**Індивідуальне домашнє завдання**

**З предмету «Інтелектуальний аналіз даних»**

**Студента групи ЕК-41**

**Боброва Владислава**

**2. Первинний аналіз даних (Data Unde**[**r**](https://dist.karazin.ua/moodle/mod/glossary/showentry.php?eid=80607&displayformat=dictionary)**standing)**

***2.1. Збір даних (Data collection)***

Для даного проекту ми беремо дані з сайту kaggle.com, а саме датасет про квартири у Мадриді(Іспанія). Ці дані були зібрані іншими людьми, вони безкоштовні і для проекту нам їх вистачить.

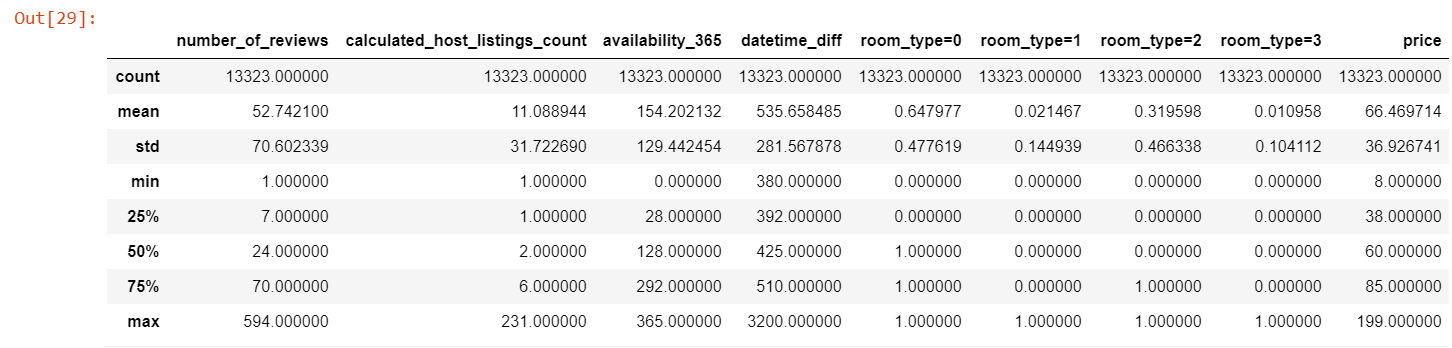
**Джерело:** <https://www.kaggle.com/rusiano/madrid-airbnb-data?select=listings_detailed.csv>

***2.2. Опис даних (Data desc***[***r***](https://dist.karazin.ua/moodle/mod/glossary/showentry.php?eid=80607&displayformat=dictionary)***iption):***

Хоча у проекту декілька файлів, ми беремо лише один – детальний опис квартир без доповнених даних(по типу відгуків). У таблиці вказані основні характеристики:

|  |  |
| --- | --- |
| Таблиця | Ми беремо лише одну, лише listings\_detailed.csv. Усі інші файли не розбираємо. Вага файлу – 77 мб, але назараз це архів вагою 19 мб. Коротка версія, з якою ми працюємо – 2 мб. |
| Ключ | Він тут один – це id типу цілий, у кожного житла він різний, але вони не йдуть поспіль – тому ідентифікатор – це тільки ім’я. |
| Кількість рядків | На даний момент в нас трохи більше 20к квартир у таблиці. Планується, що більшість з цих даних ми відкинемо чи змінемо, тому що потрібна дуже велика потужність |
| Кількість стовпців | На даний момент в нас 106 колонок. З них більшість – це цифрові дані про те, що є у квартирі(ванни, тощо), але також є багато текстових змінних(опис, ім’я господаря), а деякі змінні дублюють одна одну(наприклад, повна назва квартири та идентифікатор – вони виконують одну й ту саму функцию). Але для аналізу ми візьмемо короткий сет з 16 колонками. |
| Семплювання даних | На даний момент не зафіксовано проблеми, але, якщо потужності не вистачить, то ми можемо розбити сет на дві рівні частини по 10к об’єктів. |

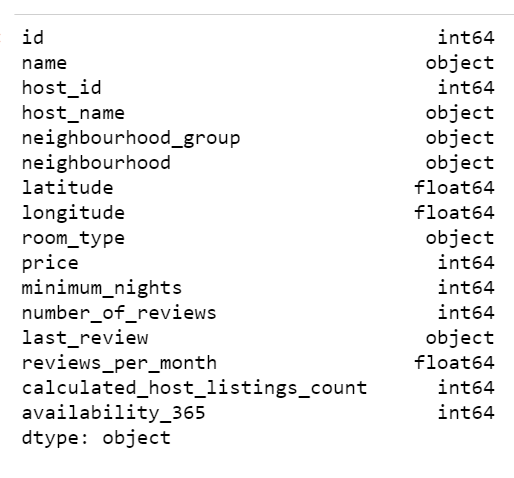
Щодо ключових статистик, то вони тут вже розраховані для тих типових регрессорів, що я планую використовувати для моделі:



**Рис. 1 Ключові статистики для регрессорів датасету до генерації даних**

Бачимо, що тут є виброси та неактуальні дані, з якими ми розберемося надалі.

