Data Mining - Homework 1

Yoosef Ghaderi

March 10, 2025

Part 1: Theoretical Questions

Question 1

1. First Part: Table below contains some attributes of features
   - Yera of construction : discrete , numerical
   - Area in square meters :Countinues , numerical
   - Floor :discrete , numerical
   - total number of floors: discrete , numerical
   - Type of house : discrete , nominal
   - Number of rooms : Discrete , numerical
   - has elevator : binary , nominal
   - has parking : binary , nominal
   - secrurity level of area : ordinal
   - type of floor covering : nominal
   - price unit : numerical , continues
2. Second Part: Below is a description of the appropriate plots for each feature in the dataset, along with the reasoning for choosing each plot type.
   - **Year of construction:**
     - **Plot:** Histogram (to show the distribution of construction years).
     - **Reason:** Numerical and discrete data.
   - **Area in square meters:**
     - **Plot:** Histogram (to show the distribution of areas).
     - **Reason:** Numerical and continuous data.
   - **Floor:**
     - **Plot:** Bar chart (to show the frequency of each floor).
     - **Reason:** Discrete and ordinal data.
   - **Total number of floors:**

- **Plot:** Bar chart (to show the frequency of buildings with a specific number of floors).
  - **Reason:** Discrete and ordinal data.

- **Type of house:**
  - **Plot:** Pie chart (to show the proportion of each house type).
  - **Reason:** Nominal and categorical data.

- **Number of rooms:**
  - **Plot:** Bar chart (to show the frequency of houses with a specific number of rooms).
  - **Reason:** Discrete and ordinal data.

- **Has elevator:**
  - **Plot:** Pie chart (to show the proportion of houses with and without elevators).
  - **Reason:** Binary and nominal data.

- **Has parking:**
  - **Plot:** Pie chart (to show the proportion of houses with and without parking).
  - **Reason:** Binary and nominal data.

- **Security level of the area:**
  - **Plot:** Bar chart (to show the frequency of each security level).
  - **Reason:** Ordinal and discrete data.

- **Type of floor covering:**
  - **Plot:** Pie chart (to show the proportion of each floor covering type).
  - **Reason:** Nominal and categorical data.

- **Price unit:**
  - **Plot:** Box plot (to show the distribution and outliers of prices).
  - **Reason:** Numerical and continuous data.

3. **Relationship Between Area in Square Meters and Price Unit:**

   - **Plot**: Scatter Plot

   Reason for Using a Scatter Plot:

   - A scatter plot is used to visualize the relationship between two numerical variables.
   - It helps identify trends, correlations, and outliers between the area of a property and its price.

4. **Correlation Between Area in Square Meters and Price Unit:** "In general, there's a **positive correlation** between house price and area, meaning larger houses tend to cost more. However, this correlation isn't perfect—that is, it doesn't equal 1.

If it did, all data points would fall exactly on a **straight line**, implying that price would always increase proportionally with area, with no exceptions. Yet, as the problem suggests, some houses with larger areas have lower prices, reflecting **other influencing factors**.

This prevents the correlation from reaching 1. Still, the closer the data points align to a straight line, the nearer the correlation gets to 1, indicating a stronger linear relationship."

Question 2

1. Metric to be calculated for each course:

   Math Grades

   - Mean = 16.525
   - Standard Deviation ≈ 1.78
   - Median = 16.5
   - First Quantile (Q1): 15.0
   - Third Quantile (Q3): 17.625

   Physics Grades

   - Mean = 74.15
   - Standard Deviation ≈ 24.24
   - Median = 81
   - First Quantile (Q1): 73.75
   - Third Quantile (Q3): 87.25

2. Since in our data we have outliers I recommend to use median or even to undrestand better of our data we can report first and third quantile too , mean is not a good chioce since it is not resistant about outliers.

3. box plot is a good way to find outliers data we can consider data that are more or less than $(1.5 \times (Q3 - Q1)), somefacotrslikemeanandstandarddeviationarenotresistanttooutliersand$

4. Histogram of each course:

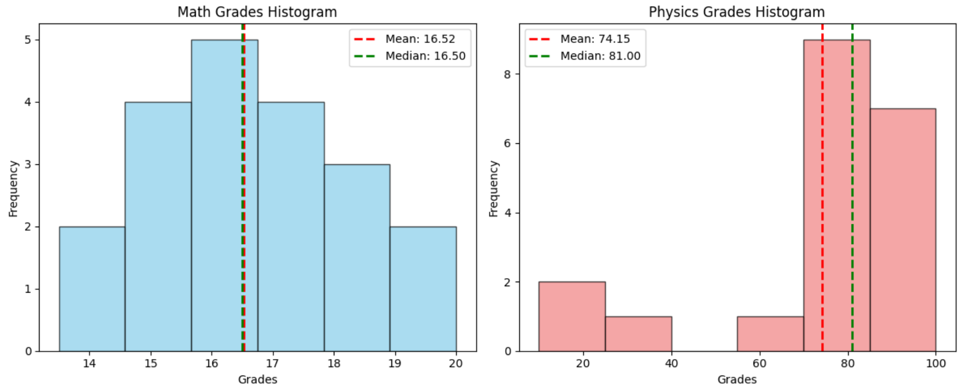


Figure 1: Histogram of two courses.

Interpretation:

Math:

The grades are clustered around 16 to 17.

The distribution appears **fairly symmetric**, indicating a normal spread of scores.

Physics:

The grades are more spread out, ranging from low scores (near 10-20) to high scores (above 80).

The mean (74.15) is lower than the median (81.00), suggesting a left-skewed distribution.

The presence of lower scores pulls the mean down, while most students have higher scores.

5. Plot boxplot of each courses

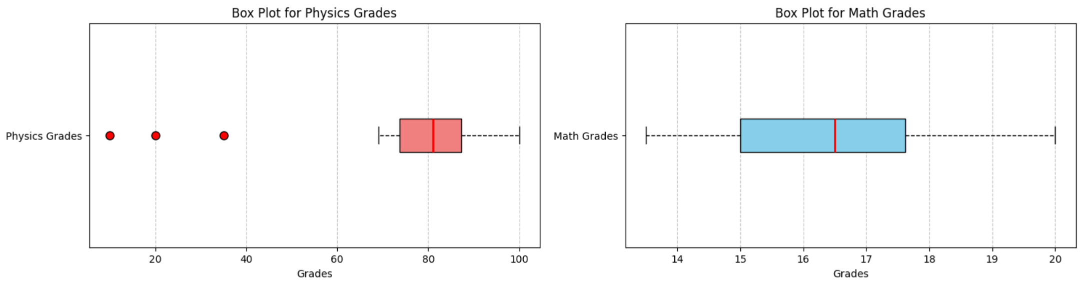


Figure 2: Boxplot of two courses.

interpretation: we can figure out there are more outliers in physics than math course and standard deviation of student in physics is more than math (Even if we convert these two scores to the similar scale)

6. After normalization of grades based on this formula :

$$\text{Normalized Value} = \frac{x - \mu}{\sigma}$$

where μ is the mean and σ is the standard deviation.

