PREDICT HOUSE PRICE IN HANOI

Group 10
Lecturer:
Vo Ngoc Yen Nhi
Semester 3, 2021
09/01/2022

TEAM

Nguyen Dang Huynh Chau

Member

Tran Ngoc Anh Thu

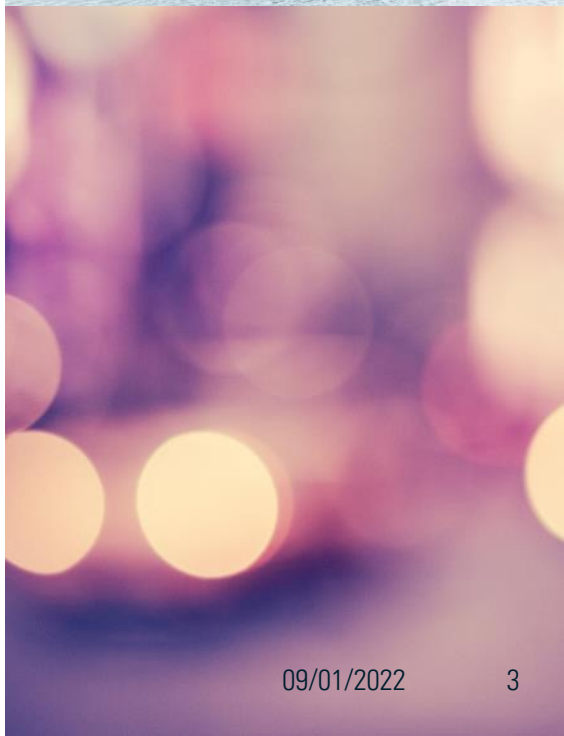
Leader

Ho Le Minh Thach

Member

AGENDA

- Introduce our problem and data set
- Describe data preparation step
- State the hypothesis
- Model Process
- Demo of Model Deployment and Dash
- Recommendation



PROJECT TARGET

- Explore the influences of some factors on sale prices.
- Develop a model for predicting the price range of houses in Hanoi

INTRODUCE OF DATASET

- Vietnam Housing Dataset (Hanoi) available on Kaggle.com.
- Real estate listings posted on Alo Nhà Đất, crawled in August 2020 by Le Anh Duc

	Ngày	Địa chỉ	Quận	Huyện	Loại hình nhà ở	Giấy tờ pháp lý	Số tầng	Số phòng ngủ	Diện tích	Dài	Rộng	Giá/m2
0	2020-08-05	Đường Hoàng Quốc Việt, Phường Nghĩa Đô, Quận C...	Quận Cầu Giấy	Phường Nghĩa Đô	Nhà ngõ, hẻm	Đã có sổ	4	5 phòng	46 m²	NaN	NaN	86,96 triệu/m²
1	2020-08-05	Đường Kim Giang, Phường Kim Giang, Quận Thanh ...	Quận Thanh Xuân	Phường Kim Giang	Nhà mặt phố, mặt tiền	NaN	NaN	3 phòng	37 m²	NaN	NaN	116,22 triệu/m²
2	2020-08-05	phố minh khai, Phường Minh Khai, Quận Hai Bà T...	Quận Hai Bà Trưng	Phường Minh Khai	Nhà ngõ, hẻm	Đã có sổ	4	4 phòng	40 m²	10 m	4 m	65 triệu/m²

DATA PREPARATION STEP

STEP OF CLEANING:

- Changing Datatypes
- Translation Step
- Homogenous Data
- Fill Missing Value
- Simplify Categories
- Whitespaces checks
- Typos checks
- Sanity Checks

CHANGING DATATYPES

Continuous numerical values: ‘Area’, ‘Length’, ‘Width’, and ‘Price’ columns

Remove its unit and format its value

Convert to Float datatype

Remove prefix: ‘District’, ‘Ward’, ‘No_floor’ and ‘No_room’

TRANSLATE STEP

Translate column ‘House_type’, ‘Legal_documents’. ‘No_floor’, and ‘No_room’.

‘House_type’ will have: ‘byroad’, ‘street_house’, ‘villa’, and ‘townhouse’

‘Legal_documents’ will have: ‘available’, ‘waiting’, and ‘others’

‘No_floor’ and ‘No_bedroom’ will need to replace value 'Nhiều hơn 10’ to 'greater_than_10'

DATAFRAME

	Address	District	Ward	House_type	Legal_documents	No_floor	No_bedroom	Area	Length	Width	Price	Day_Of_Week	Month	Year
0	Đường Hoàng Quốc Việt, Phường Nghĩa Đô, Quận C...	Quận Cầu Giấy	Phường Nghĩa Đô	Nhà ngõ, hẻm	Đã có sổ	4	5 phòng	46 m²	NaN	NaN	86,96 triệu/m²	Wednesday	08	2020
1	Đường Kim Giang, Phường Kim Giang, Quận Thanh ...	Quận Thanh Xuân	Phường Kim Giang	Nhà mặt phố, mặt tiền	NaN	NaN	3 phòng	37 m²	NaN	NaN	116,22 triệu/m²	Wednesday	08	2020
2	phố minh khai, Phường Minh Khai, Quân Hai Bà T...	Quận Hai Bà Trưng	Phường Minh Khai	Nhà ngõ, hẻm	Đã có sổ	4	4 phòng	40 m²	10 m	4 m	65 triệu/m²	Wednesday	08	2020

	Address	District	Ward	House_type	Legal_documents	No_floor	No_bedroom	Area	Length	Width	Price	Day_Of_Week	Month	Year
0	Đường Hoàng Quốc Việt, Phường Nghĩa Đô, Quận C...	Cầu Giấy	Nghĩa Đô	byroad	available	4	5	46.0	NaN	NaN	86.96	Wednesday	08	2020
1	Đường Kim Giang, Phường Kim Giang, Quận Thanh ...	Thanh Xuân	Kim Giang	street_house	NaN	NaN	3	37.0	NaN	NaN	116.22	Wednesday	08	2020
2	phố minh khai, Phường Minh Khai, Quận Hai Bà T...	Hai Bà Trưng	Minh Khai	byroad	available	4	4	40.0	10.0	4.0	65.00	Wednesday	08	2020

DATA HOMOGENOUS

Capitalize all the string value so that it will be easier to process the string and encoding process in the future step.