***2.3. Дослідження даних (Data explo***[***r***](https://dist.karazin.ua/moodle/mod/glossary/showentry.php?eid=80607&displayformat=dictionary)***ation)***



***Рис. 2. Типи даних у датасету***

Якщо детально проаналізувати ці дані, то зрозуміло, що деякі стовпчики не дадуть нам ніякої інформації – наприклад, id це ключ, або ім’я господаря(name). На мій погляд, тут в нас є декілька груп змінних:

а) Числові дані, які допоможуть нам у створенні моделі:

`min\_nights`, `num\_of\_rew`, `availability’ тощо.

б) Зміна типу об’єкт, яка є категоріальною та впливає на нашу модель:

`room\_type`

в) Текстові та числові зміні, які ми не будемо використовувати у моделі: `id`, `name`, `host\_id`, `host\_name’, `neighbourhood` тощо.

г) Зміна типу об’єкт, що трансформується у числову:

`last\_rewiew`

д) Зміна, яку ми будемо пояснювати:

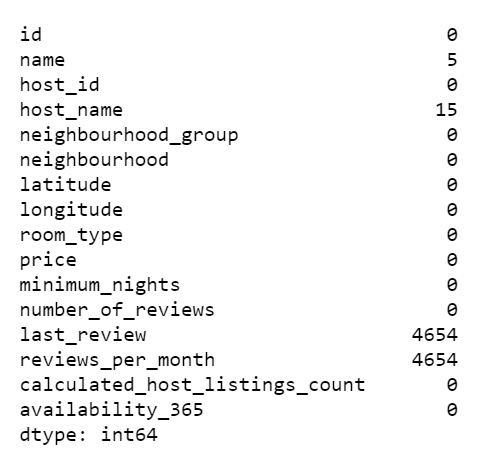
`price`.

Усі зміни та оцінку цих даних проведемо далі.

***2.4. Якість даних (Data quality)***

Перш за все, помітимо, що в нас немає проблем з неправильним зв’язком або різним кодуванням(наприклад, у `room\_type` дійсно є чотири види, що названі одним і тим же словом).

Перейдемо до оцінки пропусків:



Бачимо, що, окрім двох змінних, пропусків майже немає. Щодо вибросів, то вони існують(це можна побачити по максимумах на Рис. 1. Хоча все ж таки проблема у тому, що деякі об’екти коштують дуже багато через параметри, які не враховує наша модель), тому з цими значеннями ми також будемо боротися.

**3. Підготовка даних (Data P**[**r**](https://dist.karazin.ua/moodle/mod/glossary/showentry.php?eid=80607&displayformat=dictionary)**epa**[**r**](https://dist.karazin.ua/moodle/mod/glossary/showentry.php?eid=80607&displayformat=dictionary)**ation)**

***3.1. Відбір даних (Data Selection)***

Як я вже зазначав у пункті 2.3, ми відбираємо конкретні стовпчики, тому що:

1. Це числові дані, що можна використати в моделі;

2. Категоріальна зміна, що впливає на зміну, що пояснюється;

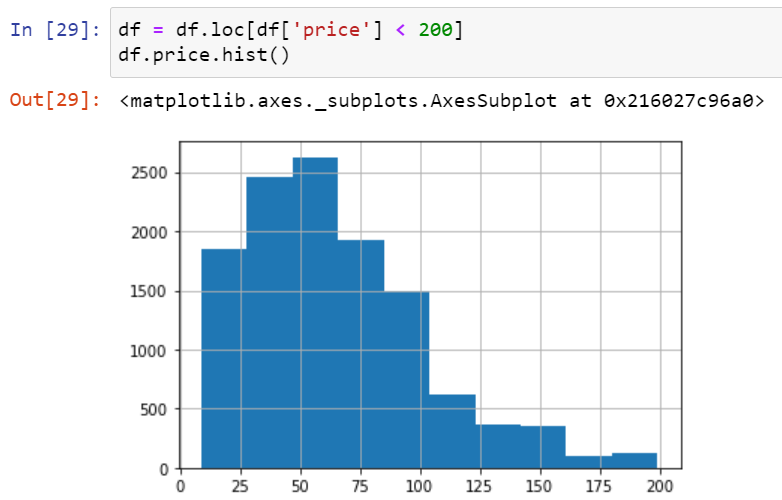
3. Зміна типу час, що також впливає на ціну.

Після видалення непотрібних стовпчиків, в нас залишається 9 змінних, що пояснюють(поки що я залишив `latitude` та `longtitude` - можливо, географічні показники теж впливають – для регресії це буде цікаво.

