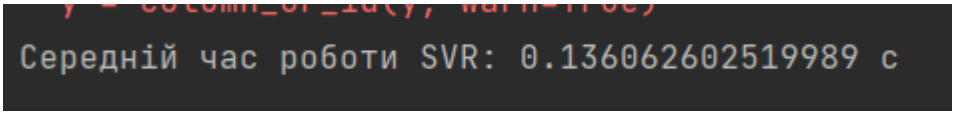

Рисунок 2.9 Середній час роботи SVR

2.5 Реалізація та результати методу випадкового ліса

Хоча даний метод і був реалізований за допомогою бібліотеки, він потребував додаткових обчислювальних операцій для визначення параметрів зазначених у пункті 1.5. Для оцінки якості моделі була використана метрика середньоквадратичної похибки.

MSE показало найменшу похибку при використанні 45% відсотків даних (рисунок 2.10)

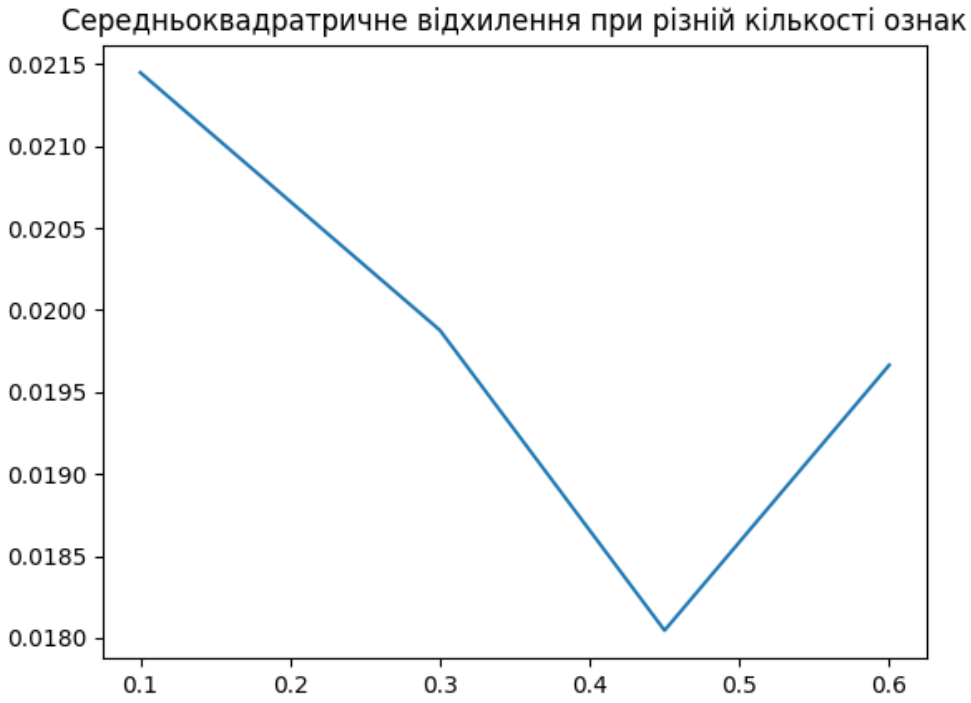

Рисунок 2.10 Середньоквадратичне відхилення при різній кількості ознак та при встановленні кількості 600 дерев (рисунок 2.11)

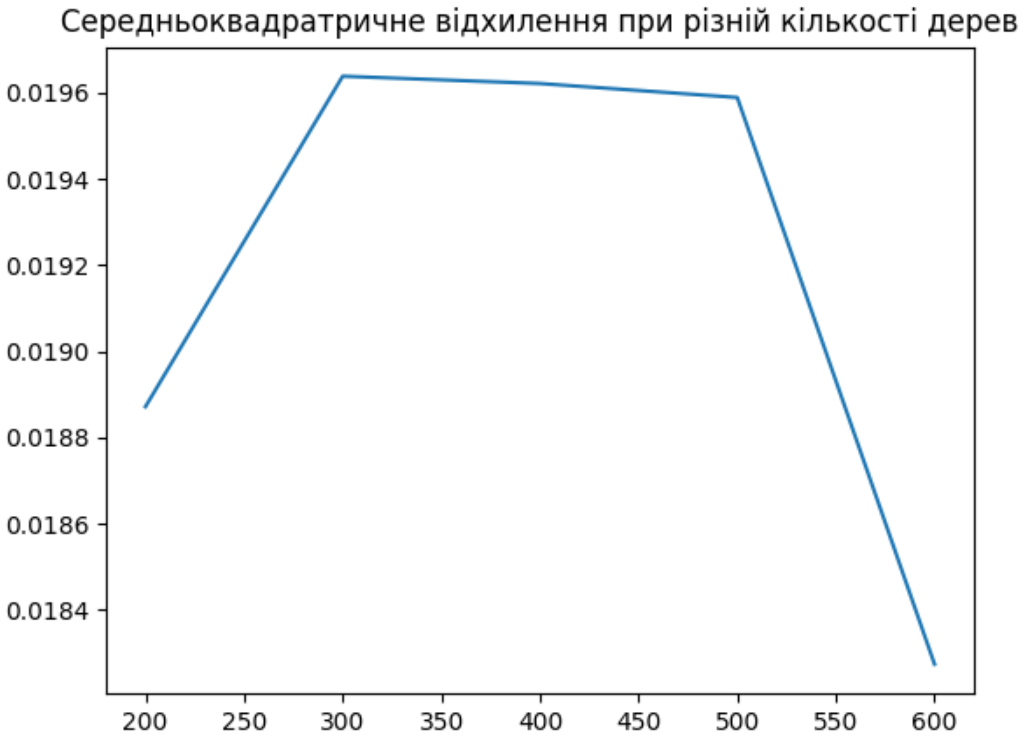

Рисунок 2.11 Середньоквадратичне відхилення при різній кількості дерев

Враховуючи це, модель була налаштована та було отримано такі результати (таблиця 2.4):

Таблиця 2.4 Результати Random Forest

R^2	0.981083
MAE	0.076542
MSE	0.018917
Median abs error	0.04516

Та графічно результат виглядає так (рисунок 2.12):

Метод показав високі метрики та на графіку вище видно, що відхилення від таргету незначне.

