DOI 10.31891/2307-5732-2022-315-6-170-178
УДК 004.415

ПАШКЕВИЧ Олег
ЗВО Університет Короля Данила
https://orcid.org/0000-0001-7254-3512
e-mail: oleh.pashkevych@ukd.edu.ua
ВАЩИЩАК Сергій
ЗВО Університет Короля Данила
https://orcid.org/0000-0002-1753-1540
e-mail: serhii.vaschyshak@ukd.edu.ua
БОЙЧУК Андрій
ЗВО Університет Короля Данила
https://orcid.org/0000-0003-3357-8274
e-mail: andrii.m.boichuk@ukd.edu.ua
СТИСЛО Тарас
ЗВО Університет Короля Данила
https://orcid.org/0000-0002-2377-7985
e-mail:taras.styslo@ukd.edu.ua
ДЕМЧИНА Микола
ЗВО Університет Короля Данила
e-mail:mykola.demchyna@ukd.edu.ua

ЗАСТОСУВАННЯ МОДЕЛЕЙ МАШИННОГО НАВЧАННЯ ДЛЯ ПРОГНОЗУВАННЯ ЦІН НА РИНКУ НЕРУХОМОСТІ

В роботі наведено результати використання машинного навчання для прогнозування змін цін на ринку нерухомості. Проаналізовано економічні моделі та чинники, що впливають на формування ціни. Ці дані використано як основу для побудови моделі машинного навчання. Особлива увага зосереджена на виборі базових даних для побудови такої моделі. Проаналізовано різні типи регресійних моделей, які можуть бути реалізовані у відповідних програмних середовищах. В кінцевому результаті створена статистична модель для прогнозування ціни на житло за допомогою лінійної регресії. Вона використана для визначення тенденцій змін цін на ринку нерухомості в середньостроковому і довгостроковому періодах. Детально розглянуто всі бібліотеки, які використовуються в розробці статистичної моделі, проаналізовано їх переваги та недоліки. Розглянуто та реалізовано наступні кроки створення моделі: імпорт бібліотек та модулів, розроблено зчитування даних з датасету, аналіз даних, очистка та середня статистична оцінка. Як результат, лінійну регресію використано для аналізу масиву даних, отриманих з відкритого ресурсу – сайту продажу та оренди нерухомості Zillow. Об'єкти нерухомості. Описані на цьому ресурсі, стосуються міста Сіетл. Використано кластеризацію за основними параметрами об'єктів нерухомості, зокрема площею, розміщенням, віком. Щоб оцінити співвідношення різноманітних характеристик об'єкта нерухомості проведено 3D-моделювання з використанням конструктора Axes3D. На основі аналізу виконана візуалізація даних з використанням різних бібліотек. Аналізовані дані відображено на карті за допомогою Folium. Реалізовано кластеризацію даних та тестування на реальних даних, що показало доволі непогані результати прогнозу ціни. Отриманий результат моделювання перевірено на реальних об'єктах нерухомості та встановлено, що точність роботи такої моделі становить 76%. Такий високий результат вказує на правильність її побудови та раціональність використання програмних рішень для її втілення. Вона в майбутньому може бути використана для аналізу подібних масивів даних в цій галузі.

Ключові слова: машинне навчання, регресійні моделі, ціна нерухомості, економічні моделі, кластеризація даних.

PASHKEVYCH Oleh, VAHSCHYSHCHAK Serhii, BOICHUK Adrii,
STYSLO Taras, DEMCHYNA Mykola
HEI King Danylo University

APPLICATION OF MACHINE LEARNING MODELS FOR PREDICTING PRICES ON THE REAL ESTATE MARKET

The work shows the results of using machine learning to forecast price changes in the real estate market. Economic models and factors affecting price formation are analyzed. This data was used as a basis for building a machine learning model. Special attention is focused on the selection of basic data for building such a model. Various types of regression models, which can be implemented in appropriate software environments, have been analyzed. As a result, a statistical model was created for predicting housing prices using linear regression. It is used to determine trends in price changes on the real estate market in the medium and long term. All libraries used in the development of statistical models are examined in detail, their advantages and disadvantages are analyzed. The following steps of model creation were considered and implemented: import of libraries and modules, developed data reading from the dataset, data analysis, cleaning and average statistical evaluation. As a result, linear regression was used to analyze an array of data obtained from an open resource - the real estate sales and rental site Zillow. The real estate objects described on this resource relate to the city of Seattle. Clustering was used according to the main parameters of real estate objects, in particular, area, location, age. In order to evaluate the ratio of various characteristics of the real estate object, 3D modeling was carried out using the Axes3D designer. Based on the analysis, data visualization was performed using various libraries. The analyzed data is displayed on a map using Folium. Data clustering and testing on real data were implemented, which showed quite good price forecast results. The obtained modeling result was checked on real estate objects and it was established that the accuracy of the model is 76%. Such a high result indicates the correctness of its construction and the rationality of using software solutions for its implementation. In the future, it can be used to analyze similar data sets in this field.