***3.2. Очищення даних (Data Cleaning)***

Щодо очищення даних, то було прийтяте рішення видалити пусті значення – в нас залишиться приблизно 7,5к значень, яких достатньо для нашої задачі та для нашої техніки.

Щодо викідів, то вони в нас будуть видалені за допомогою loc() – у наших графіках розподілення ці викиди можна було побачити. Після залочення надвеликих значеннь на прикладі нашої ціни можна побачити, що дані більш однорідні:



**Рис. 4 Ціна після залочення надвеликих значеннь**

Таким чином ми виключаємо деяку кількість рядків, щоб отримати більш однорідний датасет без пропусків та викідів.

***3.3. Генерація даних (Const***[***r***](https://dist.karazin.ua/moodle/mod/glossary/showentry.php?eid=80607&displayformat=dictionary)***ucting new data)***

***Зараз ми згенеруємо нові ознаки, а саме:***

а) `room\_type` за допомогою бінарного кодування перетворюємо у чотири зміні, кожна з яких показує, до якого типу ця комната відноситься .

б) за допомогою бібліотеки Datetime ми з нашого `last\_rewiew’ ми отримаємо числову ознаку – скільки днів пройшло до сьогодні. За допомогою перетворення ми отримаємо нову колонку числового формату.

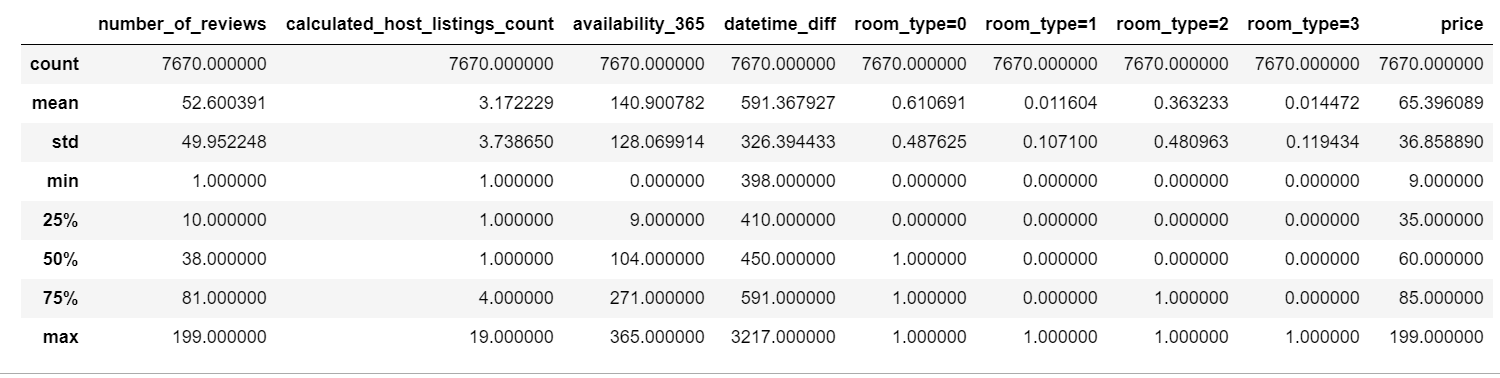
Шкалювання даних ми будемо проводити вже перед створенням моделей.

***3.4. Інтеграція даних (Integ***[***r***](https://dist.karazin.ua/moodle/mod/glossary/showentry.php?eid=80607&displayformat=dictionary)***ating data)***

У нас вся інформація знаходилась у одному сеті, тому інтегрувати нічого не потрібно.

***3.5. Форматування даних (Fo***[***r***](https://dist.karazin.ua/moodle/mod/glossary/showentry.php?eid=80607&displayformat=dictionary)***matting Data)***

Оскільки ми вже переробили та створили нові ознаки з текстових зміних, то в нас усі змінні – цілі числа, тому з форматом нам вже нічого змінювати не потрібно.



**Рис 5. Описання даних повністю підготовленого датасету**

**4. Моделювання (Modeling)**

Для цього датасету нам буде необхідно вирішити задачі регрессії, класифікації та кластерізації. Для комфортного запису кожна з задач буде записана окремо.

**Регресія**

***4.1. Вибір алгоритмів (Selecting the modeling technique)***

Оскільки маємо підготовлені дані, то у нас нема обмежень для регресійної моделі, то ми застосуємо такі алгоритми:

а) Однофакторна регресія;

б) Багатофакторна регресія;

в) Поліномінальна регресія;

г) Регресія з виключенням найгіршого;

д) Дерево рішень;

е) Випадковий ліс.