I plot it as follow :

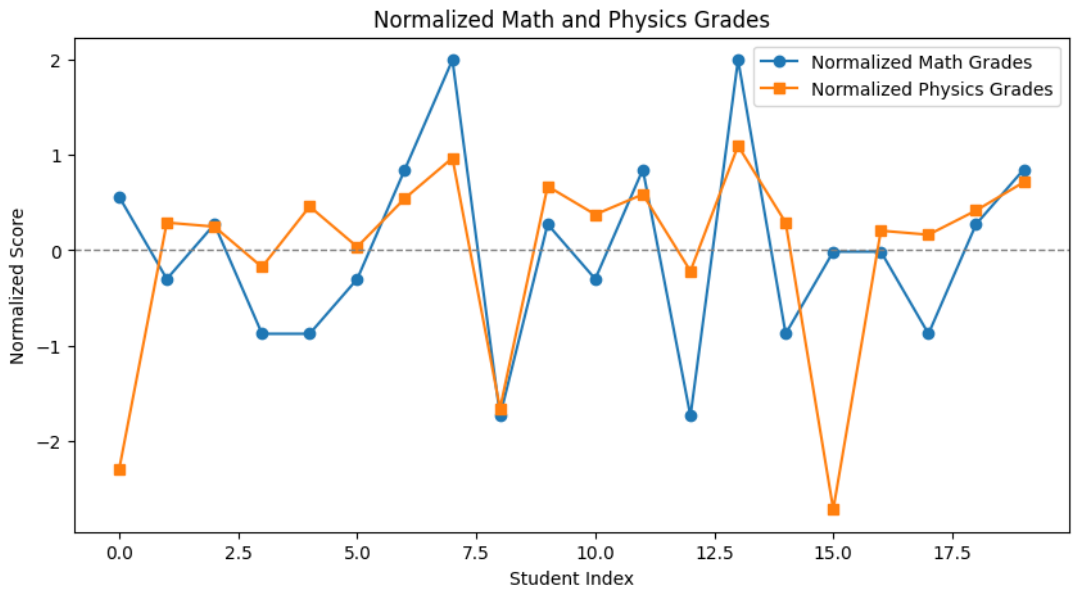


Figure 3: normalized score.

some student have a considerable gap between physics and math grade (if these scores are for a student) and it is strange and should be checked in my opinion Because it is normally expected that students who get a good grade in one of these two subjects will be able to perform at approximately the same level in the next subject, rather than their performance changing significantly.

7. **QQ-plot** is a graphical tool used to compare two probability distributions.

    - Sort the Data
    - Assign Probabilities:Each value is assigned a cumulative probability based on its rank.
    - Find Corresponding Theoretical Quantiles
    - Plot the Points

    Interpretation :

    - If the points closely follow a straight line(like y=x)the two distributions are similar
    - Deviations from the straight line indicate differences in distribution

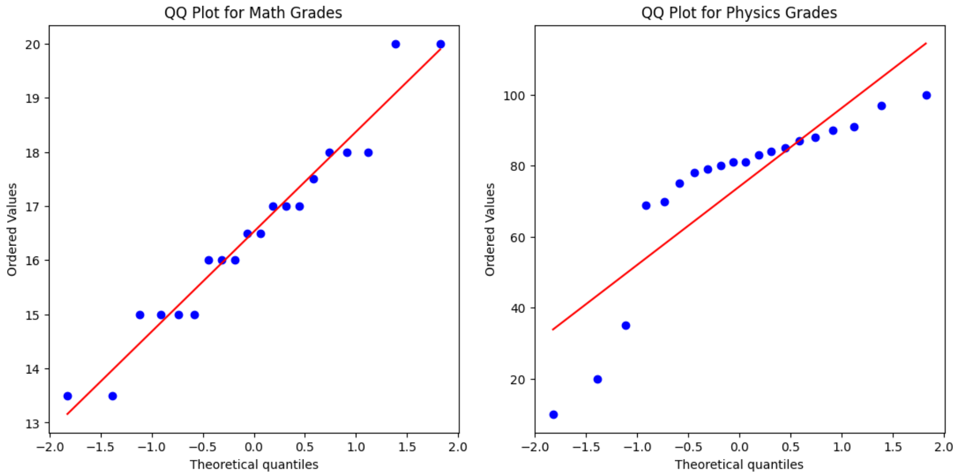


Figure 4: QQ plot of two courses compared to normal distribution.

8. **Correlation of two courses**

   to check correlation of two courses we use of scatter plot and also **Pearson Correlation Coefficient** to measure strength of linear relationship:

$$r = \frac{\sum(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum(X_i - \bar{X})^2 \cdot \sum(Y_i - \bar{Y})^2}}$$

   After calculation r = 0.359 so there is a weak positive linear relationship between Math and Physics grades.

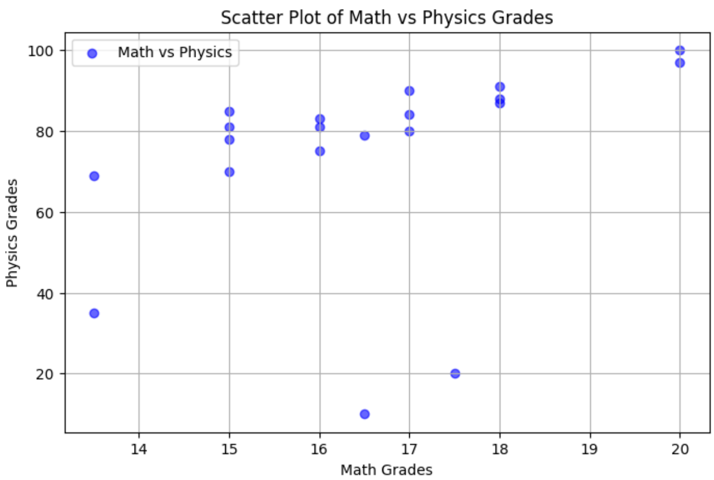


Figure 5: Scatter plot for two courses.

9. **Reasons of deletion:**

   - Non-numeric values like A+ , B in dataset.
   - Out of range data like 21 in math dataset (range 0-20)
   - Outliers data look zero in both datasets.
   - Missing values like N/A in math datset.

- Not in currect format like ”19” or ”84” convert them to right format.
- reason of deletion 90 and 76 : since we deleted corresponding value in math , we delete these grades in physics.