Keywords: machine learning, regression models, real estate price, economic models, data clustering.

Постановка проблеми у загальному вигляді
та її зв'язок із важливими науковими чи практичними завданнями

Машинне навчання та основи штучного інтелекту використовують в багатьох сучасних сферах науки, технологій, бізнесу – всюди, де є можливість опрацьовувати масиви даних, які взаємодіють між собою за сталими алгоритмами. Найважливішою частиною процесу машинного навчання є правильна інтерпретація даних і використання їх у задачі яку потрібно вирішити.

Останнім часом для всіх є актуальними проблеми купівлі чи оренди житла, тому виникає проблема аналізу формування цін. Динаміка зміни цін на ринку нерухомості в умовах кризи, спричиненої військовими діями, кардинально відрізняється від динаміки зміни вартості в умовах економічного зростання, або економічної стабільності. Відомо, що основою ціноутворення на будь-якому ринку, у ринку нерухомості, є співвідношення мікроекономічних чинників – попиту і пропозиції, які є похідною від макроекономічних показників розвитку країни, регіону, міста. Всі ці параметри можна аналізувати в рамках певних моделей, що визначаються вхідними умовами, зокрема, регресійними. Регресійні моделі використовуються для передбачення цільових змінних в безперервній шкалі, що робить їх привабливими для вирішення багатьох питань в науці і для додатків в інформаційній галузі, такими як розуміння зв'язків між змінними, оцінювання тенденцій або створення прогнозів. Одним із прикладів може бути прогнозування за допомогою моделей продажів квартир компаніями в майбутні місяці.

Аналіз останніх джерел

В моделюванні динаміки ринку використовуються положення методології розрахунку ціни на об’єкт житлової нерухомості. В межах одного міста ціни на всі типи нерухомості зростають, зменшуються, або є у межах певного стабільного рівня. Причому подібні зміни відбуваються приблизно пропорційно. На це впливають локальні та глобальні чинники. Локальні призводять до того, що ціни на нерухомість відрізняються між собою. Ці чинники більше впливають на вартість, коли загальний рівень цін залишається приблизно постійним, але майже зникають на тлі сильного зростання чи зниження цін. Глобальні чинники пов’язані з макроекономічними параметрами, якими є рівень розвитку економіки та бізнесу в країні, регіоні, місті. Загальний рівень цін є відображенням впливу зміни макроекономічних чинників у часі і не залежить від локальних чинників. Тому загальний рівень цін є тільки функцією часу, що особливо актуально в умовах фінансово-економічної кризи [1].

Для прогнозування цін на ринку нерухомості застосовуються різноманітні підходи та моделі машинного навчання [2–6]. Але достовірність цих моделей залежить від впливу факторів, значна частина з яких спрощується, або не враховується через різнорідність завдань досліджень. Через, поки що, це немає єдиного концептуального підходу до прогнозування цін на житлові об’єкти в межах певного району чи міста.

Формулювання цілей статті

Метою роботи є розроблення концептуальної моделі прогнозування ціни об’єкта на ринку житлової нерухомості в межах одного міста для систематизації загальних положень щодо прогнозування та моделювання політики ціноутворення.

Виклад основного матеріалу

Аналіз даних у моделі здійснено за допомогою Pandas, Numpy, Sklearn з використанням візуалізації Matplotlib, Pylab, Folium та інших допоміжних бібліотек, кластеризацію даних – за допомогою методу KMeans. Опрацювання даних і програмний код реалізовано на мові Python 3 з використанням інструменту Jupyter notebook [7].

Для реалізації моделі спочатку було створено фігуру-каркас за допомогою %matplotlib.inline та задано необхідні параметри розмірів фігури. Через певну закритість статистики українського ринку нерухомості модель було побудовано згідно даних відкритого сайту Kaggle. Далі було зчитано дані (рис. 1), які взято з файлу kc_house_data.csv сайту Kaggle. Цей набір даних містить ціни продажу будинку для округу Кінг у Сіетлі. Сюди входять будинки, продані в період з травня 2017 року по травень 2018 року. Результати виведення даних наведено на рис. 2. Після цього дані були систематизовані і впорядковані. За допомогою методу describe() отримано основні статистичні оцінки даних (рис.3) по кожній числовій ознаці (типи int64 і float64): число непропущених значень, середнє, стандартне відхилення, діапазон, медіана і т.д. За допомогою методу corr() проведено кореляційний аналіз даних. Оскільки обчислений коефіцієнт кореляції більший за критичне його значення (0,6 > 0,3809), то з вірогідністю 0,95 можна стверджувати про статистично достовірну залежність між даними. Тому для нашої моделі ми застосували лінійну регресію [8], де взаємозв’язок між даними моделюється за допомогою лінійних функцій, а невідомі параметри моделі оцінюються за вхідними даними. Модель запускається з набором навчальних даних і продукує результат, який потім порівнюється з ціллю для кожного вхідного вектора в тренувальному наборі. На підставі результату порівняння та відповідно до обраного алгоритму навчання, параметри моделі коригуються. Далі, підібрана модель використовується при прогнозуванні відповідей для спостережень у другому наборі даних – даних для затверджування. Цей набір даних забезпечує об'єктивну оцінку моделі, яка відповідає навчальному набору при налаштуванні гіперпараметрів моделі.

```
In [2]: %matplotlib inline
        rcParams['figure.figsize'] = 5, 4
        sb.set_style('whitegrid')

In [3]: houses = pd.read_csv("kc_house_data.csv")
        houses.head()
```