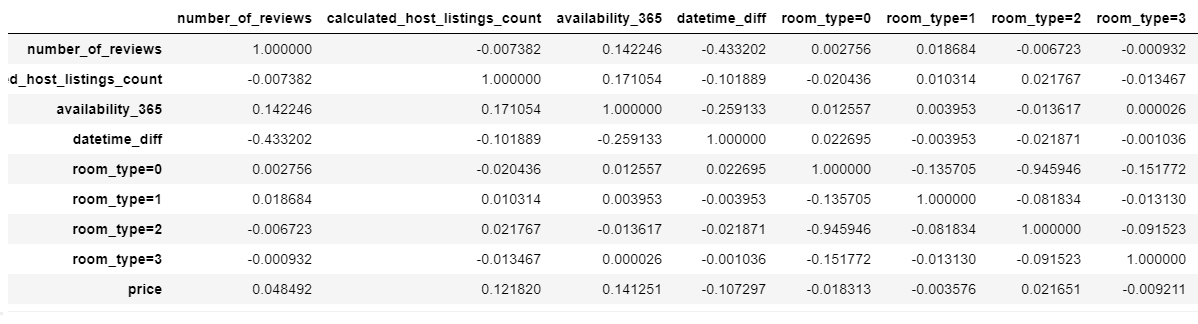
***4.2. Планування тестування (Gene***[***r***](https://dist.karazin.ua/moodle/mod/glossary/showentry.php?eid=80607&displayformat=dictionary)***ating a test design)***

Для усіх задач ми будемо використовувати ділення 80/20, де 80 це тренувальна вибірка, а 20 – тестова.

***4.3. Навчання моделей (Building the models)***

а) Однофакторна регресія;

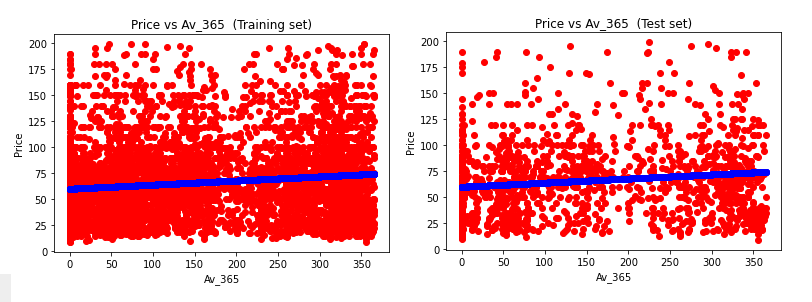
На основі нашої кореляційної матриці найбільший за модулем коефіцієнт залежності має змінна availability\_365. Тому саме від неї ми будуємо однофакторну регресійну модель.



**Рис 6. Кореляційна матриця**

Треба зазначити, що усі коефіцієнти дуже низькі, тому якість моделей теж буде низькою, але, для порівняння та наукового інтересу будуємо найбільш вірогідним шляхом.

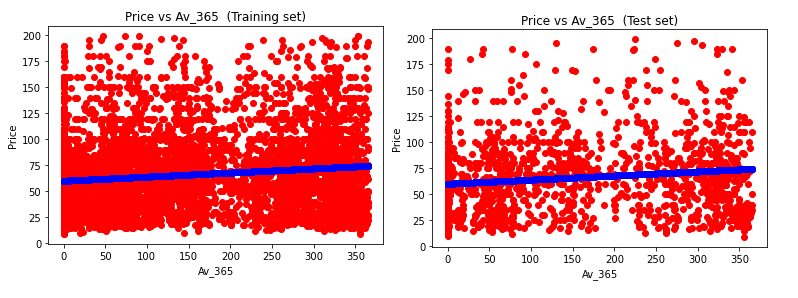
Числові параметри моделі будуть оцінені у результатах, але за графіками бачимо, що модель майже не описує зміну, яка нас цікавить.



**Рис 7. Графік однофакторної регресії**

б) Багатофакторна регресія;

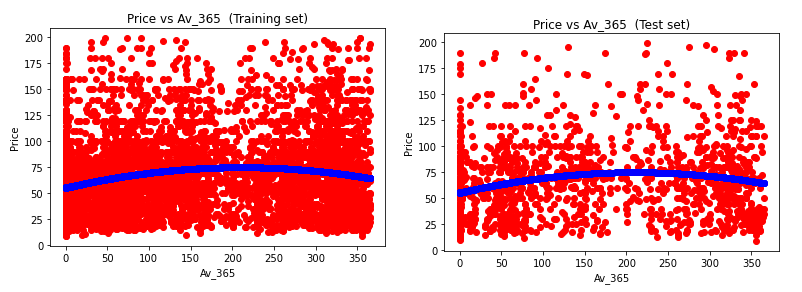
Для багатофакторної регресії використаємо всі змінні. Також бачимо, що графік нам не дає ніякої інформації.