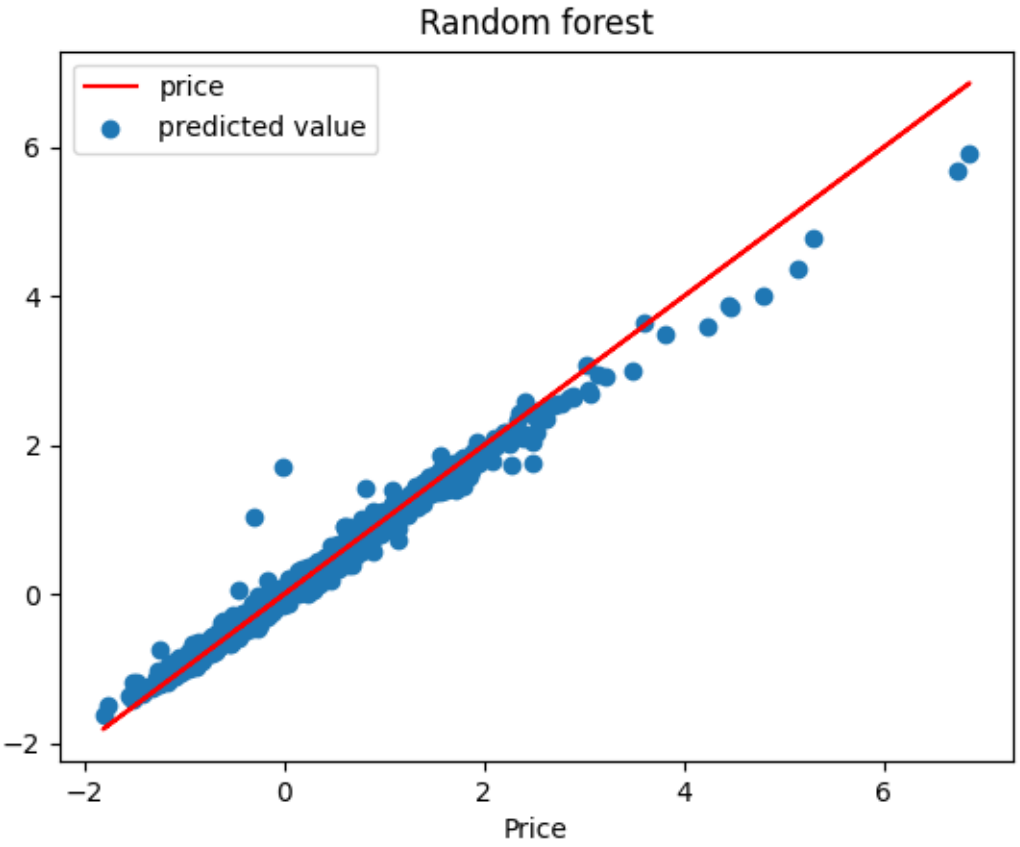

Рисунок 2.12 Графічне зображення результатів Random Forest

Час роботи цього алгоритму досить високий (рисунок 2.13):

```
Середній час роботи Випадкового Лісу: 5.46190128326416 с
```

Рисунок 2.13 Час роботи Random Forest

2.6 Реалізація та результати роботи SGD Regression

Для реалізації цього алгоритму було порівняно два параметри функції втрат (loss).

За умовчанням застосовується метод найменших квадратів, аналогічно до лінійної регресії, але він показав неприйнятні результати (рисунок 2.14, таблиця (2.5):