Рис. 1. Зчитування даних з файлу

	id	date	price	bedrooms	bathrooms	sqft_living	sqft_lot	floors	waterfront	view	...	grade	sqft_above	sqft_basement	yr_
0	7129300520	20141013T000000	221900.0	3	1.00	1180	5650	1.0	0	0	...	7	1180	0	1
1	6414100192	20141209T000000	538000.0	3	2.25	2570	7242	2.0	0	0	...	7	2170	400	1
2	5631500400	20150225T000000	180000.0	2	1.00	770	10000	1.0	0	0	...	6	770	0	1
3	2487200875	20141209T000000	604000.0	4	3.00	1960	5000	1.0	0	0	...	7	1050	910	1
4	1954400510	20150218T000000	510000.0	3	2.00	1680	8080	1.0	0	0	...	8	1680	0	1

5 rows × 21 columns

id ідентифікатор date дата коли був проданий будинок price ціна(цільовий показник) bedrooms кількість спалень bathrooms кількість ван sqft_living житлова площа sqft_lot площа лоту floors кількість поверхів waterfront будинок з видом на набережну view перегляд condition наскільки хороші умови в цілому grade загальна оцінка sqft_above квадратні метри крім підвалу sqft_basement квадратний метр підвалу yr_built рік побудови yr_renovated рік реставрації zipcode код індекс lat координати широт long координати довготи sqft_living15 додаткова площа забудови sqft_lot15 додаткова площа лоту

Рис. 2. Результат виведення даних

```
houses.describe().T
```

	count	mean	std	min	25%	50%	75%	max
id	21613.0	4.580302e+09	2.876566e+09	1.000102e+06	2.123049e+09	3.904930e+09	7.308900e+09	9.900000e+09
price	21613.0	5.400881e+05	3.671272e+05	7.500000e+04	3.219500e+05	4.500000e+05	6.450000e+05	7.700000e+06
bedrooms	21613.0	3.370842e+00	9.300618e-01	0.000000e+00	3.000000e+00	3.000000e+00	4.000000e+00	3.300000e+01
bathrooms	21613.0	2.114757e+00	7.701632e-01	0.000000e+00	1.750000e+00	2.250000e+00	2.500000e+00	8.000000e+00
sqft_living	21613.0	2.079900e+03	9.184409e+02	2.900000e+02	1.427000e+03	1.910000e+03	2.550000e+03	1.354000e+04
sqft_lot	21613.0	1.510697e+04	4.142051e+04	5.200000e+02	5.040000e+03	7.618000e+03	1.068800e+04	1.651359e+06
floors	21613.0	1.494309e+00	5.399889e-01	1.000000e+00	1.000000e+00	1.500000e+00	2.000000e+00	3.500000e+00
waterfront	21613.0	7.541757e-03	8.651720e-02	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	1.000000e+00
view	21613.0	2.343034e-01	7.663176e-01	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	4.000000e+00
condition	21613.0	3.409430e+00	6.507430e-01	1.000000e+00	3.000000e+00	3.000000e+00	4.000000e+00	5.000000e+00
grade	21613.0	7.656873e+00	1.175459e+00	1.000000e+00	7.000000e+00	7.000000e+00	8.000000e+00	1.300000e+01
sqft_above	21613.0	1.788391e+03	8.280910e+02	2.900000e+02	1.190000e+03	1.560000e+03	2.210000e+03	9.410000e+03

Рис. 3. Виведення основних статистичних характеристик моделі

```
In [17]: train_data, test_data = train_test_split(houses, train_size =0.8, random_state = 3)
         reg = linear_model.LinearRegression()
         x_train = np.array(train_data['sqft_living']).reshape(-1,1)
         y_train = np.array(train_data['price']).reshape(-1, 1)
         reg.fit(x_train, y_train)
         #evaluate simple model
         x_test = np.array(test_data['sqft_living']).reshape(-1, 1)
         y_test = np.array(test_data['price']).reshape(-1, 1)
         pred = reg.predict(x_test)
         print('Simple Model')


         print('R-squared (training) ', round(reg.score(x_train, y_train), 3))
         print('R-squared (testing) ', round(reg.score(x_test, y_test), 3))
         print('Intercept: ', reg.intercept_)
         print('Coefficient:', reg.coef_)

Simple Model
R-squared (training)  0.492
R-squared (testing)  0.496
Intercept:  [-47235.8113029]
Coefficient: [[282.2468152]]

In [18]: _, ax = plt.subplots(figsize= (10, 12))
         plt.scatter(x_test, y_test, color= 'darkgreen', label = 'data')
         plt.plot(x_test, reg.predict(x_test), color='red', label= ' Predicted Regression line')
         plt.xlabel('Living Space (sqft)')
         plt.ylabel('price')
         plt.legend()
         plt.gca().spines['right'].set_visible(False)
         plt.gca().spines['right'].set_visible(False)
```