**Рис 8. Графік багатофакторної регресії**

в) Поліномінальна регресія;

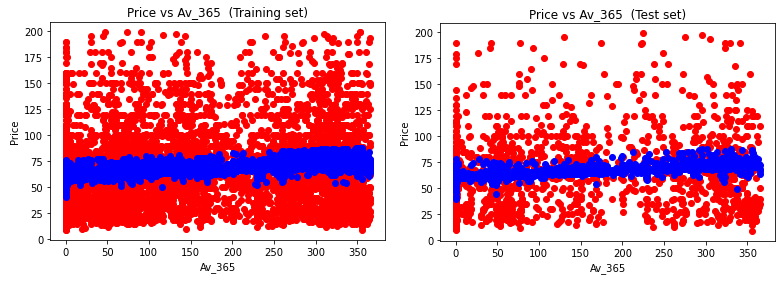
Для поліномінальної регресії ми також взяли змінну availability\_365 та також отримали приблизно такі ж самі графіки(порівняти цифрові значення зможемо у пункті 4.4):



**Рис. 9 Графік поліномінальної регрессії**

г) Регресія з виключенням найгіршого;

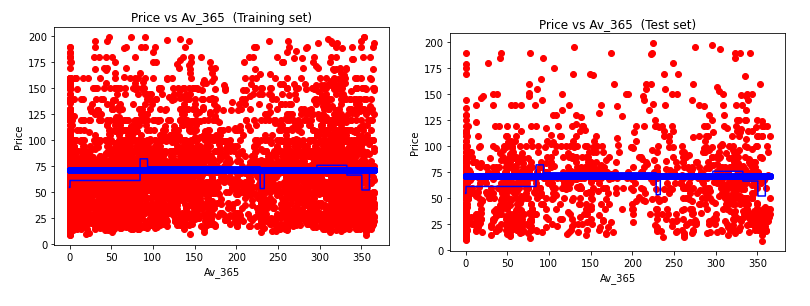
За допомогою імпортованих класів видалили з моделі змінні, що найгірше описують. До речі, графік виглядає трохи інакше.



**Рис 10. Графік регресії з виключенням**

д) Дерево рішень;

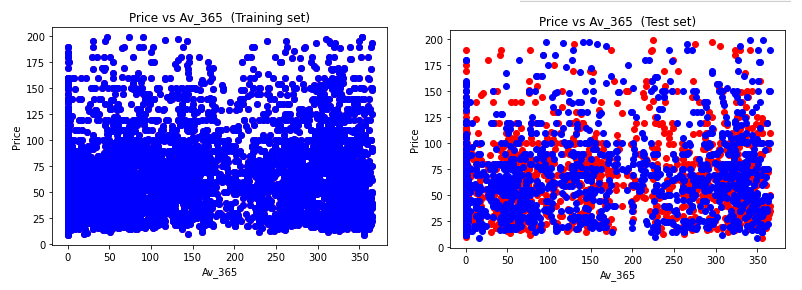
Тут також немає якихось змін, цей метод дає схожу картину.



**Рис 11. Графік дерева рішень**

е) Випадковий ліс.

На мій погляд, навіть модель випадкового лісу не допомагає на цій виборці.



**Рис 12. Графік дерева рішень**

***4.4. Оцінка результатів (Assessing the model)***

Найкращим способом порівняти моделі є таблиця з основними характеристиками.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Модель** | **Одн Р** | **Баг Р** | **РГ с викл** | **Пол Р** | **Р Дер** | **Вип Ліс** |
| **Коеф** | **0.04** | **(-0.006; 0.89; 0.03; -0.007; -0.49 ;2.15 ; 1.81; -3.33)** | **(-0.006; 0.89; 0.03; -0.0008)** | **(0.18; -0.0004)** | **-** | **-** |
| **Вільний** | **60.14** | **63.27** | **63.26** | **55.44** | **-** | **-** |
| **R^2(tr)** | **0.02** | **0.032** | **0.03** | **0.03** | **0.05** | **0.82** |
| **R^2(test)** | **0.004** | **0.004** | **0.04** | **0.06** | **0.07** | **0** |
| **MSE(tr)** | **1343** | **1324** | **1325** | **1319** | **1296** | **250** |
| **MSE(test)** | **1281** | **1261** | **1256** | **1232** | **1227** | **1354** |

**Таблиця 1. Основні характеристики регресійних моделей**

Якщо робити висновок з цього, то, на мою думку, ці моделі лише можуть показати «додану вартість» до ціни, яка формується іншими факторами. Тобто ці моделі можуть показати, чому дві однакові квартири з різною доступністю коштують 70 та 75 долларів.

На жаль, ця модель у цьому виді не може бути використана для бізнес-цілей або вирішення прикладних задач та вивчення ринку, бовона дає дуже погані та неточні результати, але деяку інформацію з неї почерпнути можна.