	Address	District	Ward	House_type	Legal_documents	No_floor	No_bedroom	Area	Length	Width	Price	Day_Of_Week	Month	Year
0	ĐƯỜNG HOÀNG QUỐC VIỆT, PHƯỜNG NGHĨA ĐÔ, QUẬN C...	CẦU GIẤY	NGHĨA ĐÔ	BYROAD	AVAILABLE	4	5	46.0	NaN	NaN	86.96	WEDNESDAY	08	2020
1	ĐƯỜNG KIM GIANG, PHƯỜNG KIM GIANG, QUẬN THANH ...	THANH XUÂN	KIM GIANG	STREET_HOUSE	NaN	NaN	3	37.0	NaN	NaN	116.22	WEDNESDAY	08	2020
2	PHỐ MINH KHAI, PHƯỜNG MINH KHAI, QUẬN HAI BÀ T...	HAI BÀ TRƯNG	MINH KHAI	BYROAD	AVAILABLE	4	4	40.0	10.0	4.0	65.00	WEDNESDAY	08	2020

FILL MISSING VALUE

There are a few approaches that need to be considered for each case including:

- Guess the missingness type
- Drop missing value by deleting rows
- Drop missing value by deleting columns
- Fill the missing data by mean, mode, or median

```
df['House_type'].value_counts(dropna=False)
```

```
BYROAD           62535
STREET_HOUSE     17095
TOWNHOUSE         1881
VILLA              952
NaN                 31
Name: House_type, dtype: Int64
```

```
df['Legal_documents'].value_counts(dropna=False)
```

```
AVAILABLE    52912
NaN          28886
WAITING        356
OTHERS         340
Name: Legal_documents, dtype: Int64
```

FILL MISSING VALUE

```
# check if there is any missing value in ward
df[['Address', 'Ward', 'District']][df['Ward'].isna()]
```

	Address	Ward	District
174	<NA>	<NA>	NAM TỪ LIÊM
324	ĐƯỜNG AN DƯƠNG VƯƠNG, QUẬN TÂY HỒ, HÀ NỘI	<NA>	TÂY HỒ
741	CẦU KHÊ TANG, QUẬN HÀ ĐÔNG, HÀ NỘI	<NA>	HÀ ĐÔNG

```
# check the No_bedroom
df['No_bedroom'].value_counts(dropna=False)
```

```
4                  29069
3                  27162
5                   7924
2                   7330
6                   6461
1                   1388
8                    938
GREATER_THAN_10      869
7                    678
10                   354
9                    283
NaN                   38
Name: No_bedroom, dtype: Int64
```

```
# fill the missing value of area base on the median of area in district
df['Area'] = df.groupby('District')['Area'].apply(lambda x: x.fillna(x.median()))
# fill the missing value of price base on the median of price in district
df['Price'] = df.groupby('District')['Price'].apply(lambda x: x.fillna(x.median()))

# fill the missing value of length base on the median of length of area
df['Length'] = df.groupby('Area')['Length'].apply(lambda x: x.fillna(x.median()))
# fill the remaining missing value base on the mean of the entire column
df['Length'].fillna(df['Length'].mean(), inplace=True)
# fill the missing value of width base on the median of width of area
df['Width'] = df.groupby('Area')['Width'].apply(lambda x: x.fillna(x.median()))
# fill the remaining missing value base on the mean of the entire column
df['Width'].fillna(df['Width'].mean(), inplace=True)
```

```python
# define a function to feed into the lambda
def No_floor_count(value):
    if value == '1' or value == '2' or value == '3' \
            or value == '4' or value == '5' or value == '6' \
            or value == '7' or value == '8' or value == '9' or value == '10':
        return value
    else:
        return 'GREATER_THAN_10'

# make any floor greater than 10 to be in category 'GREATER_THAN_10'
df['No_floor'] = df['No_floor'].map(lambda n: No_floor_count(n))
# check if the above operation is success or not
df['No_floor'].value_counts(dropna=False)
```

```
5                 61865
4                 12278
3                  3619
6                  2119
2                  1028
1                   636
7                   597
8                   188
9                    88
GREATER_THAN_10      40
10                   36
Name: No_floor, dtype: int64
```