Рис. 4. Код створення моделі

Набори даних для затверджування були використані для регуляризації шляхом ранньої зупинки: навчання переривалося, коли помилка на наборі даних для затверджування збільшувалась, оскільки це є ознакою перенавчання на навчальному наборі даних. Ця процедура ускладнювалась тим, що помилка набору даних для затверджування могла коливатися під час тренування, утворюючи декілька локальних мінімумів. Таке ускладнення призвело до появи багатьох ad-hoc правил для визначення того, що перенавчання справді почалось (рис. 4).

За допомогою метода fit() модель навчається, викликається метод predict() – результат прогнозування з використанням лінійної регресії (рис. 5).

Рис. 5. Результат прогнозування з використанням лінійної регресії

```
n [25]: #split in X,y
y=houses.loc[:,"price"].apply(lambda x: log(x))

X=houses.loc[:,["sqft_living","grade","floors","lat","long","bathrooms","bedrooms","yr_built","yr_renovated","view"
X["sqft_living"]= X["sqft_living"].apply(lambda x: log(x))
X["lat"]= X["lat"].apply(lambda x: abs(47.63-x))
X["long"]= X["long"].apply(lambda x: log(abs(x)))
X["yr_built"]= X["yr_built"].apply(lambda x: log(abs(x-1955)+1))
X["yr_renovated"]= X["yr_renovated"].apply(lambda x: log(x+1))
X["sqft_lot"]= X["sqft_lot"].apply(lambda x: log(x))

X["lat*long"]=X["lat"]*X["long"]
X["sqft_living*sqft_lot"]=X["sqft_living"]*X["sqft_lot"]

# split in train, test
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.40, random_state=123)
#split in validation, test
X_val, X_test, y_val, y_test = train_test_split(X_test, y_test, test_size=0.5, random_state=123)
```

Рис. 6. Код вдосконалення моделі

```
In [26]: # Validation Set
regr= linear_model.LinearRegression()
regr.fit(X_train,y_train)
y_pred= regr.predict(X_val)

print("Validation Set")
print("Root Mean Square Error: %6.f" % sqrt(mean_squared_error(np.exp(y_val.values),np.exp(y_pred))))
print('R2 score: %.6f' % r2_score(np.exp(y_val.values),np.exp(y_pred)))

Validation Set
Root Mean Square Error: 142748
R2 score: 0.841722

In [27]: #Test set
y_pred= regr.predict(X_test)
print("Test Set")
print("Root Mean Square Error (squared): %2.f" % sqrt(mean_squared_error(np.exp(y_test.values),np.exp(y_pred))))
print('R2 score: %.2f' % r2_score(np.exp(y_test.values),np.exp(y_pred)))

Test Set
Root Mean Square Error (squared): 154262
R2 score: 0.83
```

Рис. 7. Результати моделювання з навчальними і тестовими даними

Але, як видно з рис. 5, результат прогнозування вийшов доволі грубим, бо розподіл даних є далеким від лінійного закону, тому було вирішено удосконалити модель з використанням додаткових ознак, що є в даних. Ці ознаки, наприклад розміщення об’єктів на карті, координати широта та довготи, житлова площа, поштовий індекс мали допомогти підвищити якісь результати моделювання (рис. 6 та 7). На останньому етапі для забезпечення об'єктивної оцінки кінцевої моделі, яка відповідає навчальному набору даних

використовувався тестовий набір даних. Результат на наборі для перевірки та тестових даних набагато піднявся до кращої оцінки (рис. 8).

```
In [28]: # Predicting real life data (Training)
         #split in X,y
         y=houses.loc[:,"price"].apply(lambda x: log(x))

         X=houses.loc[:,["sqft_living","sqft_lot","yr_built","bedrooms","bathrooms","lat","long","waterfront"]]
         X["sqft_living"]= X["sqft_living"].apply(lambda x: log(x))
         X["sqft_lot"]= X["sqft_lot"].apply(lambda x: log(x))
         X["lat"]= X["lat"].apply(lambda x: abs(47.63-x))
         X["long"]= X["long"].apply(lambda x: abs(x))
         X["yr_built"]= X["yr_built"].apply(lambda x: log(x))


         X["lat*long"]=X["lat"]*X["long"]
         X["sqft_living*sqft_lot"]=X["sqft_living"]*X["sqft_lot"]

         # split in train, test
         X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.30, random_state=123)

         regr_test= linear_model.LinearRegression()
         regr_test.fit(X_train,y_train)
         y_pred= regr_test.predict(X_test)
         p = r2_score(np.exp(y_test.values),np.exp(y_pred))
         print(p)
         print("Root Mean Square Error: %2.f" % sqrt(mean_squared_error(np.exp(y_test.values),np.exp(y_pred))))
         print('R2 score: %.6f' % p)

0.7622146910961412
Root Mean Square Error: 179697
```