***Класифікація***

***4.1. Вибір алгоритмів (Selecting the modeling technique)***

Для задачі класифікації ми вибираємо такі методи:

а) Логістична регресія

б) Метод К найближчих сусідів

в) Метод опорних векторів

г) Натівний Байес

д) Логістичне дерево(або дерево класифікацій)

***4.2. Планування тестування (Geneating a test design)***

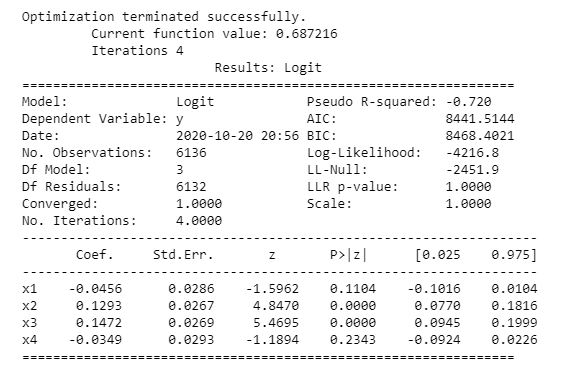
Для усіх задач ми будемо використовувати ділення 80/20, де 80 це тренувальна вибірка, а 20 – тестова.

***4.3. Навчання моделей (Building the models)***

Перед використанням моделей важливо зазначити, що ми не мали так званої «логічної» змінної, яку ми описуємо(наша ціна була цілим числом, але не факторним). Для вирішення задачі класифікації ми введемо «розкішне житло» lux\_pr, яка дорівнює нулю якщо житло коштує менше 100 долларів і 1, якшо більше. Після цього у цих моделях працюємо саме з lux\_pr.

Відмічаємо, що усі бінарні змінні room\_type ми відкидаємо – у моделі їх використати неможливо.

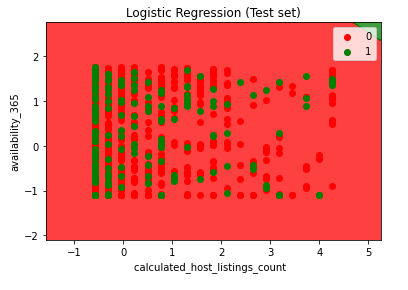
Перед реалізацією усіх методів, ми маємо знайти дві змінні, з якими будемо працювати для опису. Це робиться за допомогою статистичної моделі з такими результатами:



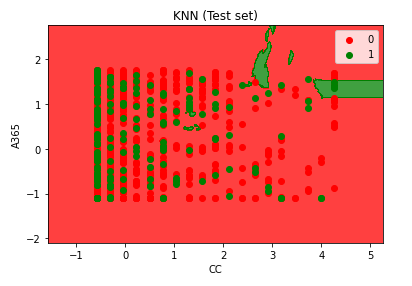
**Рис 13. Перевірка параметрів логістичної регресії**

Очевидно, що беремо найменші за Р-значенням – х2 та х3. Це в нас calculated\_host\_listings\_count та availability\_365 – саме на них плануємо будувати графіки. Перейдемо до моделей, але зауважу, що числові параметри та їх порівняння ми розглянемо у пункті 4.4. До речі, саме тут обов’язково шкалюємо дані для роботи.

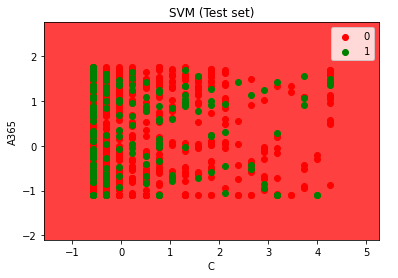
А поки покажемо графіки з результатами.



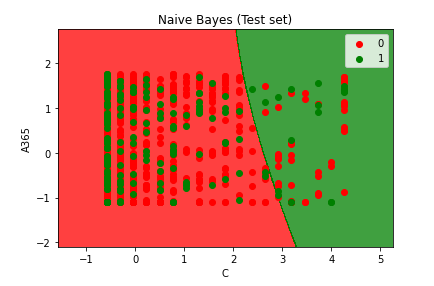
**Рис 14. Графік логістичної регресії**



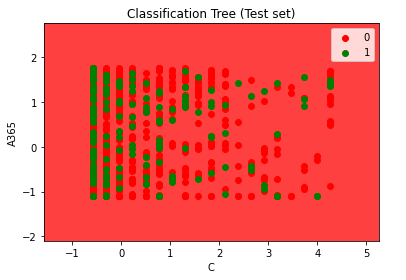
**Рис 15. Графік методу К найближчих сусідів**



**Рис 16. Графік методу опорних векторів**



**Рис 17. Графік Байесового методу**



**Рис 18. Графік дерева класифікацій**

***4.4. Оцінка результатів (Assessing the model)***

Для оцінки результатів побудуємо порівняльну таблицю та калькулюємо основні параметри перевірки:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Модель | R | TN | FP | FN | TP |
| Log Reg | 0.87 | 1333 | 0 | 0 | 0 |
| K-NN | 0.81 | 1332 | 1 | 5 | 5 |
| SVM | 0.87 | 1333 | 0 | 0 | 0 |
| NB | 0.85 | 1285 | 48 | 18 | 18 |
| Tree | 0.87 | 1333 | 0 | 0 | 0 |