Question 3

We set up the null (H_0) and alternative (H_1) hypotheses:
- H_0 (Null Hypothesis):The two variables (gender and field of study) are independent.
- H_1 (Alternative Hypothesis): The two variables are not independent
contingency table:

	Computer Science	**Electrical Eng.**	**Mechanical Eng.**	**Total**
Boys	30	40	50	120
Girls	50	30	20	100
Total	80	70	70	220

This table contains the observed frequencies (O_{ij}). The expected frequency for each cell (E_{ij}) is calculated using:

$$E_{ij} = \frac{(\text{Row Total}) \times (\text{Column Total})}{\text{Grand Total}}$$

after computing contingency table for expected values we calculate Chi-square statistic as follow :

$$\chi^2 = \sum \frac{(O_{ij} - E_{ij})^2}{E_{ij}}$$

where:
- O_{ij} = observed frequency for cell (i, j)
- E_{ij} = expected frequency for cell (i, j)
Also to calculate degree of freedom :

$$df = (r - 1) \times (c - 1)$$

$$df = (2 - 1) \times (3 - 1) = 1 \times 2 = 2$$

After checking p-value for this statistic (p-value = 0.00014) , so we reject H_0

Part 2: Practical Questions

1. We did it :)
2. We implemented it in ipynb file :)
3. Here are the challenges we faced:
   - **Inconsistent Number of Columns in DataFrames**: We have a total of 5 DataFrames, some of which have varying numbers of columns. To address this, we identified the extra columns and added the necessary values to align them. During this process, we noticed that df1 contained a unique column named **”exchangeable”** that was not present in any other DataFrame. As a result, we added this feature to other df with concat function and the value now is NaN.

- Missing "Price per Meter" in df2 : We observed that the "price per meter" parameter was missing in df2. However, this value can be derived from two other features in the DataFrame. Therefore, we removed this column and decided to calculate it dynamically using the two relevant features whenever needed.
- **Inconsistent Naming Conventions**: The column names across the DataFrames are inconsistent and need to be standardized. For example, the "title" column is written differently in some DataFrames (e.g., "Title"). We need to unify the naming conventions to ensure consistency across all DataFrames.

Part B is done in notebook.

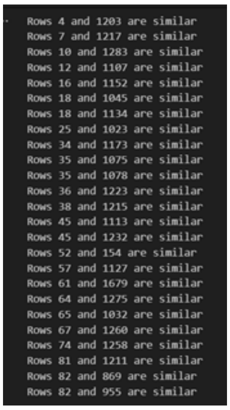
4. After analyzing the data related to the two features, **buildyear** and **total price**, we identified specific formatting requirements. The **total price** should be at least 7 digits long, considering that the prices are in **Tomans**. Additionally, the buildyear must follow a **4-digit format** and should be greater than 1300.

   Since the data is currently in **string (Str)** format, it needs to be converted to **int64** before applying these validation rules. After performing the necessary conversions and applying the filters also I checked about some invalid rows that The total price field value was incorrectly filled with the propertyisze value. By examining these rows, I noticed that the other values in these rows were also not null or not found, so I deleted them as well, the dataset was significantly reduced.

5. We want to identify duplicate listings in the dataset. Some listings may have different titles but similar or identical features (e.g., description, price, build year, etc.). Additionally, some listings may have high similarity in text-based features like **title** and **description**, even if they are not exact duplicates.

   - **Exact Matching:**
     - Compare non-text features (e.g., `price`, `buildyear`, `location`) to find listings with identical values.
   - **Text Similarity:**
     - Use cosine similarity on text-based features like `title` and `description` to identify listings that are semantically similar.
     - Set a threshold for cosine similarity to determine if two listings are duplicates.

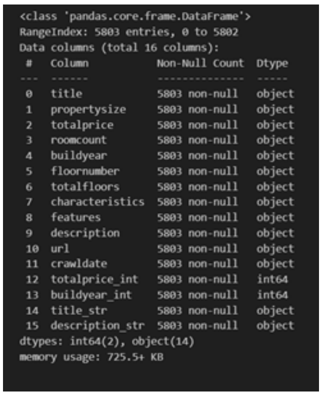
   My approach was focused on finding simialar advertisement with same feature in this dataset (I considered really tight condition to find similar advertisement) but there was many duplicated advertisment.



```
Rows 4 and 1203 are similar
Rows 7 and 1217 are similar
Rows 10 and 1283 are similar
Rows 12 and 1107 are similar
Rows 16 and 1152 are similar
Rows 18 and 1045 are similar
Rows 18 and 1134 are similar
Rows 25 and 1023 are similar
Rows 34 and 1173 are similar
Rows 35 and 1075 are similar
Rows 35 and 1078 are similar
Rows 36 and 1223 are similar
Rows 38 and 1215 are similar
Rows 45 and 1113 are similar
Rows 45 and 1232 are similar
Rows 52 and 154 are similar
Rows 57 and 1127 are similar
Rows 61 and 1679 are similar
Rows 64 and 1275 are similar
Rows 65 and 1032 are similar
Rows 67 and 1260 are similar
Rows 74 and 1258 are similar
Rows 81 and 1211 are similar
Rows 82 and 869 are similar
Rows 82 and 955 are similar
```

Figure 6: similar rows in dataset.

6. (a) It is don in ipynb file here is result :



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5803 entries, 0 to 5802
Data columns (total 16 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   title            5803 non-null   object
 1   propertysize     5803 non-null   object
 2   totalprice       5803 non-null   object
 3   roomcount        5803 non-null   object
 4   buildyear        5803 non-null   object
 5   floornumber      5803 non-null   object
 6   totalfloors      5803 non-null   object
 7   characteristics  5803 non-null   object
 8   features         5803 non-null   object
 9   description      5803 non-null   object
 10  url              5803 non-null   object
 11  crawldate        5803 non-null   object
 12  totalprice_int   5803 non-null   int64
 13  buildyear_int    5803 non-null   int64
 14  title_str        5803 non-null   object
 15  description_str  5803 non-null   object
dtypes: int64(2), object(14)
memory usage: 725.5+ KB
```

Figure 7: Histogram of two courses.

Pandas dtype	**Python/NumPy type**	**Usage**
object	str/mixed, string_, unicode_	Text/mixed values
int64	int, int_, int8-64, uint8-64	Integers
float64	float, float_, float16-64	Floating points
bool	bool, bool_	True/False
datetime64	datetime64[ns]	Date/time
timedelta[ns]	-	Time differences
category	-	Text categories

(b) To have a currect analyze we need to change type of each feature based on its usage in practical so: ”propertysize”, ”roomcount”, ”floornumber”, ”totalfloors” to int64 and ”crawldate” to datetime.

(c) when we were checking data types of variables we understood that in some columns datatypes are numerical and string for example in totalprice there were some rows datatype was numerical and some rows datatype was string to handle this we build a new row that just contains int64 datatypes.

(d) Before changing memory usage: 634.8+ KB After changing memory usage: 623.4+ KB