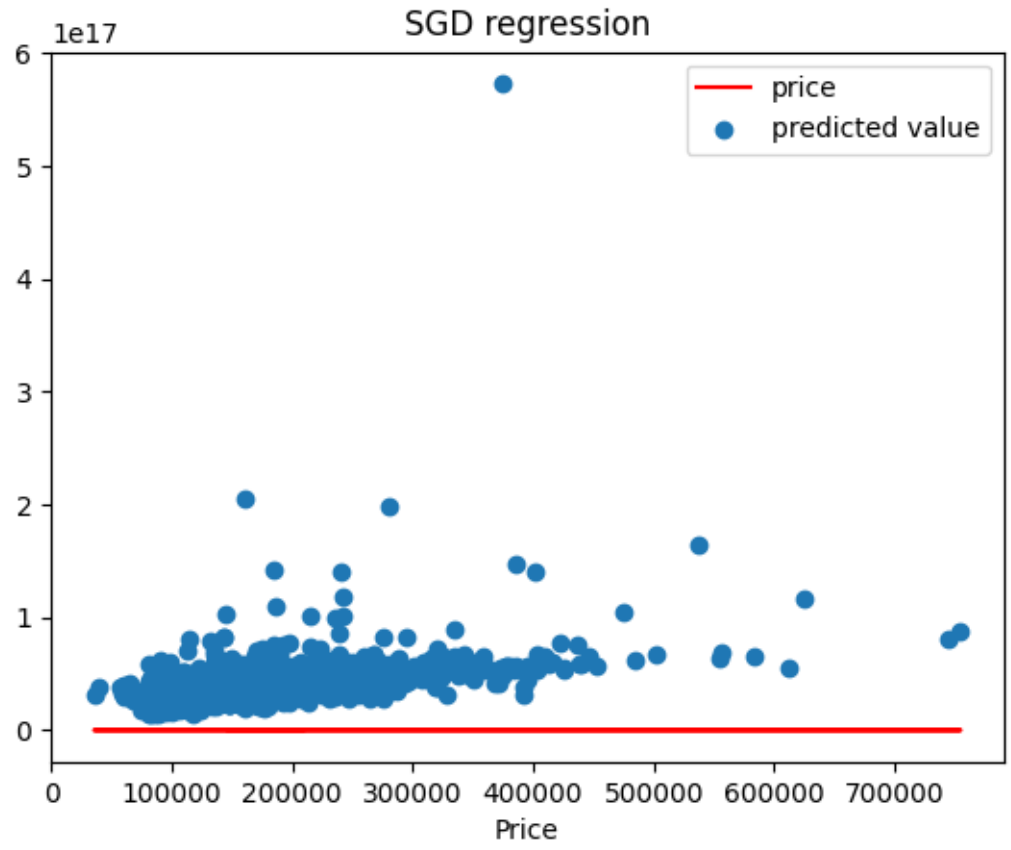

Рисунок 2.14 Результати роботи SGD Regression з функцією втрат за замовчуванням

Таблиця 2.5 Результати роботи SGD Regression з функцією втрат за замовчуванням

R^2	-4.45E+23
MAE	5.28E+16
MSE	3.06E+33
Median abs error	5.08E+16

Тому було вирішено спробувати замінити цей параметр на параметр “huber”.

Цей метод модифікує «squared_error», щоб менше зосереджуватися на виправленні викидів, перемикаючись з квадратних втрат на лінійні за відстань епсілон.

Епсілон – порогове значення при якому стає менш важливим отримати точне значення.

На графіку (рисунок 2.15) показано при якому значенні епсілон можливо отримати найменшу середньоквадратичну похибку.

На графіку видно, що найменша середньоквадратична похибка при значенні 0.05.

Результати роботи даного методу з налаштуванням вище (рисунок 2.16, таблиця 2.6)