**Таблиця 2. Розподілення TN, TP, FP, FN**

Також розрахуємо акуратність, помилковість, чутливість та специфічність кожної з моделей.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Модель | АR | ER | Sen | Spec |
| Log Reg | 0.868 | 0.131 | 0 | 1 |
| K-NN | 0.871 | 0.128 | 0.024 | 0.999 |
| SVM | 0.868 | 0.131 | 0 | 1 |
| NB | 0.85 | 0.15 | 0.089 | 0.96 |
| Tree | 0.868 | 0.131 | 0 | 1 |

**Таблиця 3. Калькулювання AR, ER, Sensitivity, Specialty**

Якщо ж робити висновок, то, якщо перевіряти цифри, то можна віддати перевагу методу найближчих сусідів. Але, якщо подивитися, то саме Байесов метод, незважаючи на показники трохи гірше, вірно визначив деяку частину дорогих квартир і у цьому показнику все ж дав цікаву інформацію.

Загалом ці дані я би не використовував для вирішення бізнес-цілей, але с точки зору залежностей для загального ринку ці цифри можна перевірити і дослідити додатково.

***Кластеризація***

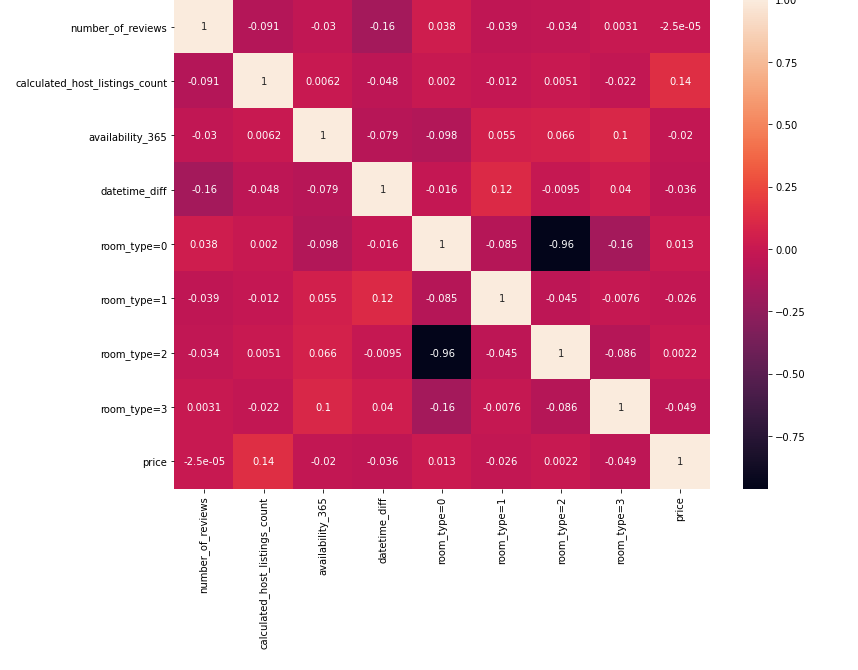
***4.1. Вибір алгоритмів (Selecting the modeling technique)***

Для вирішення задачі кластеризації ми використаємо два методи – ієрархічну кластеризацію та кластеризацію К-середніх.

Треба зазначити, що через обмежені можливості техніки для вирішення ми беремо перші 500 значень з нашого сету. Звісно, це може вплинути на результати, але загалом якщо в нас є цікаві кластери, ми їх знайдемо.

На цей раз шкалюємо дані іншим методом.

Після будування кореляційної матриці я отримав тільки одну цікаву залежність – двох типів кімнат. Це не має сенсу у прямому значенні, але цікаво, які саме кластери елементів за іншими признаками знайде алгоритм.



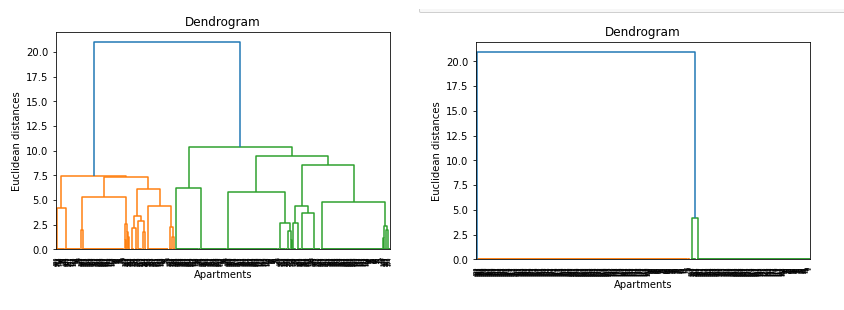
***Рис. 19. Кореляційна матриця***

***4.2. Планування тестування (Gene***[***r***](https://dist.karazin.ua/moodle/mod/glossary/showentry.php?eid=80607&displayformat=dictionary)***ating a test design)***

Оскільки це алгоритм навчання без вчителя, тому усі 500 елементів це тестова вибірка.