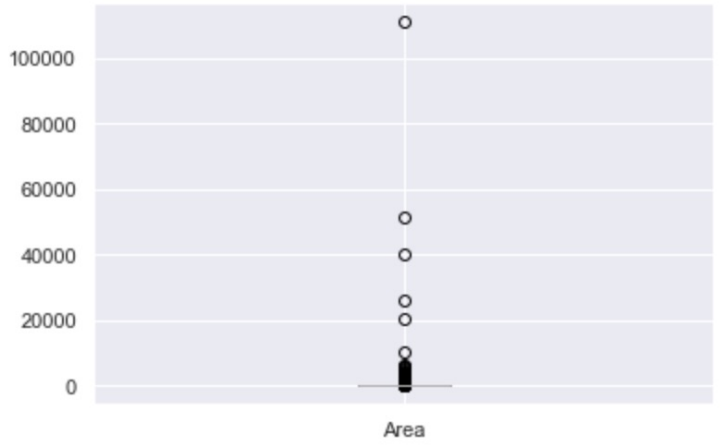
FILL MISSING VALUE

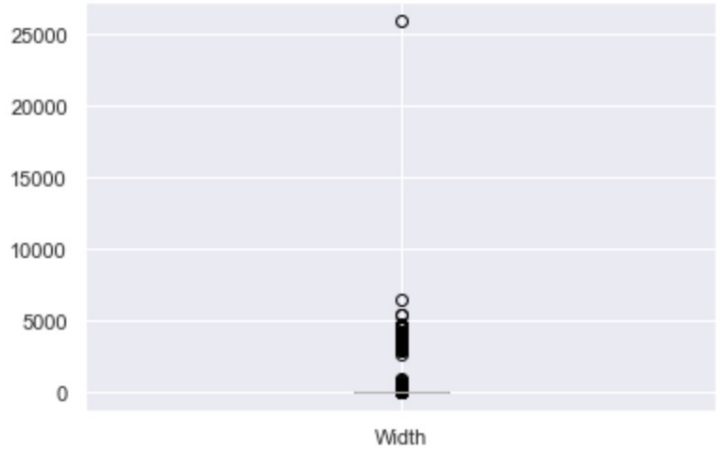
EXTRA WHITESPACES

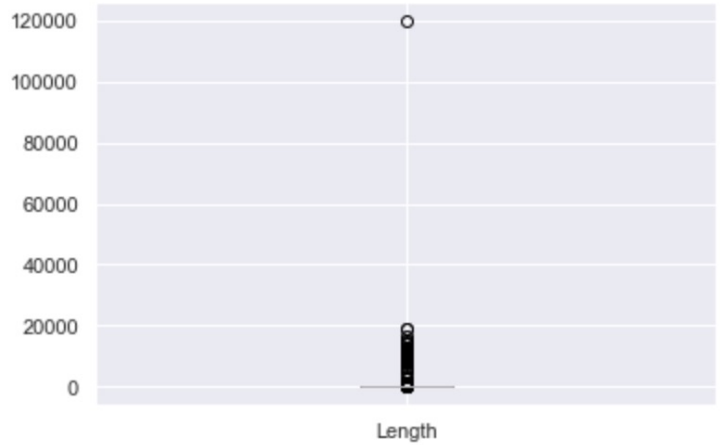
Stripping whitespaces at the beginning of the string value, and the back of the string value

SANITY CHECK

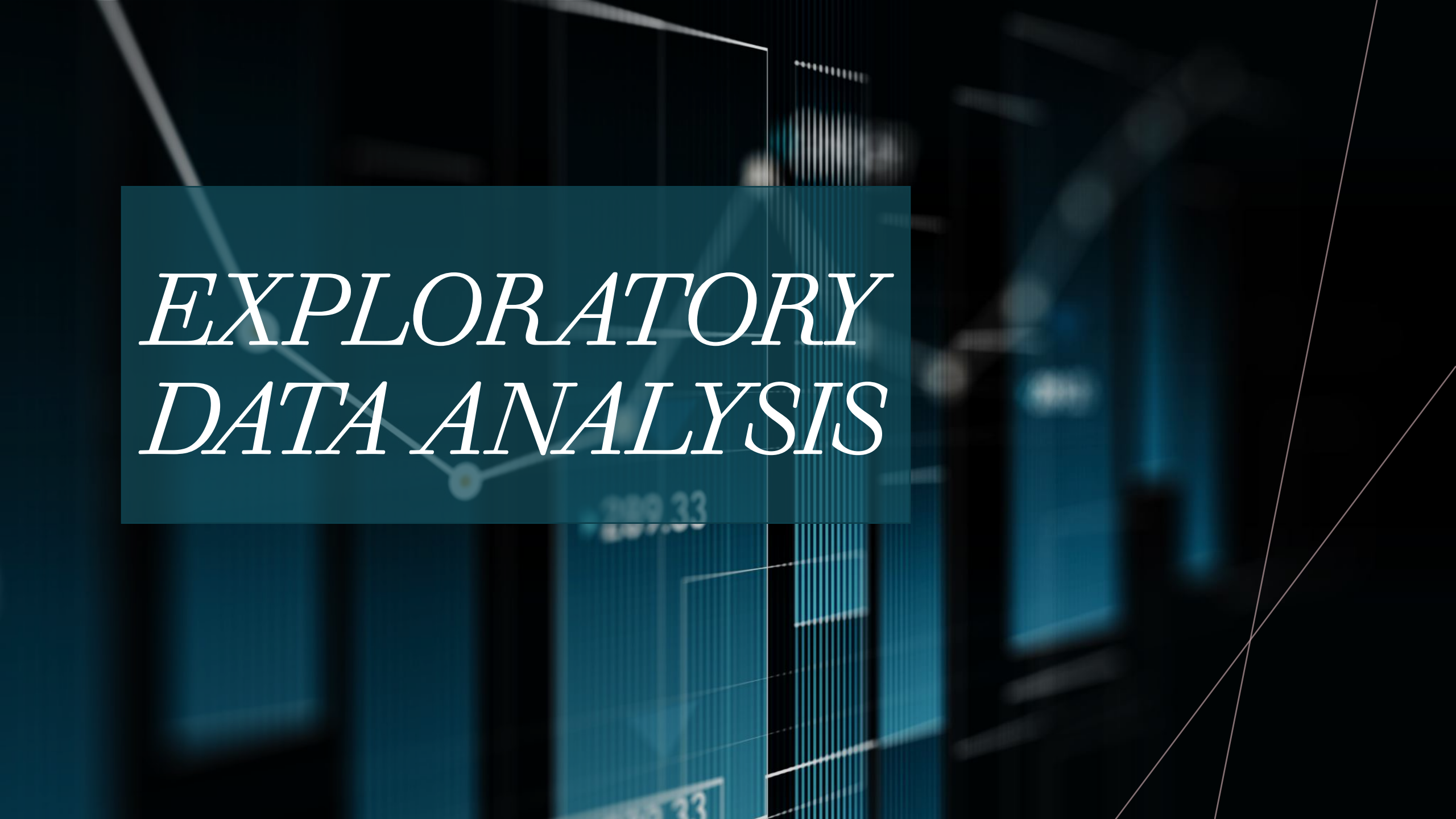
Outlier	Area, Length and Width	Length and Width
Remove outlier by fill those value with mean.	The area must be less than the product of length and width.	Width and length equal or greater than 2.4 meter, else it needs to have the legal documents available.