```
... <class 'pandas.core.frame.DataFrame'>
RangeIndex: 4763 entries, 0 to 4762
Data columns (total 16 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   title            4763 non-null   object
 1   propertysize     4409 non-null   Int64
 2   totalprice       0 non-null      Int64
 3   roomcount        4718 non-null   Int64
 4   buildyear        4763 non-null   Int64
 5   floornumber      2586 non-null   Int64
 6   totalfloors      2591 non-null   Int64
 7   characteristics  4763 non-null   object
 8   features         4763 non-null   object
 9   description      4763 non-null   object
 10  url              4763 non-null   object
 11  crawldate        4727 non-null   datetime64[ns]
 12  totalprice_int   4763 non-null   int64
 13  buildyear_int    4763 non-null   int64
 14  title_str        4763 non-null   object
 15  description_str  4763 non-null   object
dtypes: Int64(6), datetime64[ns](1), int64(2), object(7)
memory usage: 623.4+ KB
```

Figure 8: Data types after changing .

(e) in figure 7 it is clear that we changed datatype to currect types , reason of object type is because of these features contain string and numbers.

7. We convert these two columns and get this results :

```
... Feature | Ads Count | Percentage
تأمین‌کننده آب گرم پکیج | 2296 | 39.57%
جنس کف سنگ | 571 | 9.84%
not found | 34 | 0.59%
سرویس بهداشتی ایرانی | 3171 | 54.64%
Not Found | 40 | 0.69%
جنس کف کفپوش PVC | 16 | 0.28%
جنس کف موزائیک | 7 | 0.12%
گرمایش داکت اسپلیت | 380 | 6.55%
گرمایش از کف | 73 | 1.26%
انباری | 5729 | 98.72%
سرمایش اسپلیت | 391 | 6.74%
گرمایش بخاری | 389 | 6.70%
سرویس بهداشتی ایرانی و فرنگی | 2623 | 45.20%
گرمایش شوفاژ | 2104 | 36.26%
گرمایش اسپلیت | 39 | 0.67%
سرمایش فن کوئل | 107 | 1.84%
گرمایش شومینه | 12 | 0.21%
گرمایش فن کوئل | 114 | 1.96%
آسانسور | 5729 | 98.72%
تأمین‌کننده آب گرم موتورخانه | 379 | 6.53%
جنس کف پارکت لمینت | 92 | 1.59%
تأمین‌کننده آب گرم آبگرم‌کن | 436 | 7.51%
سرمایش کولر گازی | 584 | 10.06%
سرمایش داکت اسپلیت | 464 | 8.00%
...
جنس کف سرامیک | 2432 | 41.91%
انباری ندارد | 451 | 7.77%
پارکینگ ندارد | 614 | 10.58%
بالکن | 3382 | 58.28%
```

Figure 9: Extract features and precent in adv.

```
Feature | Ads Count | Percentage
سند: تکبرگ | 2814 | 48.49%
وضعیت واحد: بازسازی شده | 1208 | 20.82%
تعداد واحد در طبقه: ۲ | 1393 | 24.00%
not found | 33 | 0.57%
Not Found | 36 | 0.62%
تعداد واحد در طبقه: ۷ | 11 | 0.19%
تعداد واحد در طبقه: ۴ | 255 | 4.39%
تعداد واحد در طبقه: ۸ | 48 | 0.83%
تعداد واحد در طبقه: بیشتر از ۸ | 17 | 0.29%
جهت ساختمان: شرقی | 77 | 1.33%
جهت ساختمان: جنوبی | 2028 | 34.95%
تعداد واحد در طبقه: ۱ | 1213 | 20.90%
جهت ساختمان: شمالی | 905 | 15.60%
سند: منگوله‌دار | 15 | 0.26%
سند: سایر | 138 | 2.38%
تعداد واحد در طبقه: ۵ | 44 | 0.76%
تعداد واحد در طبقه: ۶ | 50 | 0.86%
تعداد واحد در طبقه: ۳ | 275 | 4.74%
سند: قولنامه‌ای | 240 | 4.14%
جهت ساختمان: غربی | 25 | 0.43%
وضعیت واحد: بازسازی نشده | 4511 | 77.74%
```

Figure 10: Extract charecteristic and precentage in adv.

when I checked each features and precentage and importance of each attribute in real world I added 4 columns name :

- documnet type
- parking
- warehouse
- elevator

these are some features important and we can consider specific columns for them. documentt type can have some string value so it is categorical and three other attributes are binary.

8. - I did it in notebook.
   - we can extract useful information of these two column like neighborhood of advertisment and a database of people who are in the business of buying and selling property.We can also find out if the person who posted the ad is the **property owner** or someone who is in the business of buying and selling houses.
     - sample 1:

'سلام و درود ❤️\n\nهلدینگ املاک پلاس ❤️\n\xa0\n ❤️ بزرگترین بانک اطلاعاتی املاک شمال اصفهان
✅\n\n❤️فروش آپارتمان 110 متر / دو خواب فول کمد دیواری /بدون پرتی/ خوش نقشه/جنوبی
شمالی ✅\n\nنقشه فوق العاده عالی بدون پرتی ☑️\n\n ... عالی برای عروس و داماد ها برای سرمایه گذاری
✅\n\nخوش نقشه بدون پرتی (بی نظیر) ☑️\n\n آدرس : خانه اصفهان / فلکه اطلسی /گلستان غربی
✅\n\nبرای اطلاعات بیشتر و بازدید از منزل تماس بگیرید ☑️\n\n...مشابه این نمونه فایل در منطقه خانه اصفهان
با قیمت های متفاوت و متراژهای متفاوت موجود می باشد ✅\n\n... قیمت قابل مذاکره می باشد داخل اتاق جلسه قرار
داد ☑️\n\n... لوکیشن عالی دسترسی عالی منطقه ای شناخته شده ✅\n\nمناسب برای عروس و داماد ها ....
☑️\n\nدارای همسایگان عالی و کارمند ✅\n\nفروش منزل خودتون رو به ما بسپارید تا به نتیجه عالی و دلخواه
خود برسید ☑️\n\n...دارای : انباری /پارکینگ/تراس/آسانسور ✅\n\nمعاوضه انجام نمی دهند فقط فروش
نقدی ☑️\n\nوضعیت سند : تک برگ 400 ✅\n\nمیلیون رهن کامل مستاجر داده میشود ❇️\n\n عکس این واحد
آرشیو می باشد ❌\n\n... مشاور شما : مهندس بهنامی فر' ❌

Figure 11: discription 1

     - sample 2:

'فروش فوری امتیاز زیر قیمت بازار\n موقعیت ملک بسیار عالی\n دارای سند تک برگ\nقیمت کارشناسی ملک
الان 3میلیارد و دویست هست\nبه دلیل مشکلات مالی 2میلیارد و نهصد میدم'

Figure 12: discription 2.

and corresponding title was :

'فروش باغ ویلا 4دیواری(سنددار،برجاده،امتیازات کامل)'

'چشمه توتی فروش امتیاز صداوسیما آپارتمان 130 متری'

Figure 13: titles

- if we want to use of data in these two field most important problem is than these two column are not structured and we can't find **specific pattern** such that pattern was in charecteristic and features (we can use — for split them) instead we can use of tools like LLM's to extract important detail of each adv (what I did in bonus part) for example two smples in previeus item will clear everything some of discription is too long and some of them are too short.

9. about these columns : property size , total price if both of them were NaN we should remove them because there is no any data validation but if one of them is a number we can figure out its value based on filtering on other parameters like city and predicted neighboor, like bonus part but since now we have no information about it we can use of room count and charecteristics and features to make a group and then use median or mean to guess missing value. we can also estimate build year of each row based on other features like near price and property size feature. because of time and grace I could not implement these two ways ask you to be kink about it :)