Рис. 8. Результати моделювання набору для перевірки з тестовими даними

Результат, показаний на навчальних даних (підмножині всього набору даних) виявився навіть кращим, за попередній. Але важливо відзначити, що результат на тестових даних є гіршим. В нашому випадку відмінність між вірністю на навчальних і на тестових даних невелика. Але, якщо модель складна, то цілком можливо на навчальних даних отримати стовідсоткову вірність, а на тестових – не більшу, ніж при випадковому вгадуванні.

Суттєве покращення ситуації дає застосування методу перехресної перевірки. Одна з його форм називається перехресною перевіркою з виключенням по одному. В нашій моделі вибирається який-небудь приклад з наявних даних, навчається модель на всіх даних, крім цього прикладу, а потім перевіряється, чи правильно модель класифікує цей приклад. Такий процес повторюється для всіх елементів набору даних. У цьому циклі тестується послідовність моделей на всіх прикладах, а по його завершенні друкується середній результат. При використанні перехресної перевірки зацикленість не виникає, тому що кожна модель тестується на прикладі, який вона не бачила при навчанні. Отже, прогнозуючу таким чином оцінку можна вважати надійним показником загальної моделі на нові дані.

Основна проблема перехресної перевірки з вилученням по одному полягає в тому, що доводиться навчати по суті нову модель для кожного прикладу, і з ростом набору даних суттєво зростає складність процесу моделювання. Зменшити складність виключення по одному дозволяє x-прохідна перехресна перевірка, де x – невелике ціле число. В нашому випадку дані точкової прохідної перевірки розбиваються на 5 груп. Потім навчаються п’ять моделей, при цьому кожен раз виділяється з навчального набору одна група. Добутий код резервується відразу 20% даних, а не одним елементом. Всі навчені моделі тестуються на даних, що входять у виключену групу, а результати усереднюються. Після виконання тренування нашої моделі можна поглянути на значення кожного параметра. Наприклад, P0, так званий зсув або перетин, можна переглянути за допомогою атрибута intercept().

Основною перевагою лінійної регресії є її інтерпретованість (рис. 9) в значній мірі тому, що модельні коефіцієнти є хорошими показниками.

```
actual_values = y_test
predictions = y_pred
sb.scatterplot(actual_values, predictions)

<matplotlib.axes._subplots.AxesSubplot at 0x7fa25e70d9e8>
```

Рис. 9. Код для відображення результатів моделювання

На рис. 10 наведено найкращий результат прогнозування з використанням удосконаленої моделі.

Рис. 10. Результат прогнозування з використанням удосконаленої моделі

Для покращення візуалізації залежності даних ще раз застосовуємо лінійну регресію (рис. 11, 12).

```
In [30]: y_train_pred = regr_test.predict(X_train)
         y_test_pred = regr_test.predict(X_test)

In [31]: # Plot residuals
         plt.scatter(y_train_pred, y_train_pred - y_train, c = "blue", marker = "s", label = "Training data")
         plt.scatter(y_test_pred, y_test_pred - y_test, c = "lightgreen", marker = "s", label = "Validation data")
         plt.title("Linear regression")
         plt.xlabel("Predicted values")
         plt.ylabel("Residuals")
         plt.legend(loc = "upper left")
         plt.hlines(y = 0, xmin = 11.5, xmax = 15.5, color = "red")
         plt.show()
```

Рис. 11. Код програми для повторного застосування лінійної регресії

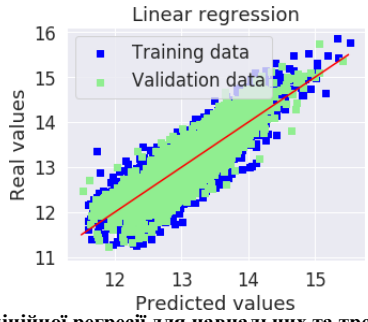

Рис. 12. Графік лінійної регресії для навчальних та тренувальних даних

Після цього здійснено сортування отриманих даних до максимальних, мінімальних, середніх значень для кожного атрибута (параметра) (рис. 13 та 14).

```
In [33]: index=["min","max","mean","coef"]
         columns=list(X_test)
         test = pd.DataFrame(index=index, columns=columns)
         test.loc["min"]=list(X_test.min())
         test.loc["max"]=list(X_test.max())
         test.loc["mean"]=list(X_test.mean())
         test.loc["coef"]=regr_test.coef_
```