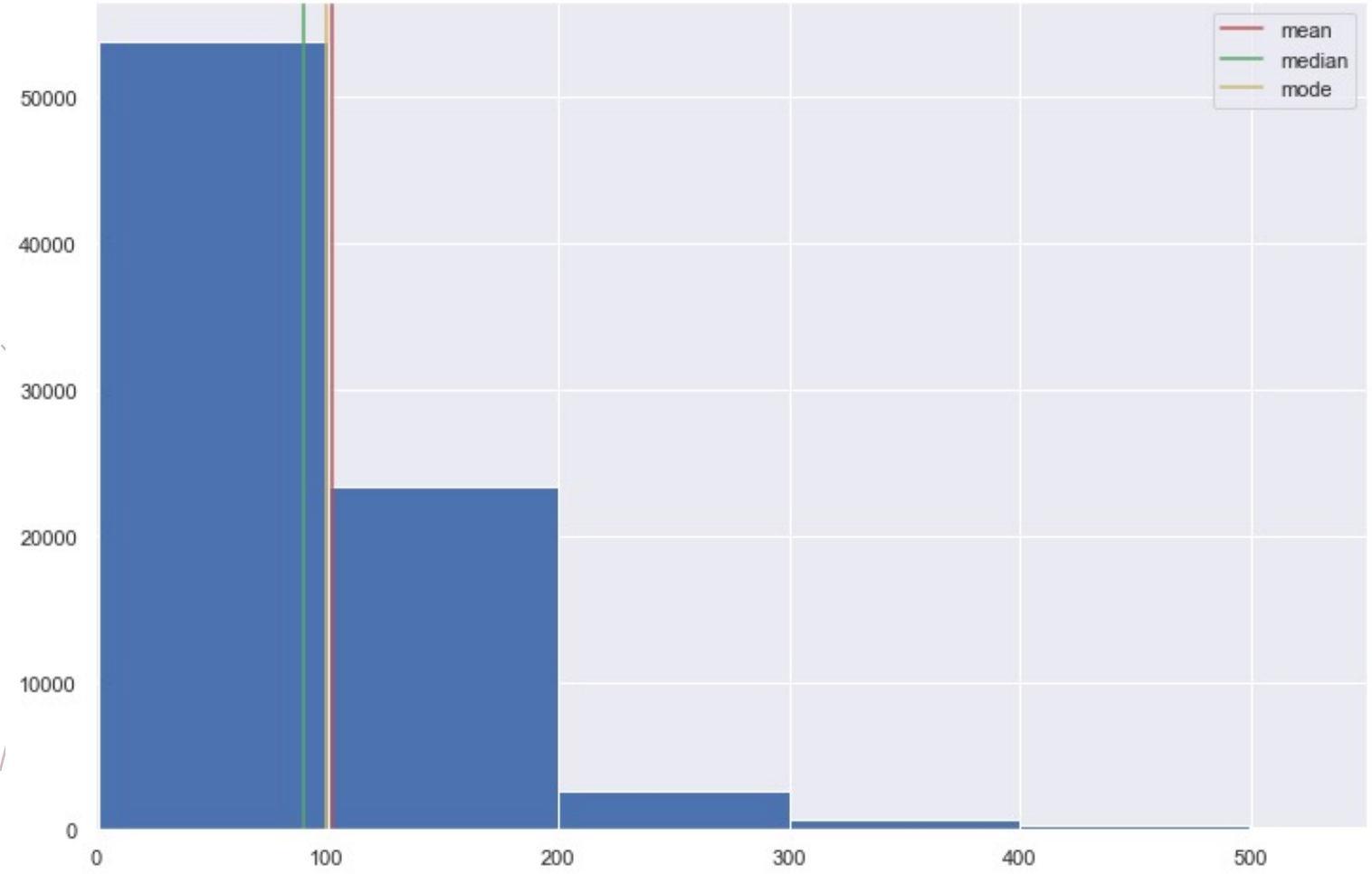






SANITY CHECK


EXPLORATORY
DATA ANALYSIS



DESCRIPTIVE STATISTICS FOR VARIABILITY

- Mean:
- SKEWED TO THE RIGHT

STATE HYPOTHESIS

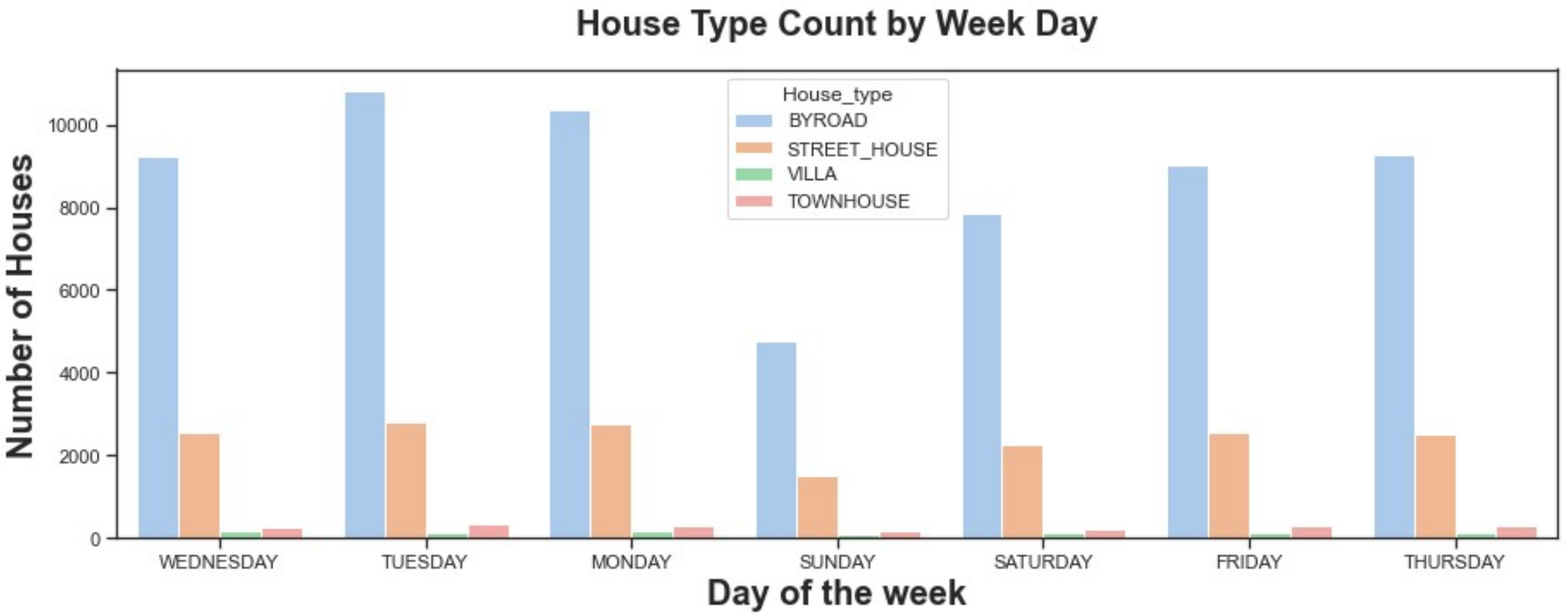
Are house sold more in the weekday than in the weekend?

Is houses price on the weekend is lower than in the weekday?