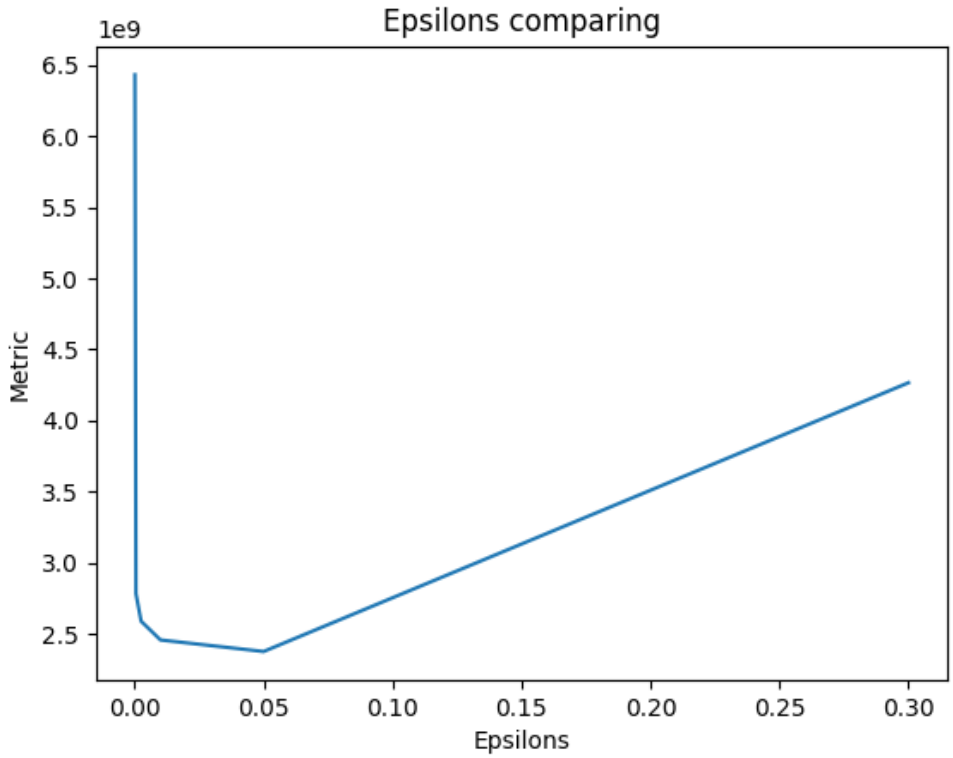

Рисунок 2.15 Порівняння значень епсілон

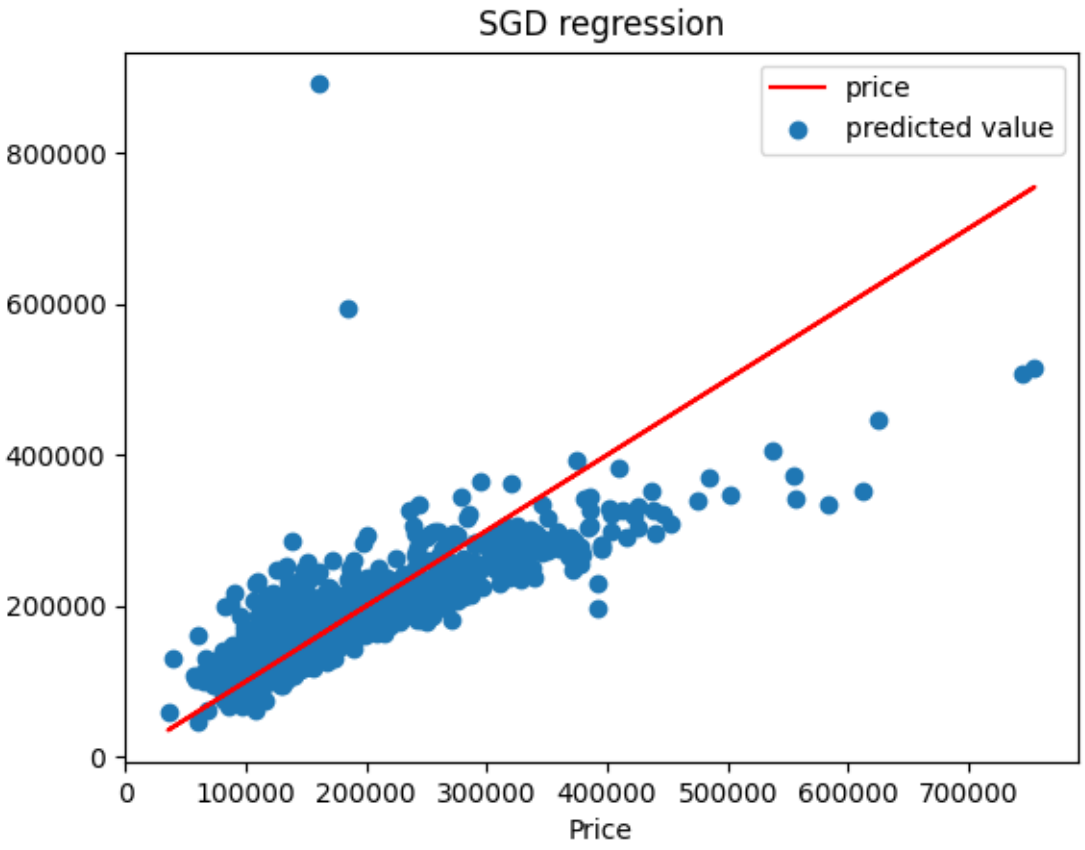

Рисунок 2.16 Результати роботи з підібраними параметрами

Таблиця 2.6 Результати роботи з підібраними параметрами

R^2	0.641076348
MAE	30120.56073
MSE	2470369243
Median abs error	20079.89225

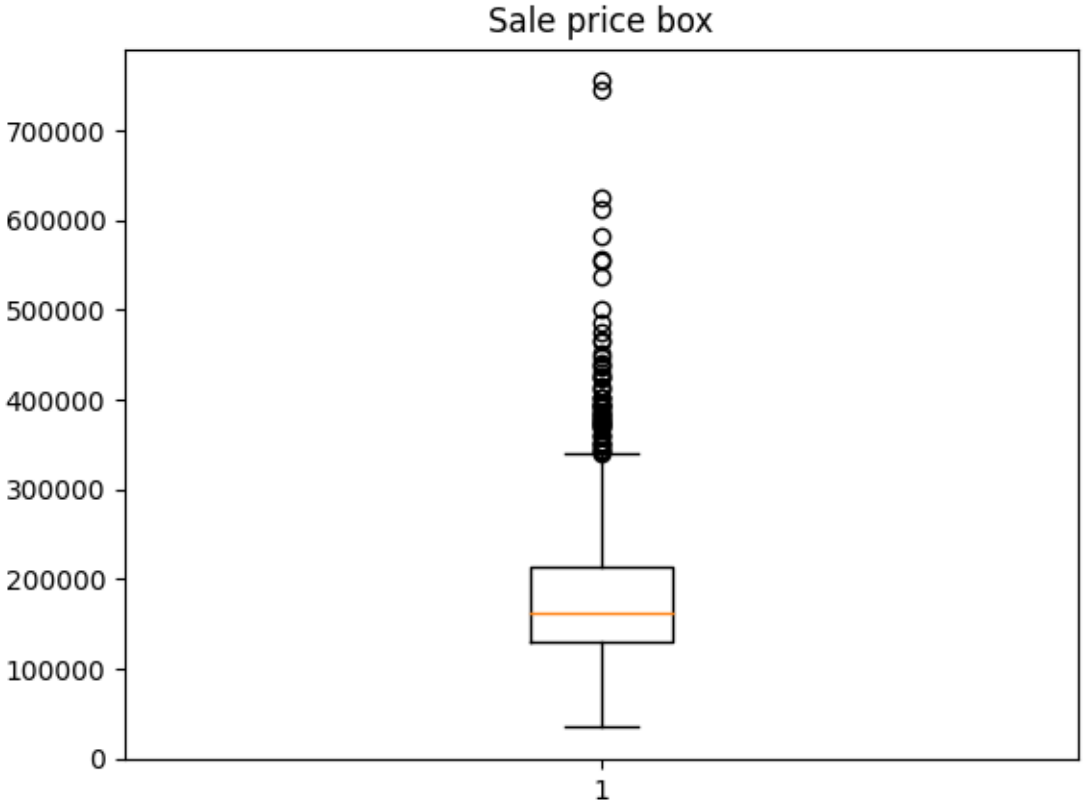

Рисунок 2.17 Виброси в даних

При прогнозуванні модель займає порівняно небагато часу (Рисунок 2.18):

```
Середній час виконання SGDRegressor 0.011513165950775147 с
```

Рисунок 2.18 Час виконання SGD Regressor

2.7 Порівняння моделей

Метою роботи стоїть порівняння різних моделей для поставленої задачі, прогнозування цін на житло.

Моделі порівнюються за 4 метриками та часом.