Рис. 13. Код програми для виводу середньої статистики

```
In [34]: test
```

Out[34]:

	sqft_living	sqft_lot	yr_built	bedrooms	bathrooms	lat	long	waterfront	lat*long	sqft_living*sqft_lot
min	6.04025	6.39693	7.54961	0	0	0.0001	121.676	0	0.0121911	44.3775
max	9.19928	13.9681	7.60837	10	7.75	0.4524	122.515	1	55.1978	113.265
mean	7.55232	8.99356	7.58617	3.37276	2.11636	0.122584	122.215	0.0089451	14.9804	68.0463
coef	0.636933	0.00800451	-0.348176	-0.0477763	0.0823561	682.224	1.05851	0.728813	-5.60684	0.00903068

Рис. 14. Вивід результату статистичного аналізу

Для того, щоб оцінити співвідношення різноманітних характеристик об’єкта нерухомості (наприклад, житлової площі, житлового лоту і спалень з ваннами) проведено 3D-моделювання з використанням конструктора Axes3D (рис. 15, 16).

```
In [20]: fig = plt.figure(figsize=(19,12.5))
         ax=Axes3D(fig)
         ax.scatter(houses['sqft_living'],houses['sqft_lot'],houses['bathrooms'],c="darkgreen",alpha=.5)
         ax.set(xlabel='\n sqft Living',ylabel='\nsqft Lot',zlabel='\nBathrooms / Bedrooms')
         ax.set(ylim=[0,250000])

Out[20]: [(0, 250000)]
```

Рис. 15. Код програми побудови 3D графіка

Сегментація 3D-моделі – непросте завдання. Навіть якщо сегментація задана попередньо (рис. 16), складно з точністю визначити місця стикування, симетрію і паралелізм в деталях зображення. Тому було створено 3D-графік, щоб визначити співвідношення між видом, загальною оцінкою та роком побудови об’єкта нерухомості (рис. 17, 18). З графіка видно, що нові будинки мають кращі оцінки, але все одно виявити точні дані візуальним спостереженням досить важко.

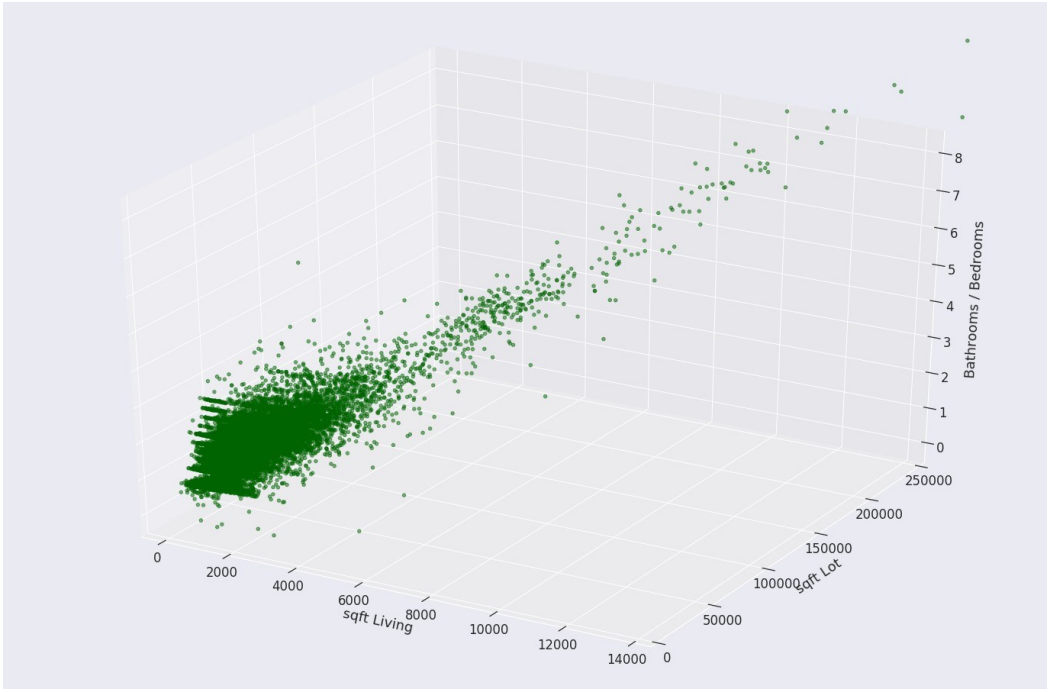

Рис. 16. 3D графік співвідношення характеристик об'єкта нерухомості

```
In [22]: fig=plt.figure(figsize=(9.5,6.25))
         ax=Axes3D(fig)
         ax.scatter(train_data['view'],train_data['grade'],train_data['yr_built'],c="darkgreen",alpha=.5)
         ax.set(xlabel='\nView',ylabel='\nGrade',zlabel='\nYear Built');
```