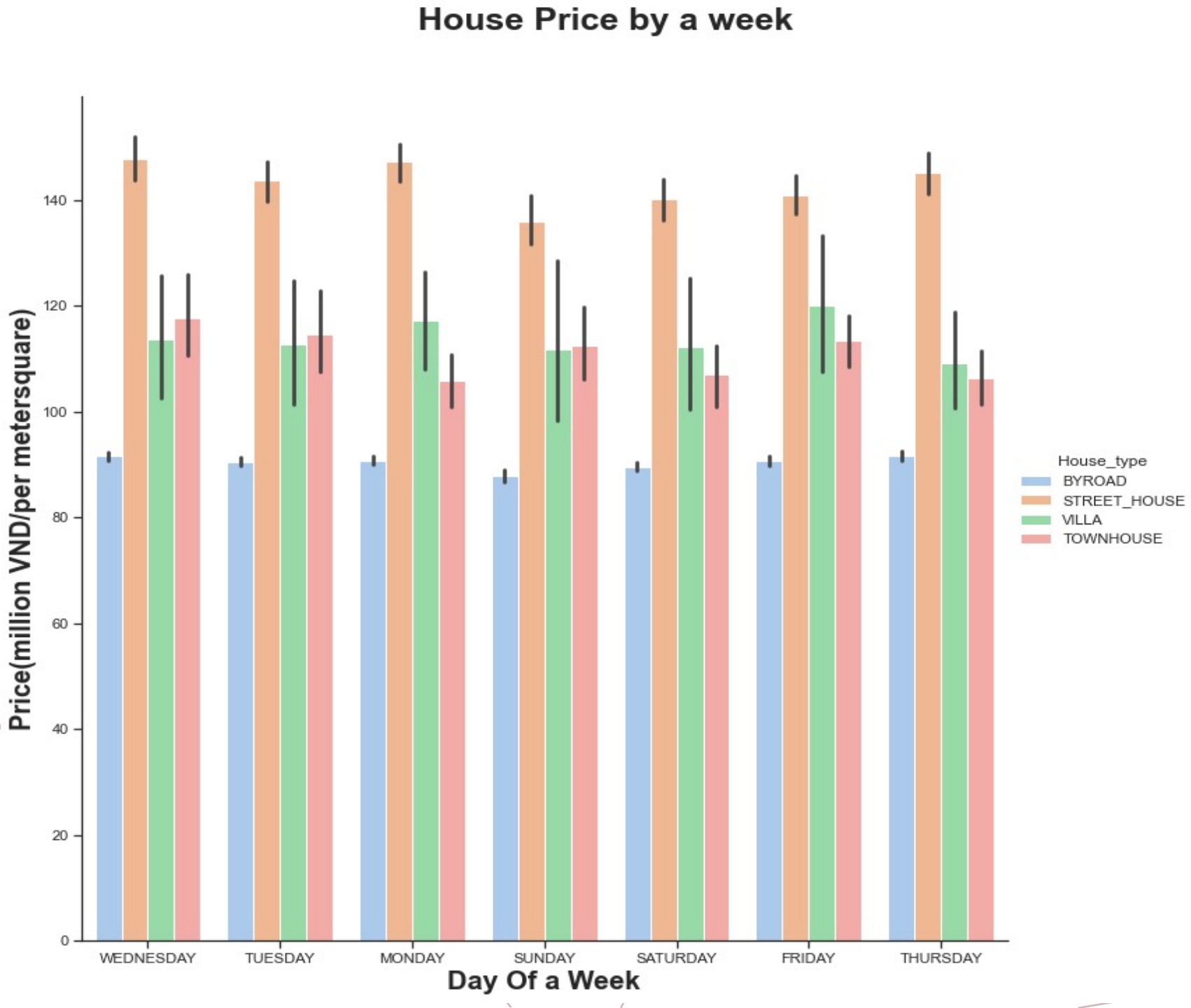
Are houses sold more in the end of the year?