Графічне порівняння R^2 (рисунок 2.19):

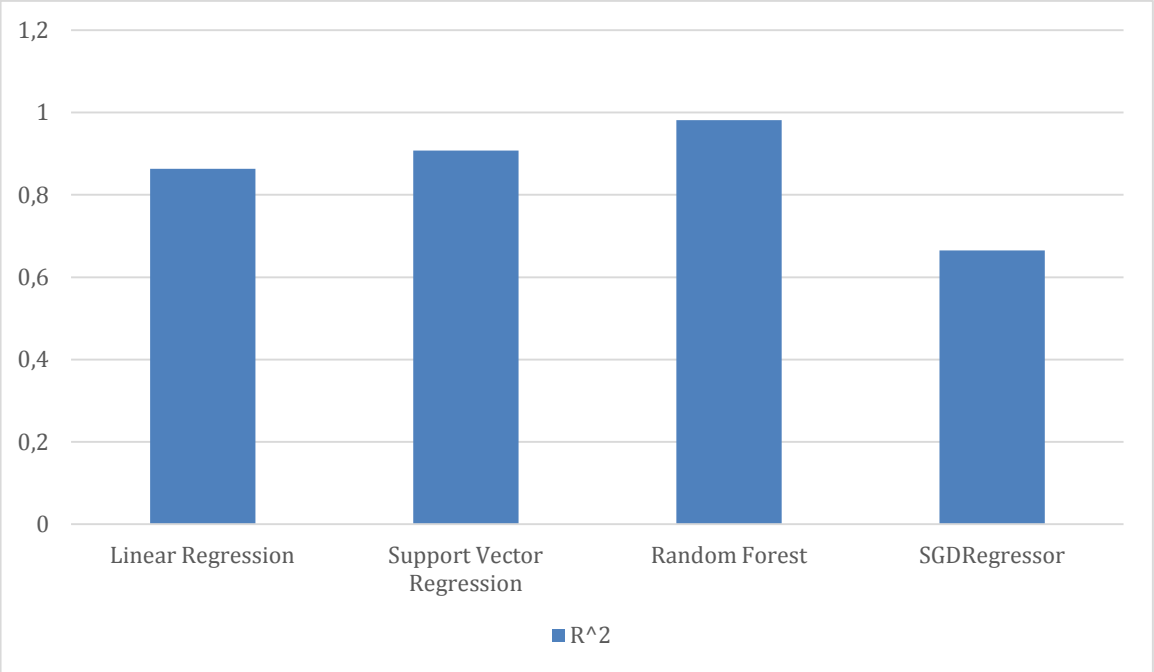

Рисунок 2.19 Діаграма метрики R^2

Як було зазначено раніше, ця метрика вважається кращою, коли вона ближче до 1.

Тому слід визначити, що згідно даної метрики метод Random Forest спрогнозував найточніше.

Графічне порівняння інших метрик показано на Рисунок 2.20 – 2.22.