Рис. 17. Код програми побудови 3D графіка

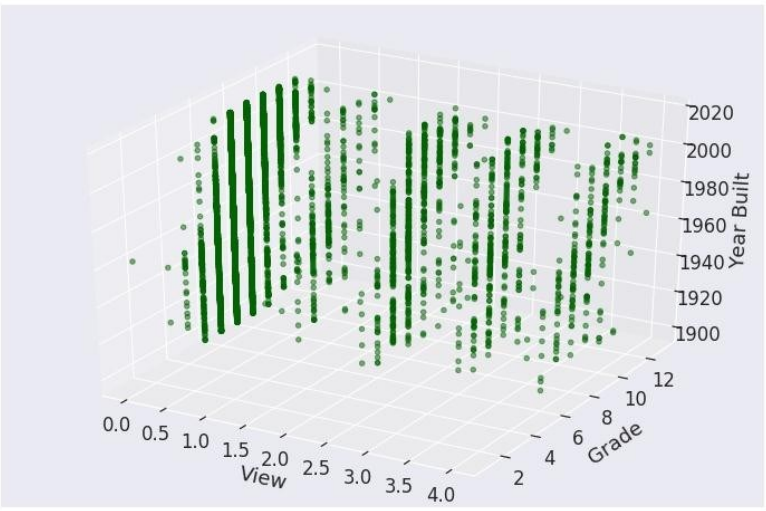

Рис. 18. 3D візуалізація взаємозв'язків характеристик об'єкта нерухомості

Для зменшення наслідків незначних помилок спостереження застосовано метод зв'язування даних (data binning). В нашій моделі використовуються yr_built та yr_renovated. До даних додано вік та епоху ремонту будинків, коли вони продавались. Також розділено ці стовпці на інтервали, і можна це спостерігати на наведених нижче гістограмах (рис. 19). Після цього створено веб-карту з використанням бібліотеки для візуалізації географічних даних та інформації Folium, яка містить координати та місцеположення (рис. 20).

Рис. 19. Візуалізація зв'язування даних

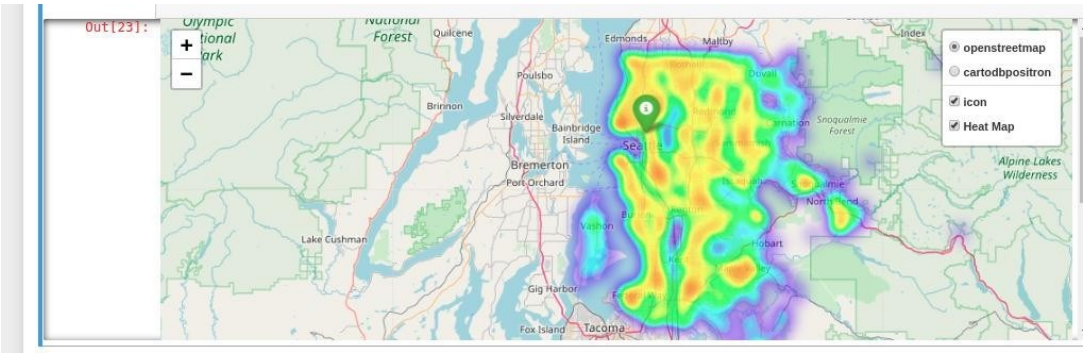

Рис. 20. Зображення на карті

Останнім етапом проектування моделі було проведення кластеризації даних для знаходження груп схожих об'єктів у вибірці за допомогою методу k-середніх. Місто Сіетл має 7 районів. Тому партії об'єктів нерухомості поділяються на сім кластерів за координатами. Кластери розділяються за місцем розташування і поділяються за ціною та відстанню від центру. Також Сіетл має 127 мікрорайонів. Отже, координати розділяються на 127 кластерів, тому що кластеризація по мікрорайонах може дати вищу точність у прогнозуванні ціни, ніж кластеризація по районах (рис. 21).

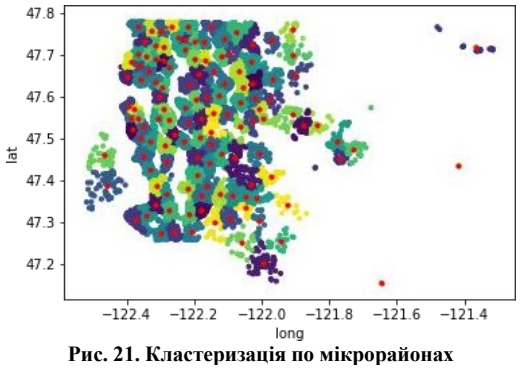

Рис. 21. Кластеризація по мікрорайонах

Модель було протестовано на десяти об'єктах Zillow – сайту, що дозволяє отримати детальну інформацію про об'єкт нерухомості, оренду та викупи, які не можна знайти в MLS. Результати тестування дали 76% збіжності, що свідчить про адекватність моделі прогнозування.