Relationship with other columns

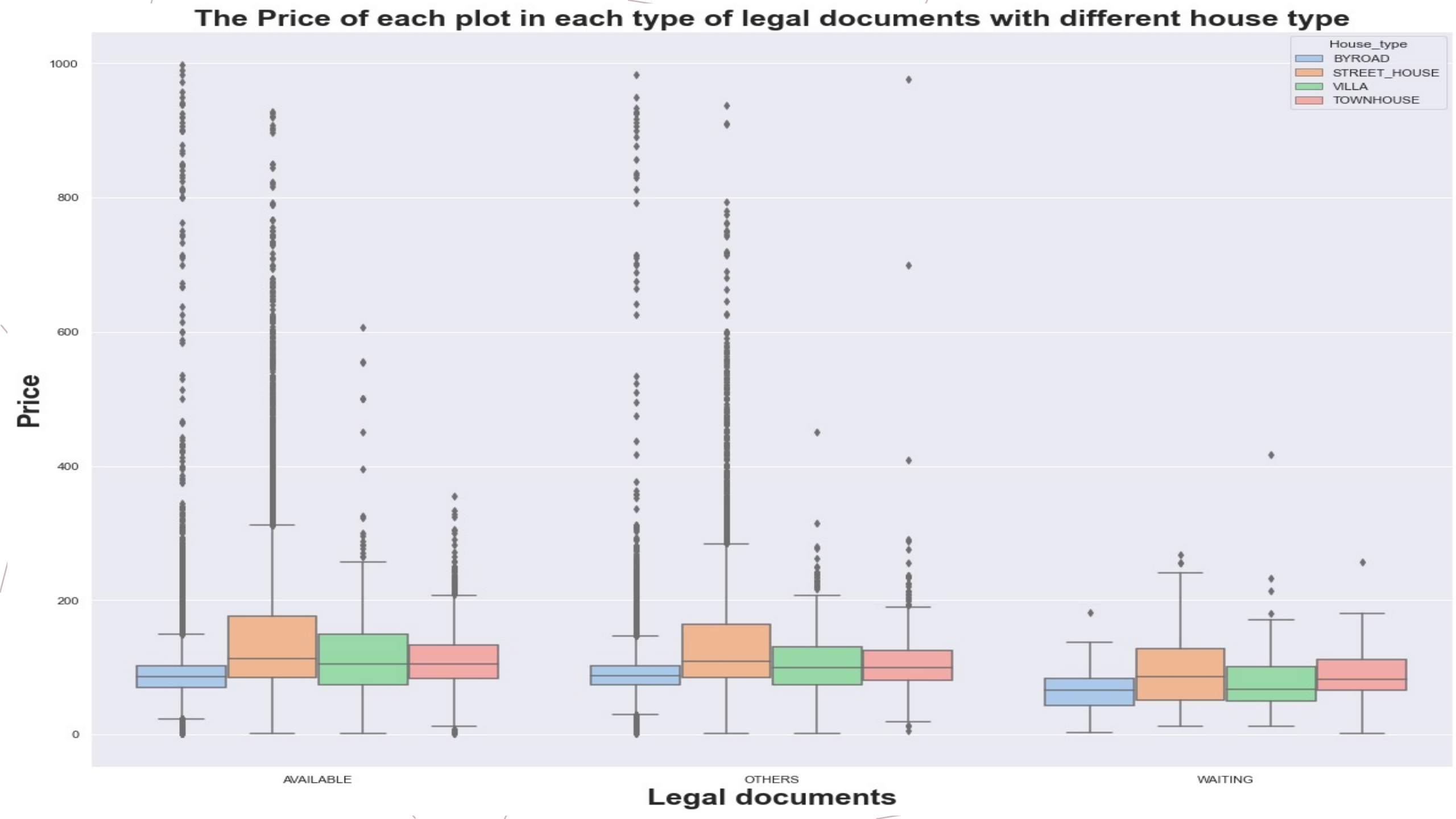
THE FREQUENCY OF PRICE RANGE



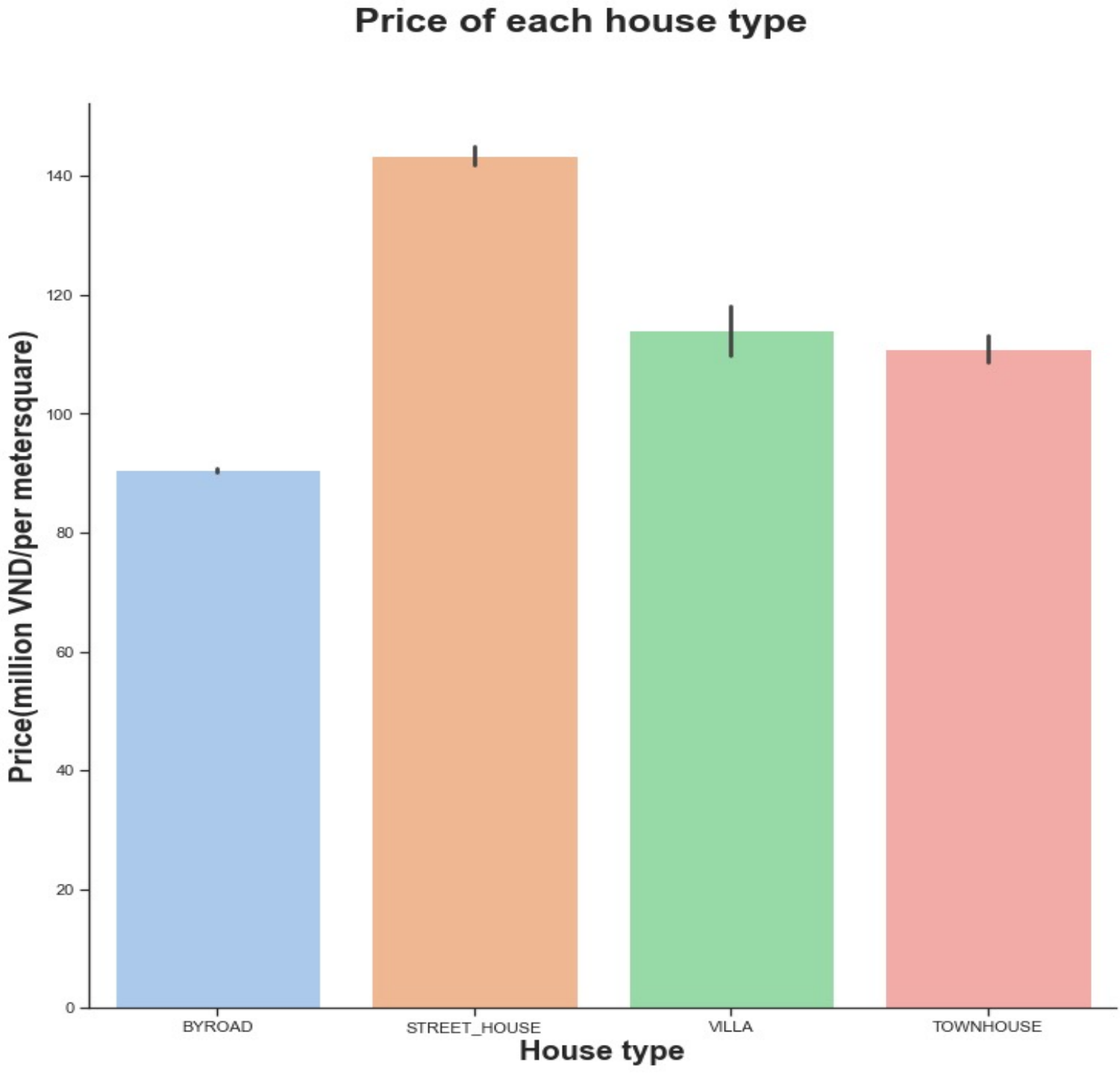
HOUSE TYPE COUNT IN A WEEK

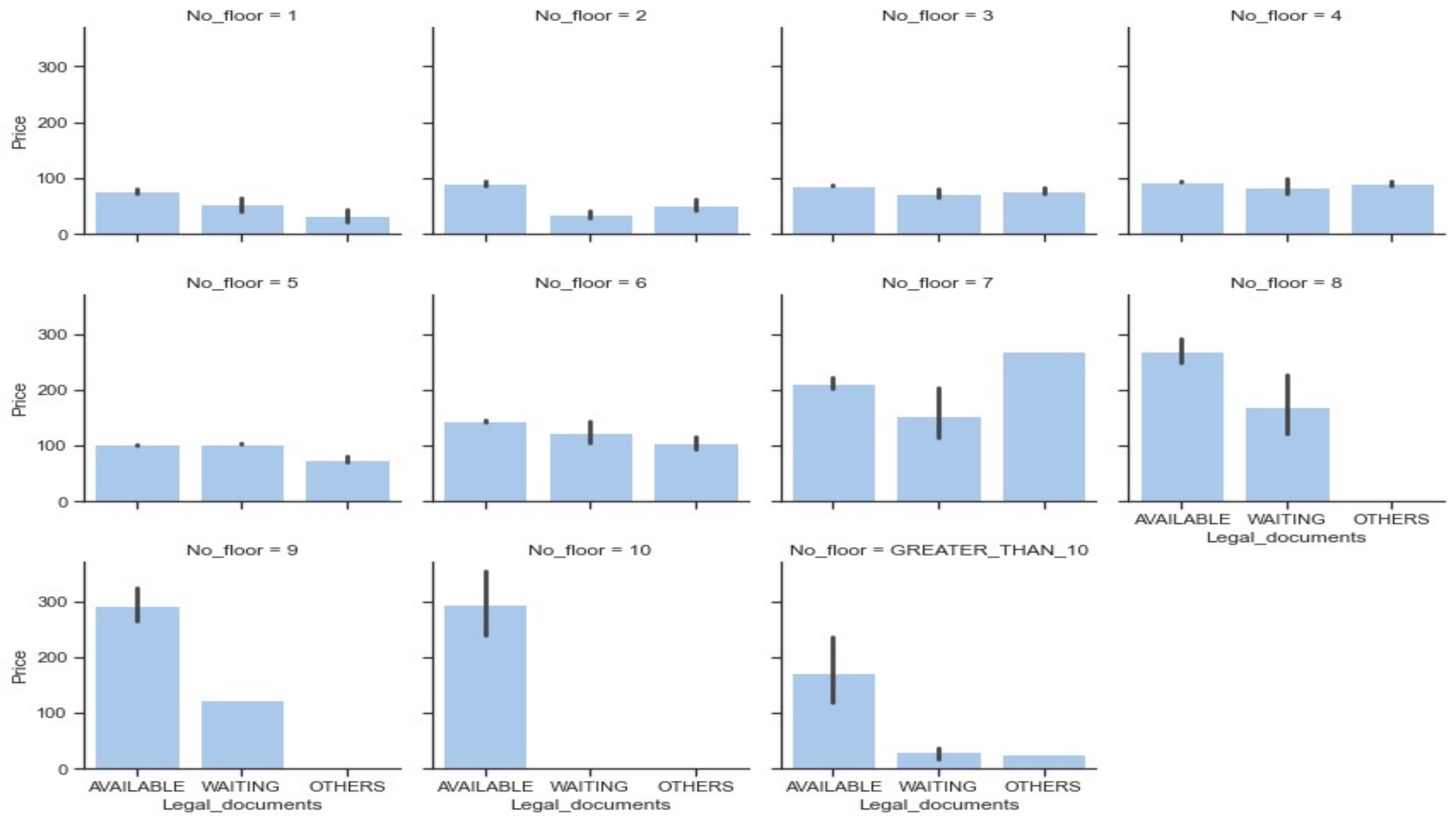


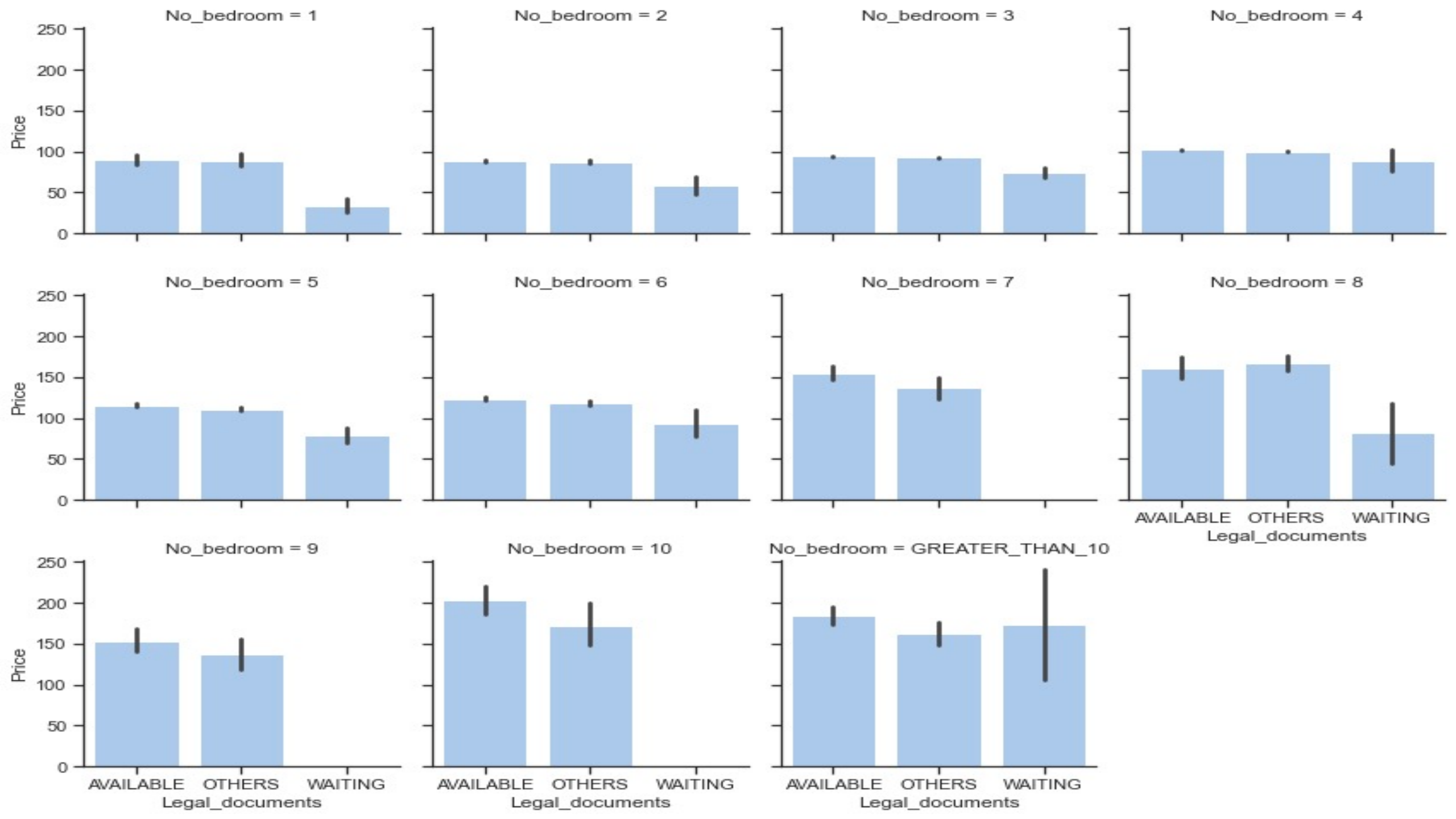
HOUSE PRICE IN A WEEK

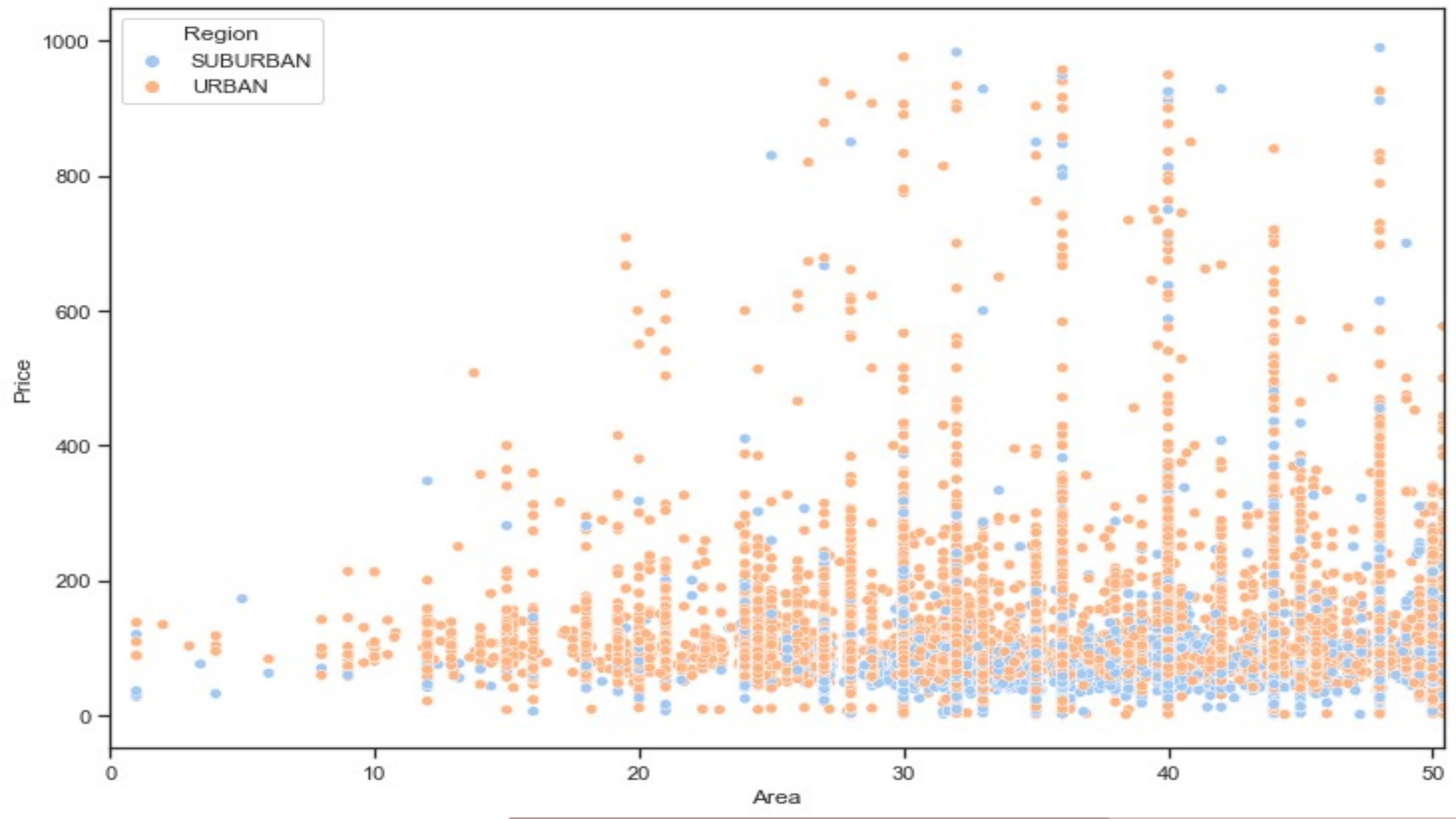

The Price of each plot in each type of legal documents with different house type
House_type
BYROAD
STREET_HOUSE
VILLA
TOWNHOUSE
Price
1000
800
600
400
200
0
AVAILABLE
OTHERS
WAITING
Legal documents

PRICE OF EACH HOUSE TYPE




No_floor = 1
No_floor = 2
No_floor = 3
No_floor = 4
No_floor = 5
No_floor = 6
No_floor = 7
No_floor = 8