```
title               0.000000
propertysize       10.836653
totalprice          0.000000
roomcount           5.609562
buildyear           0.000000
floornumber        48.223108
totalfloors        48.000000
characteristics     0.000000
features            0.000000
description         0.000000
url                 0.000000
crawldate           4.462151
totalprice_int      5.737052
buildyear_int       5.848606
elevator            0.000000
warehouse           0.000000
parking             0.000000
document_type      47.298805
dtype: float64
```

Figure 14: Precentage of each feature missing data.

-

10. Well to figure out outliers we can use Box plot and we use it for numerical variable here is plot of them :

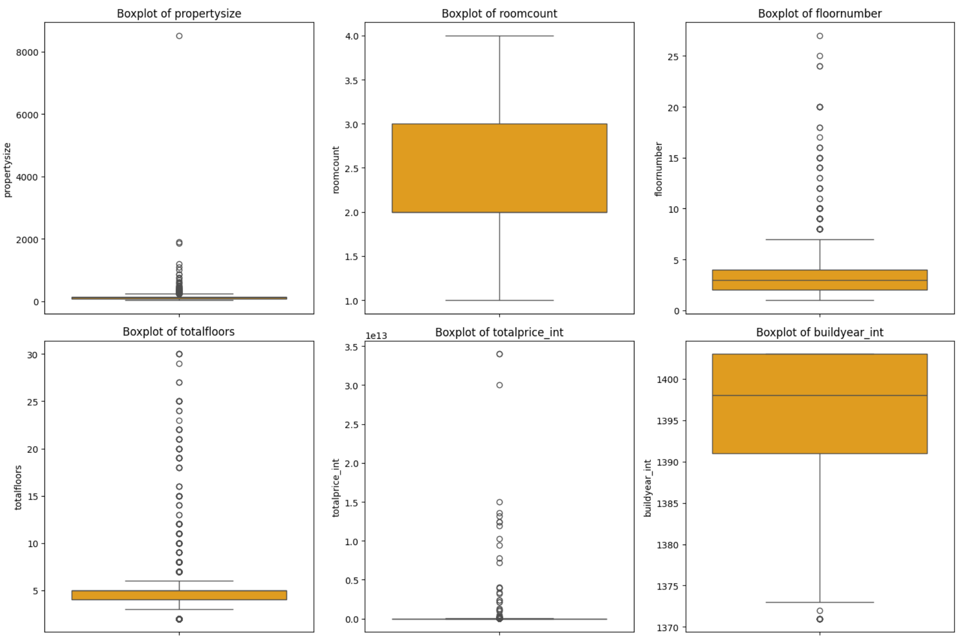


Figure 15: plot of each variable before deletion of outliers.

Handling Outliers:

Outliers can be managed using the following approaches:

- **Remove Outliers**: If outliers are errors or irrelevant, they can be removed.
- **Transform Data**: Apply transformations (e.g., log) to reduce the impact of outliers.
- **Cap/Floor Values**: Replace outliers with the nearest non-outlier value (e.g., whisker limits).
- **Keep Outliers**: If outliers are meaningful, they can be retained for analysis.

but what about our data ? in my opinion it is better to choose removing some unusual data that are not really related to our next conclusions. so I used of this formula to delete outliers:

$$\text{lower\_bound} = Q_1 - 1.5 \times \text{IQR}$$
$$\text{upper\_bound} = Q_3 + 1.5 \times \text{IQR}$$

and I removed them just in features :

1. total price
2. property size

and plot them again.