Висновки з даного дослідження і перспективи подальших розвідок у даному напрямі

На основі аналізу даних була розроблена концептуальна модель прогнозування ціни об'єкта на ринку житлової нерухомості, яка складається з п'яти основних блоків, а саме: вхідні дані, оцінка впливу факторів, блок моделювання, синтез моделей та, безпосередньо, прогнозування ціни. Побудова концептуальної моделі дозволяє систематизувати загальні положення щодо прогнозування та моделювання політики ціноутворення та дає змогу не тільки спрогнозувати середній показник рівня ціни для наступного періоду, але й вводити в майбутньому в модель принципово нові корегуючі коефіцієнти для підвищення достовірності прогнозування.

Література

1. Прогнозування цін на нерухомість в умовах фінансово-економічної кризи. Сейл Прайс Компані [Електронний ресурс]. – Режим доступу : http://www.saleprice.com.ua/ua/publications/real_estate_price_forecasting.html (дата звернення: 03.10.2022 р.).

2. Мірошниченко І.В., Крашеніннікова О.В. Прогнозування ціни на нерухомість з використанням алгоритмів машинного навчання. Ефективна економіка. 2022. № 1. URL: http://www.economy.nayka.com.ua/?op=1&z=9890 (дата звернення: 05.11.2022). DOI: 10.32702/2307-2105-2022.1.81

3. Григорків В. С., Ярошенко О. І., Філіпчук Н. В. Нейронні мережі та їх використання для прогнозування тенденцій ринку нерухомості. Науковий вісник НЛТУ України. 2012. Вип. 22.5. С. 324-330.

4. Шаповалова В.О. Прогнозування цін на нерухомість з урахуванням тенденцій ділових циклів. Економіка та держава. 2015. № 9/2015. С. 37–41.

5. Калиніченко Ю.В., Добровольська В.В., Абрамчк А.Р. Моделювання тенденцій ринку нерухомості. Економічний аналіз : зб. наук. праць Тернопільського національного економічного університету, 2016. Том 23. № 1. С. 28-36.

6. Хлевна Ю.Л., Бура Ю.С. Інформаційне забезпечення прогнозування цін на нерухомість методами

машинного навчання. Sciences of Europe. 2021. № 71. P. 54–62.

7. Элбон К. Машинное обучение с использованием Python. Петербург : СПб БХВ, 2019. 890 с.

8. Лінійна регресія. Sandro Skansi, Introduction to Deep Lerning. Springer. 2018 [Електронний ресурс]. – Режим доступу : https://uk wikipedia.org/.

References

1. Prohnozuvannia tsin na nerukhomist v umovakh finansovo-ekonomichnoi kryzy. Seil Prais Kompani [Elektronnyi resurs]. – Rezhym dostupu : http://www.saleprice.com.ua/ua/publications/real_estate_price_forecasting.html (data zvernennia: 03.10.2022r.).

2. Miroshnychenko I.V., Krasheninnikova O.V. Prohnozuvannia tsiny na nerukhomist z vykorystanniam alhorytmiv mashynnoho navchannia. Efektyvna ekonomika. 2022. № 1. URL: http://www.economy.nayka.com.ua/?op=1&z=9890 (data zvernennia: 05.11.2022). DOI: 10.32702/2307-2105-2022.1.81

3. Hryhorkiv V. S., Yaroshenko O. I., Filipchuk N. V. Neironni merezhi ta yikh vykorystannia dlia prohnozuvannia tendentsii rynku nerukhomosti. Naukovyi visnyk NLTU Ukrainy. 2012. Vyp. 22.5. S. 324-330.

4. Shapovalova V.O. Prohnozuvannia tsin na nerukhomist z urakhuvanniam tendentsii dilovykh tsykliv. Ekonomika ta derzhava. 2015. № 9/2015. S. 37–41.

5. Kalynichenko Yu.V., Dobrovolska V.V., Abramchk A.R. Modeliuvannia tendentsii rynku nerukhomosti. Ekonomichnyi analiz : zb. nauk. prats Ternopilskoho natsionalnoho ekonomichnoho universytetu, 2016. Tom 23. № 1. S. 28-36.

6. Khlevna Yu.L., Bura Yu.S. Informatsiine zabezpechennia prohnozuvannia tsin na nerukhomist metodamy mashynnoho navchannia. Sciences of Europe. 2021. № 71. R. 54–62.

7. Elbon K. Mashynnoe obuchenie s ispolzovaniem Python. Peterburh : SPb BKhV, 2019. 890 s.

8. Liniina rehresiia. Sandro Skansi, Introduction to Deep Lerning. Springer. 2018 [Elektronnyi resurs]. – Rezhym dostupu : https://uk wikipedia.org/.