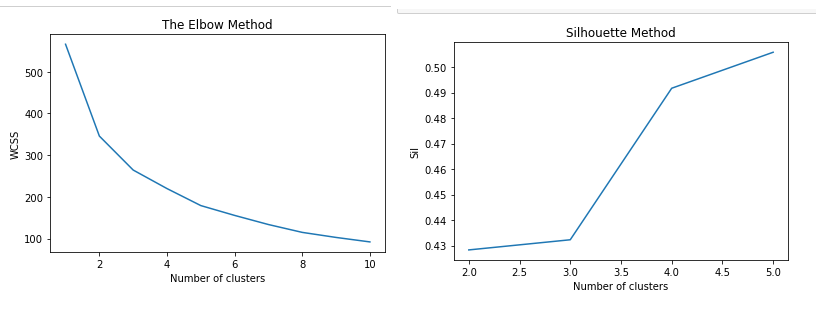
***4.3. Навчання моделей (Building the models)***

а) Ієрархічна кластерізація. Після будування двох дендрограм зрозуміло, що оптимальна кількість кластерів – 2. Їх і будуємо.



**Рис. 20. Дендрограми**

б) К-середніх. За допомогою метода ліктя та методу силуету будуємо найкращу кількість кластерів. Бачимо, що за методом ліктя найкраща кількість – 2, а ось метод силуету показує 5, але очевидно, що якщо поставити ранжування більше, то він покаже і більше кластерів.



**Рис. 21. Метод ліктя та метод силуету**

Два кластери які видало для нас тут, це:  
1. Квартири що доступні, мають мало обзорів та невисоку ціну типу 0;

2. Доступні квартири, мало обзорів та невисока ціна типу 2;

Загалом бачимо, що у цьому сеті дійсно є перекіс до таких апартаментів, і ці два типові кластери виділяються.

***4.4. Оцінка результатів (Assessing the model)***

Очевидно, що практичного змісту ми з цих моделей не отримаємо. Але з точки зору оцінювання стійкості кластерів, можна побачити, що вони перетинаються майже на 50%, тому цей показник ми можемо прийняти як задовільний з технічної точки зору.

**5. Оцінка результату (Evaluation)**

***5.1. Оцінка результатів моделювання (Evaluating the [r](https://dist.karazin.ua/moodle/mod/glossary/showentry.php?eid=80607&displayformat=dictionary" \o "Словник основних термінів: R)esults)***

В нас не стояло бізнес-завдань та економічних задач, тому усі висновки по моделюванню вже були зроблені у попреденіх пунктах.

***5.2. Аналіз процесу виконання проекту (***[***R***](https://dist.karazin.ua/moodle/mod/glossary/showentry.php?eid=80607&displayformat=dictionary)***eview the p***[***r***](https://dist.karazin.ua/moodle/mod/glossary/showentry.php?eid=80607&displayformat=dictionary)***ocess):***

**А) чи можна було якісь кроки зробити більш ефективними:**

1. Звісно, ми могли не видаляти багату кількість пропущених значень та викідів, а працювати з ними. Але через обмежені потужності обчислювальної техніки цей вибір можна прийняти;

2. Деякі змінні можна було логарифмувати – це могло спотворити коефіціенти, але покращити моделі загалом;

3. Можна було кодувати більшу кількість якісних змінних, які ми видалили. Це ускладнило би моделювання, але, можливо, дало більше цікавих залежностей для навчання;

4. У нас була тільки одна змінна, що пояснювалася, тому для вирішення задач класифіції довелось створювати фіктивне обмеження у 100 долларів, яке не має економічного обґрунтування.

5. Можливо, ще деякі змінні(особливо ті, які ми не використали для моделі) повинні були бути перероблені, змінені, поєднані. Можливо треба було взяти додаткові дані з інших джерел, для того, щоб створити більш цінні моделі.

6. Можна було у моделях класифікації перевірити різну кількість кластерів.

**Б) які були допущені помилки, як їх уникнути в майбутньому:**

1. Відсутність логарфмування деяких змінних;

2. Помилкове видалення частини масиву даних, який можливо було відновити;

3. Низька точність деяких моделей регресії, треба більш точно вибирати дані для них;

**В) чи були гіпотези, що не спрацювали, чи варто їх перевіряти повторно:**

Єдина гіпотеза, до якої є питання – це зміна часу у моделі – різниця дат. Очевидно, що через місяць там зміняться усі значення, тому, можливо, вплив цієї змінної можна розглянути у майбутньому.

**Г) чи були несподіванки під час реалізації кроків, як їх передбачити в майбутньому:**

Несподіванок не було, але результати прогону моделей незадовільні. Це неможливо передбачити, але можливо покращити – наприклад, достатньо ретельно перевіряти дані до старту проекту.

***5.3. Ухвалення рішення (Dete***[***r***](https://dist.karazin.ua/moodle/mod/glossary/showentry.php?eid=80607&displayformat=dictionary)***mining the next steps)***

Основи цих моделей ми, звісно, залишемо, але необхідно узяти нові дані та, можливо, переглянути задачі для моделювання.