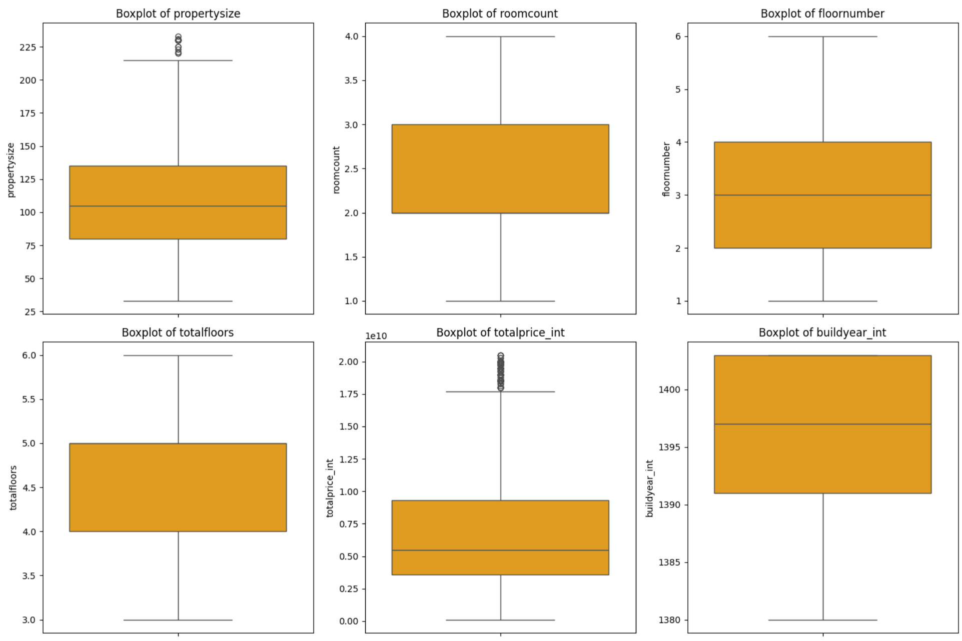


Figure 16: Box plots after removing outliers.

and then we continue to analyzing in next sections.

11. - Bar charts are excellent for comparing values across distinct categories price per meter for each city we use bar chart for this :

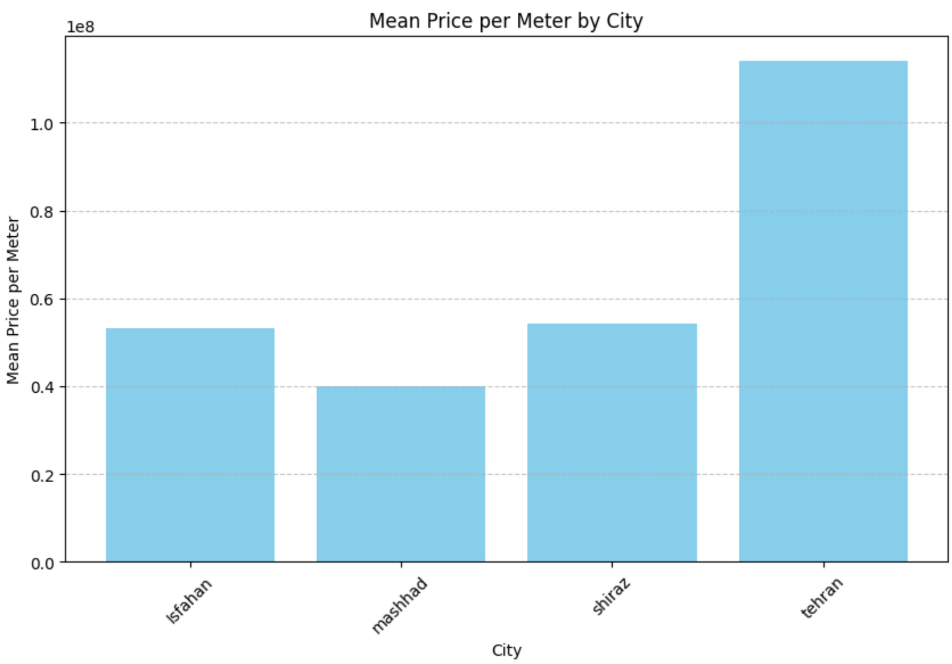


Figure 17: Mean price per meter for each city.

- also we can use barchart to compare mean price per meter for each city based on building year and is suitable to compare:

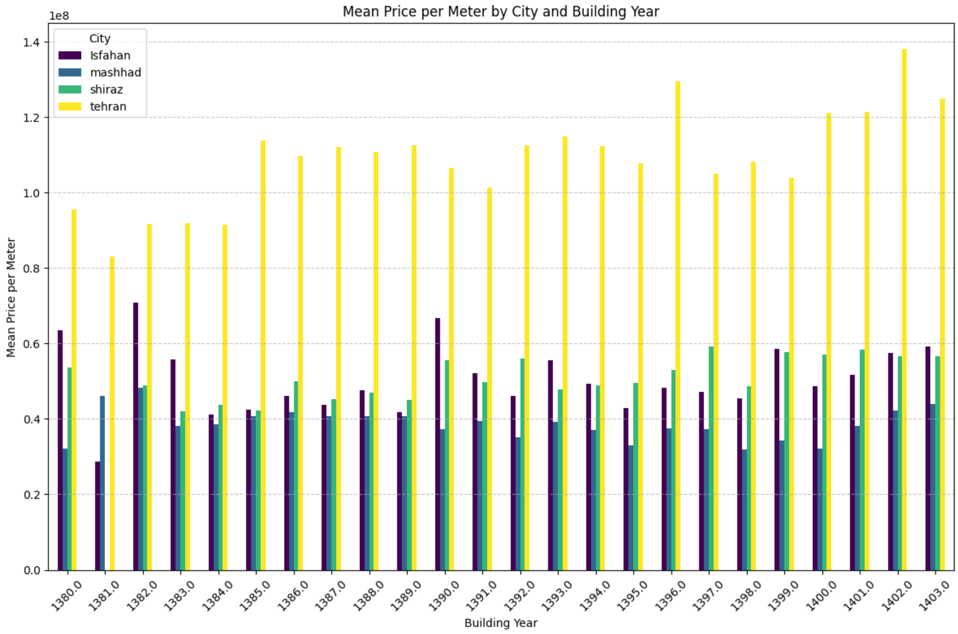


Figure 18: Mean price per meter for each city based on building year.

We can undrestand generally price of house has increased during time but also we should consider price of home is not just related to build year and some other factors are important.

- Now I want to plot distribution of each following features based on city :
  - build year
  - room count
  - price per meter
  - total price

  also we add median for each feature for each city :

  - Isfahan:

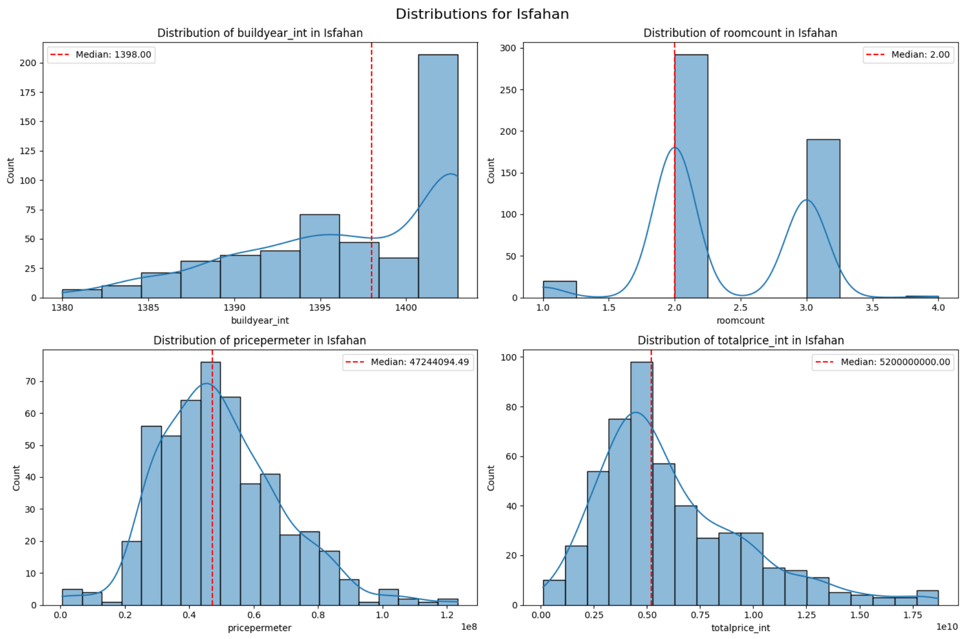


Figure 19: distribution for Isfahan.

– Mashhad:

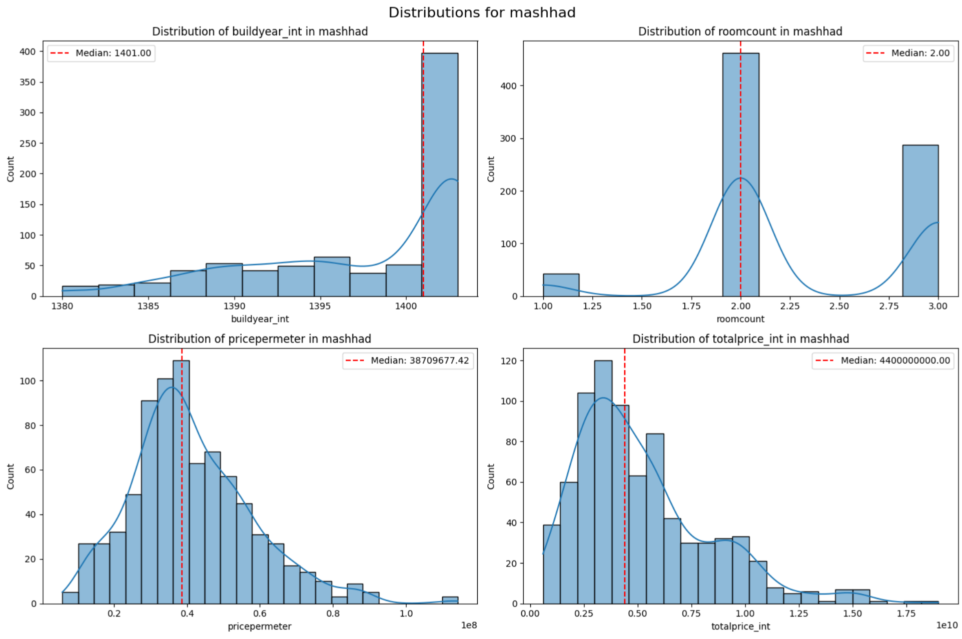


Figure 20: distribution for Isfahan.

– Tehran:

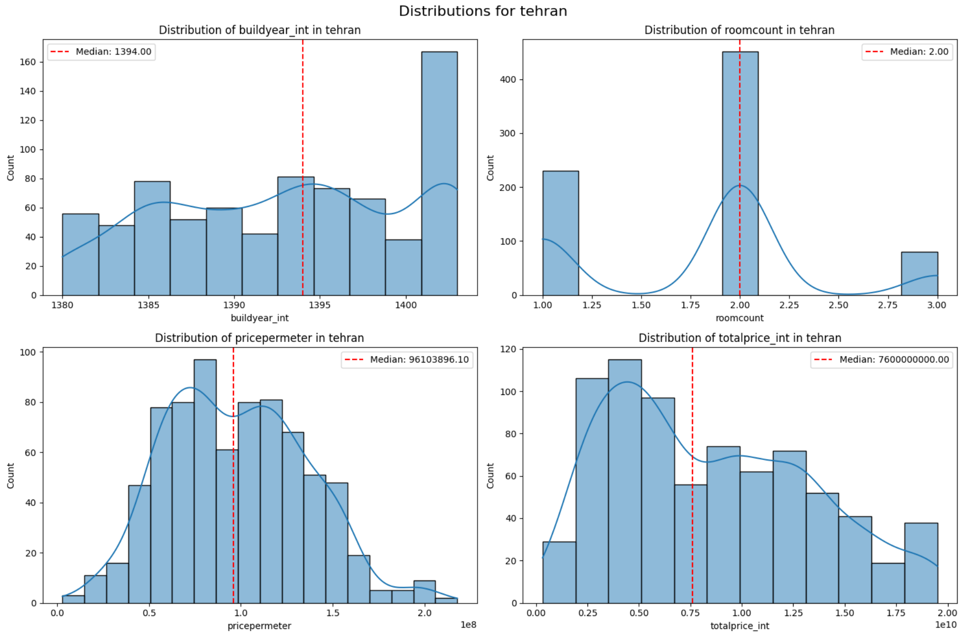


Figure 21: distribution for tehran.

– Shiraz:

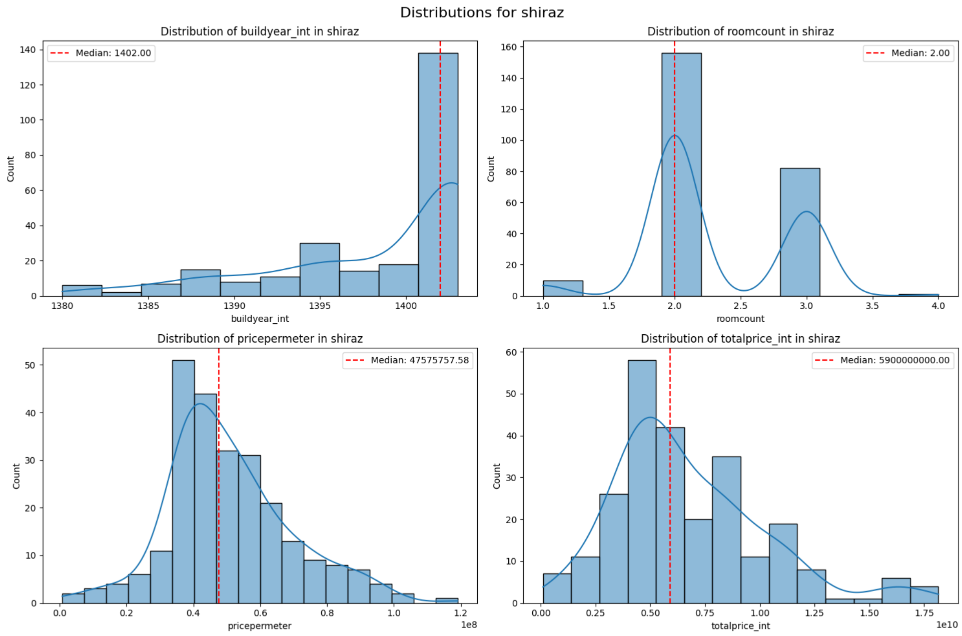


Figure 22: distribution for shiraz.

– Zahedan:

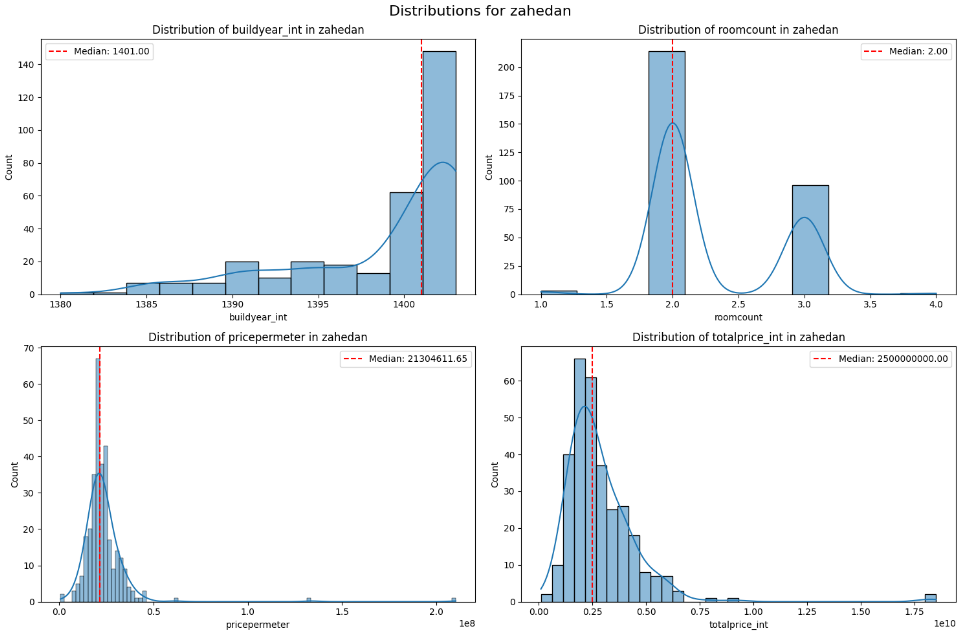


Figure 23: distribution for zahedan.

- now I plot these three features with bubble chart and I have 2 dimension price per meter and property size and also to show room count volume of bubble may help us so :

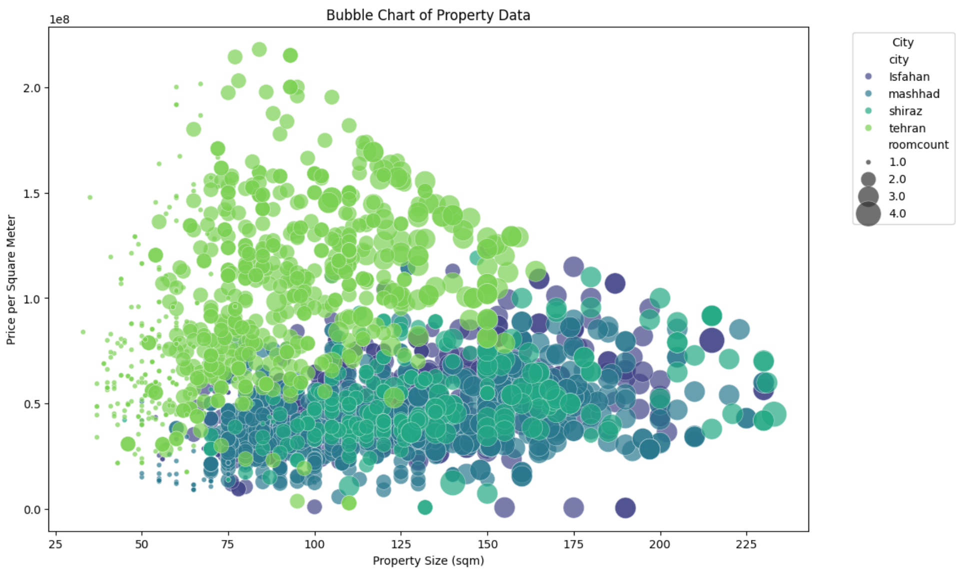