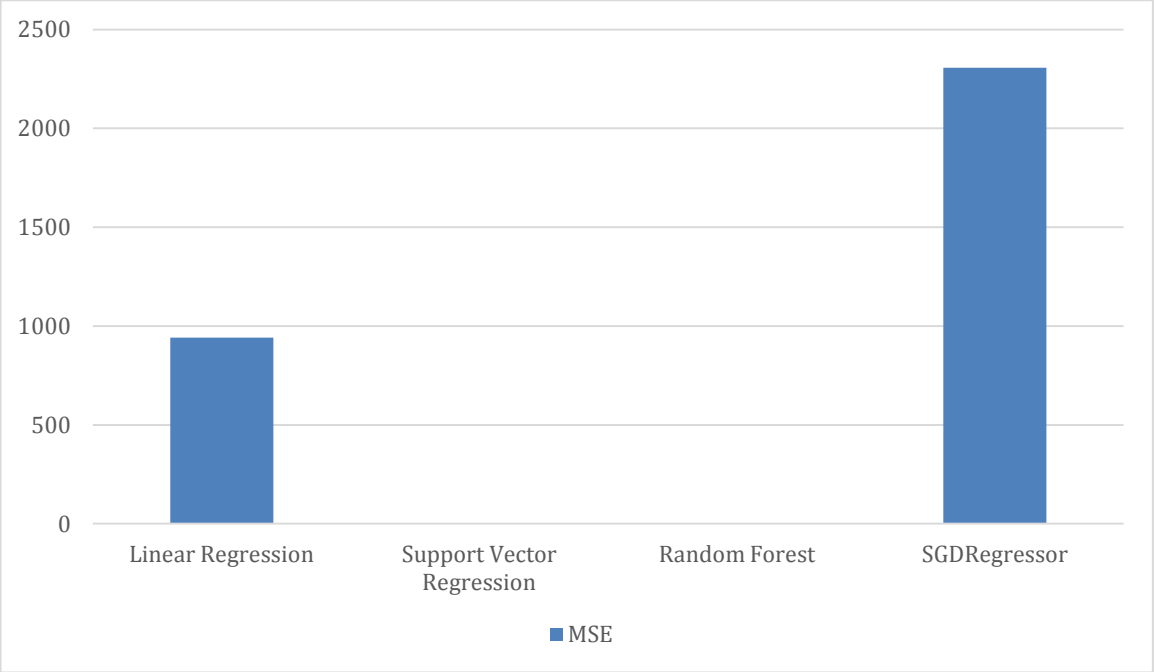

Рисунок 2.20 Порівняння середньоквадратичної похибки

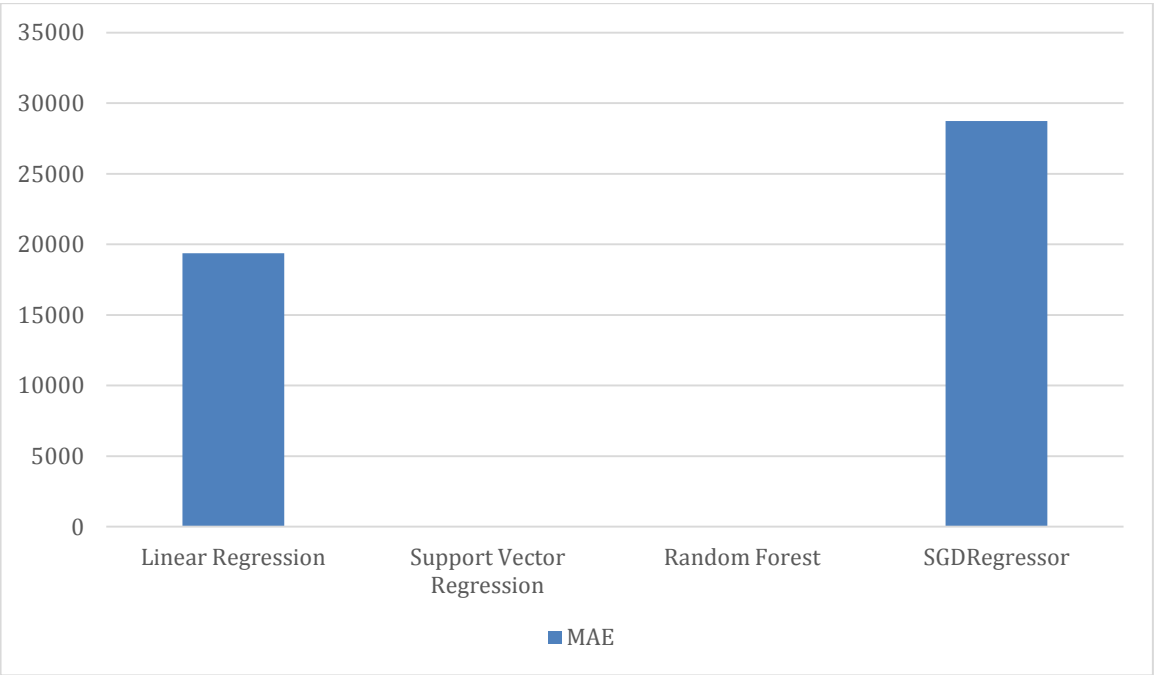

Рисунок 2.21 Порівняння середньої абсолютної похибки

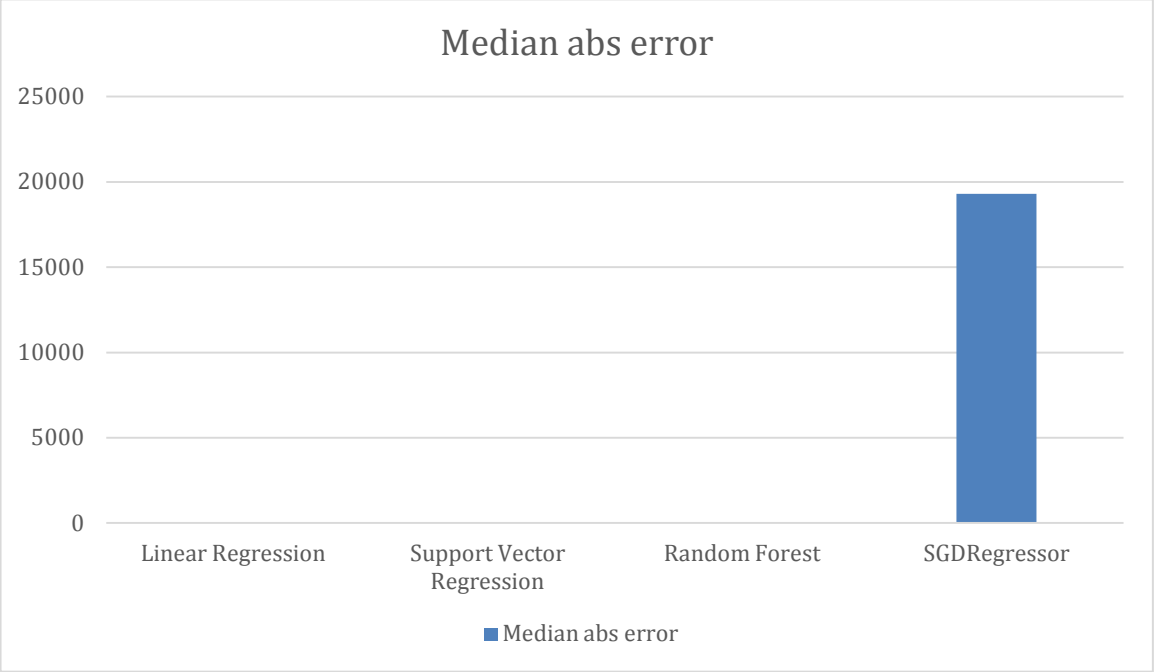

Рисунок 2.22 Порівняння медіани абсолютної похибки

Неозброєним поглядом видно, що Лінійна Регресія та SGD Regression взагалі не підходять для даної задачі. Оскільки, метрики досить високі, то на діаграмах неможливо побачити різницю між регресією з методом опорних векторів та методом Random Forest. Тому, їх було порівняно на окремих діаграмах (рисунок 2.23-2.25)

Чисельно та за методом краще працює метод Random Forest. Це пояснюється кількістю дерев та великою кількістю умов, що модель бере до уваги [19].

Також слід зазначити, що регресії показали поганий результат через кількість гарно скорельованих ознак та через те, що в класичному випадку регресія (функція мінімізації) враховує лише одну ознаку [20]. Регресія з використанням метода стохастичного спуску теж показала незадовільні результати. Це може бути пов’язано з неправильним вибором ознак.