No_floor = 9
No_floor = 10
No_floor = GREATER_THAN_10
Price
0
100
200
300
AVAILABLE
WAITING
OTHERS
Legal_documents


No_bedroom = 1
No_bedroom = 2
No_bedroom = 3
No_bedroom = 4
No_bedroom = 5
No_bedroom = 6
No_bedroom = 7
No_bedroom = 8
No_bedroom = 9
No_bedroom = 10
No_bedroom = GREATER_THAN_10
Price
0
50
100
150
200
250
AVAILABLE
OTHERS
WAITING
Legal_documents


Region
SUBURBAN
URBAN
1000
800
600
400
200
0
Price
0
10
20
30
40
50
Area

House Price Range in Hanoi Prediction

Multiclass Classification

Contents

Project Overview

Goal 1

Provide context, goals and objectives for the project.

Goal 2

Provide context, goals and objectives for the project.

Goal 3

Provide context, goals and objectives for the project.

Goal 4

Provide context, goals and objectives for the project.

Future Improvement

1 Step

Transform the code in Jupyter notebooks into production code

2 Step

Including tests, logging and OOP

3 Step

Deploy into a realistic production environment

4 Step

use docker to control software and model versions

5 Step

Add a CI/CD layer

Models	F1	Note	Result
Logistic	0.68	solver='lbfgs'	same on train
Random Forest	0.93	StandardScaler	Overfitting
K-Nearest Neighbors + GridSearchCV	0.99	'n_neighbors': 5, 'weights': 'uniform'	Best

Thank you

LINK OF THE PRESENTATION

HTTPS://YOUTU.BE/AC0HD0ER2H4