Figure 24: price per meter and property size and room count.

Part 3: Bonus Part

- it is done in notebook (I show 5 element of this dataframe)
- when we checked we undrestood it is possible to find neighborhood of each adv with discription and title

- When we checked discription and title we found these problem:

  1. it ususally contains some number like 47 , 200 , ... and we know neighborhood name does not contain these values

  2. it contains some stickers that we need to eliminate s.th like check point

  3. there are some escape sequences like \n \t \r

  4.there are emotional sentence that we can not label them easily.

- We conclude that these advertisment have different structurs in title and discription so we need to use some feature like LLM to figure out how to extract neighborhood of it we used a togather api ,api key and model Llama 70B LLM helped us to detemine neighborhood of each adv but a problem we faced in this solution was sometimes LLM does not generate a word instead it prints a sentence , my approach was deleting part of speech is in english and just keep persian part of it since my prompt was in english answer of LLM is in english too I also asked LLM to generate final answer in persian so we solved it . unfortunatly number of different neighborhood have increased(For example, the word Yusufabad was written in two separate ways and one above the other.) so I dicided to do additional setting and solved it by using cosine similarity option.

	title	description	predicted_neighborhood
4218	فروش واحدهای کوهک (برج و شخصی)	واحدها در متراژهای مختلف\n در منطقه کوهک\n کافی هستم متخصص خرید و فروش و تهاتر\n با سلام با امکانا...	کوهک