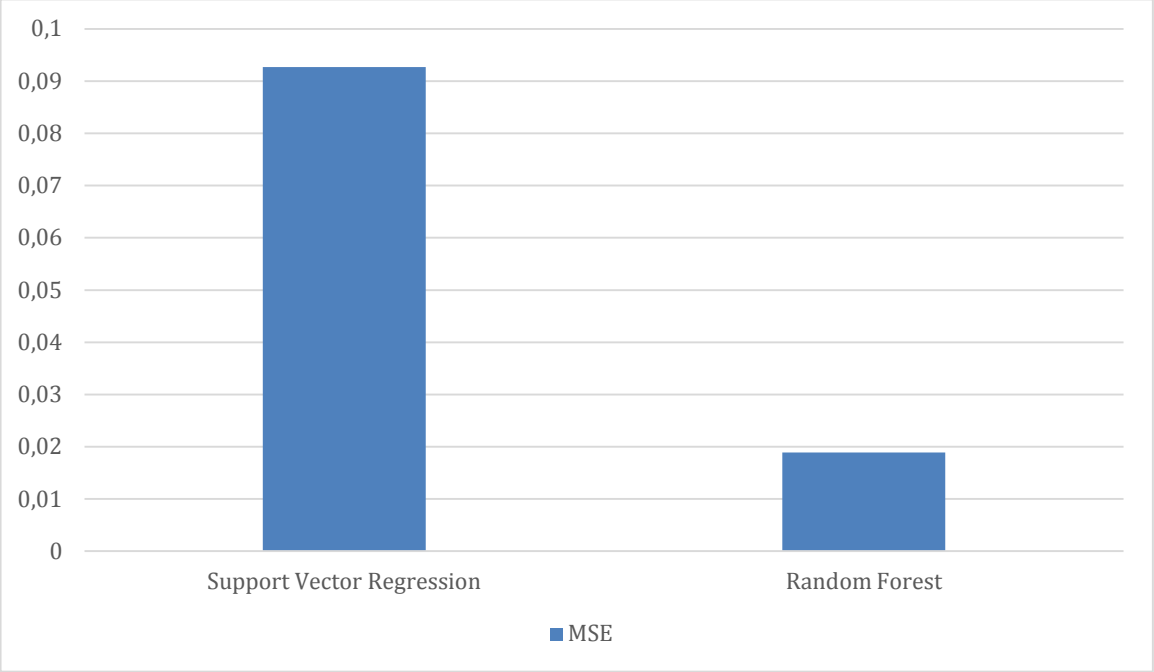

Рисунок 2.23 Часткове порівняння середньоквадратичної похибки

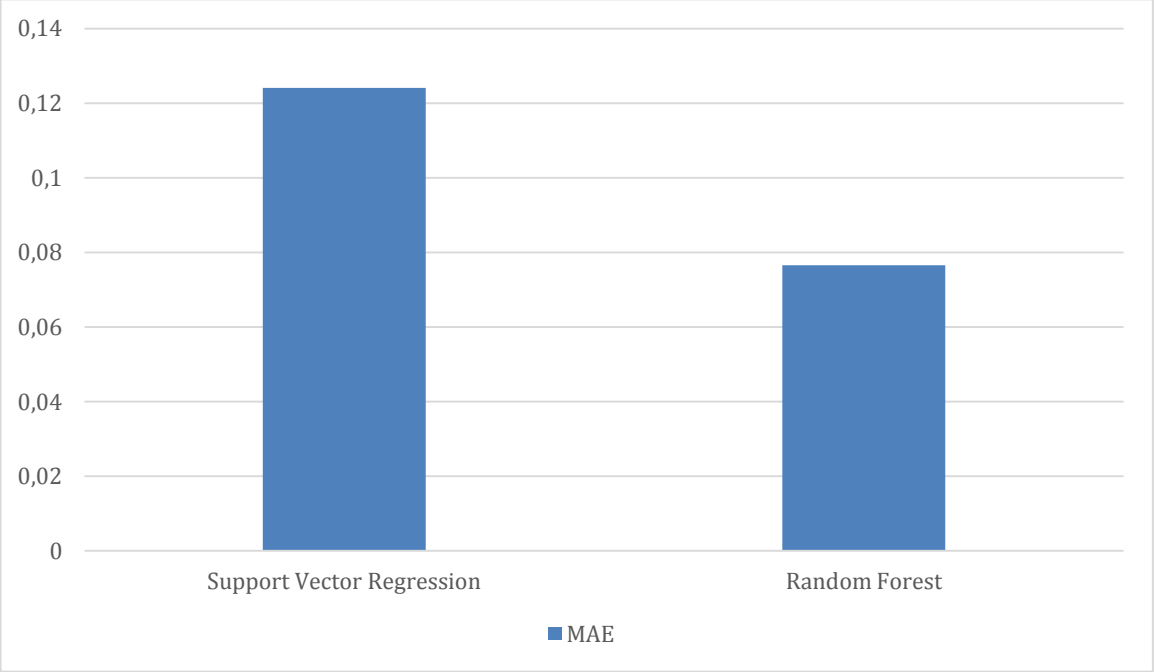

Рисунок 2.24 Часткове порівняння середньої абсолютної похибки

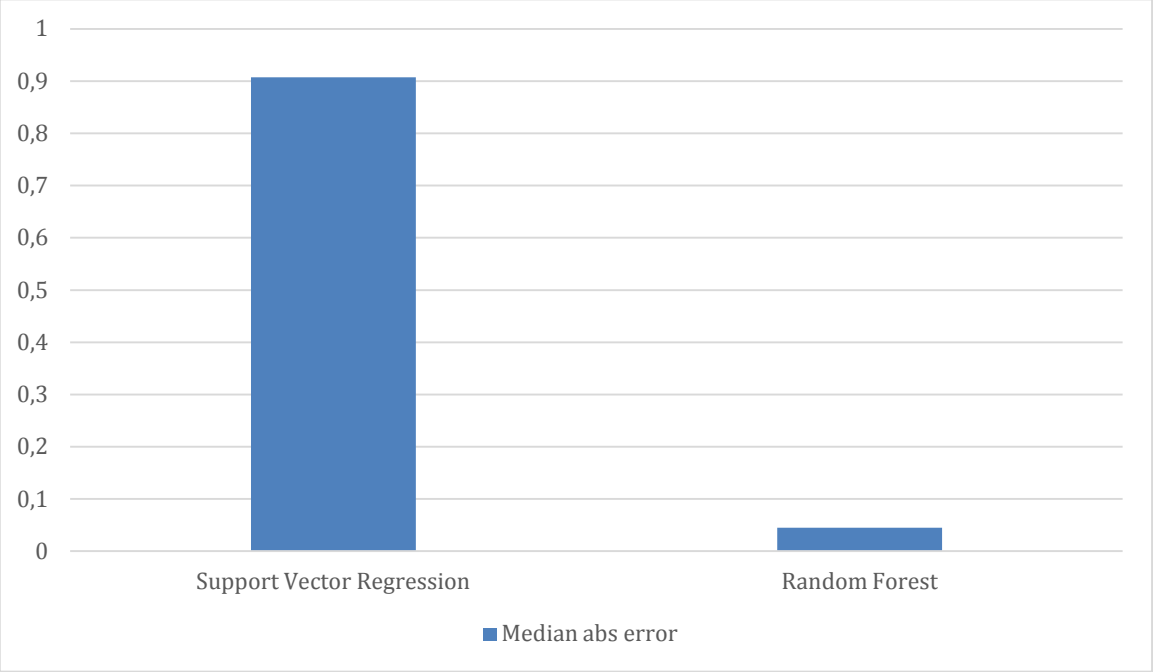

Рисунок 2.25 Часткове порівняння медіани абсолютної похибки

ВИСНОВКИ

Було проаналізовано чотири моделі для вирішення задачі прогнозування цін на майно. Теоретичний матеріал складається з досліджень формул, методів та алгоритмів роботи.

В практичній частині було досліджено моделі машинного навчання: Лінійна Регресія, Support Vector Regression, SGD Regression, Random Forest. Моделі було реалізовано за допомогою відкритих бібліотек та Python. Моделі було порівняно за метриками: середньоквадратична похибка, абсолютна похибка, медіана абсолютної похибки, коефіцієнт детермінації. В результаті аналізу можна зробити висновок, що алгоритм Random Forest є найбільш ефективним при прогнозуванні цін на житло.

Результати досліджень показали, що можливо доволі точно спрогнозувати дані з великою кількістю ознак. Але, в подальшому дослідженні варто більше звернути увагу на параметри та їх покращення. Особливо в моделях з поганими або середніми показниками.

Використовуючи результати даної роботи, можна прогнозувати кількісні дані з великою кількістю скорельованих ознак. Також, слід зазначити, що для практичного використання даної роботи варто розробити більш гнучку систему з автоматизованим та зручним інтерфейсом. Що буде доцільно розробити в подальшому дослідженні.

ПЕРЕЛІК ПОСИЛАНЬ

1. Machine learning https://www.ibm.com/cloud/learn/machine-learning (дата звернення: 15.05.2022)
2. Cohen, J., Cohen P., West, S.G., & Aiken, L.S. Applied multiple regression/correlation analysis for the behavioral sciences, 2003, 736c.