4219	واحد3 خواب خوش نقشه/دوجهت افتابگیر/لوکیشن بدون عیب	سرایداری مقیم 🔹\n\n پارکینگ انباری اسانسور 🔹\n\n.اپارتمان فول امکانات بدون مشابه در منطقه ✅ ... 🔹\n\n	شمیران
4220	متر زیر همکف آذری (اردشیر) پرداختی فقط 1650 47	ورودی n\طبقه زیرهمکف کلا 3 تا پله واقعی\n\n متر تک خواب مستر47\n\nآذری قدرت پاکی افراحته مجزا تک...	آذری
4221	متر شمال جردن با 12 متر پاسیو اختصاصی و لابی 85	پارکینگ انباری\n\nبه قیمت رسیده\n\nخوش قیمت\n\nتاپ لوکیشن شمال جردن\n\n متر بدون پرتی 85 آسانسور...	جردن
4222	آپارتمان 115 متری 3 خواب مقابل پارک کوروش	✅\n\n خواب بزرگ و پر نور 3 ✅\n\n آپارتمان 115 متری ، خوش نقشه ✅\n\n\n بنام معمار هستی هال پذیرا...	شریعتی

Figure 25: labeled neighborhood.

-

- I could find neighborhood of each adv it is done by LLM api.

- 5 most expensive neighborhood :

```
predicted_neighborhood
2.070588           ابن سینا e+08
2.014925              گاندی e+08
1.950000          یوسف اباد e+08
      \n\n1.916667   جهرانپارس e+08
         1.903614        کیشا e+08
Name: pricepermeter, dtype: float64
```

Figure 26: richest neighborhoods.

- 5 most expensive houses :

	title	totalprice_int	predicted_neighborhood
4414	126متر همیلا/جنب پارک نهج‌البلاغه/آکواریوم نور	1.940000e+10	همیلا
5120	خ دربند150 متر 3خ فول مشاعات 10 ساله ویودار نورگیر	1.930000e+10	دربند
4708	متر/3خواب/سازمان برنامه شمالی/نوساز120	1.930000e+10	سرتیپ شفاهی
5529	آپارتمان 133 متری 3 خوابه / صادقیه / کلیدنخورده	1.928500e+10	صادقیه
4757	متر 3 خواب تکواحدی/نوساز/شهران 153	1.920000e+10	ناصری

Figure 27: most expensive houses

- check it in my ipynb file please I did not append its figure because it was large I did it with df.sample().