3. David A. Freedman Statistical Models: Theory and Practice, 2009, 458 c.
4. Hsu, Chih-Wei, Lin, Chih-Jen A Comparison of Methods for Multiclass Support Vector Machines, 2002, 26c.
5. Ho, Tin Kam Random Decision Forests, 1995, 40c.
6. Bottou, Léon Stochastic Learning 2004, 22c.
7. Joel, Grus Data Science from Scratch : First Principles with Python, 2019, 330c.
8. Berger, James O. Statistical decision theory and Bayesian Analysis, 1985, 634c.
9. Spall, J. C. Introduction to Stochastic Search and Optimization: Estimation, Simulation, and Control, 2003, 618c.
10. Rencher, Alvin C.; Christensen, William F. Methods of Multivariate Analysis, 2012, 800c.
11. Croxton, Frederick Emory; Cowden, Dudley Johnstone; Klein, Sidney Applied General Statistics, 1968, 754c.
12. van de Geer, Sara A New Approach to Least-Squares Estimation, with Applications, 1987, 40c.
13. Stuart J. Russell, Peter Norvig Artificial Intelligence: A Modern Approach, 2010, 1136c.
14. Polyak, Boris Introduction to Optimization, 1978, 438c.
15. Davies, Alex; Ghahramani, Zoubin The Random Forest Kernel and other kernels for big data from random partitions, 2014, 8c.

16. Financial crisis of 2007-2008 https://en.wikipedia.org/wiki/Financial_crisis_of_2007%E2%80%932008 (дата звернення: 13.05.2022)
17. Top Computer Languages https://statisticstimes.com/tech/top-computer-languages.php (дата звернення: 12.05.2022)
18. scikit-learn Machine Learning in Python https://scikit-learn.org (дата звернення: 20.05.2022)
19. Smith, Paul F.; Ganesh, Siva; Liu, Ping A comparison of random forest regression and multiple linear regression for prediction in neuroscience, 2013, 20c.
20. Bühlmann, Peter; van de Geer, Sara Statistics for High-Dimensional Data: Methods, Theory and Applications, 2011, 576c.