**SUMMARY**

**Group Two**

**Jianuo Wang, Leyu Pan, Weizheng Liang, Jia Sun**

**Abstract—With the continuous development of China's economy, Guangzhou, as one of the four most powerful cities in China (Peking, Shanghai, Guangzhou, Shenzhen), has attracted many people to go and settle down there. This makes housing demand in Guangzhou continue to rise, and also has led to higher housing prices in recent years. Because we are not able to count the number of new-house transactions, we start with the second-hand housing in Guangzhou, looking for something that affects house prices.**

**This paper starts from retrieving and visualizing the data. And by using ‘TF-IDF’ and ‘K-Means’ in Scikit-Learn, we found that the demand for housing type had a great impact on the price, after which we used Google Charts to make the results clearer.**

**Also, we used Logistic regression in SK-learn to analyze the page views and the price difference between the listed price and the last recorded price. We build a practice regression model.**

**Index Terms— second-hand house, house price，house type，page view，purchase price**

1. **INTRODUCTION**

In some cities, especially the big cities, the downtown area is almost built up, there are fewer new communities, and it is easier to buy new houses in the suburbs. Therefore, many people turn to consider about buying the second-hand houses.

We chose to study the information of second-hand housing transactions. We’ve found that the price of second-hand houses can vary considerably even in the same area. We attempted to compare some factors that may influence the price. Also, among all the data we've retrieved, we thought that the difference between the last recorded price and listed price may be relative to page views, so we tried to figure out it by using regression.

1. **DATA PREPARATION**
   1. **Crawl data from the website**

For the project of our final work, we need to scrape the data of second-hand housing from the internet(‘https://gz.ke.com/chengjiao/’). We wrote a program to access the data by using python and several python libs, such as ***requests, threadingpool,*** and ***pandas.***

First of all, we need to connect pycharm to mongodb to make sure it’s ready to store the data for future interpretation. For this, we defined a mongodb class and overload methods ***insert\_one*** to ease the storage for us. Then the ***init*** method integrates the two methods (***insert\_one*** and ***\_collection***) together to store the value.



Secondly, we start writing the main body of the data scraping. 14 methods are included in the body (***req\_get, filter\_key, filter, make\_deal\_list, \_detail\_download\_image, \_detail\_base\_info, \_detail\_msg, \_detail\_agent, \_xiaoqu, make\_deal\_detail, thread\_task, thredef thread\_success, execute***).

The ***req\_get*** method is defined for solving the network problem. It helps restrict the retry account so that the whole project would be in an infinite error cycle when our internet connection is not stable.



Then the ***filter*** and ***filter\_key*** are the key methods to solve the problem of limiting data. For this website, it only posts 100 pages each time, so we cannot get enough data for further analyzing. Therefore, we use these two methods to select features (price, house type, and area of the house) to access different listings.

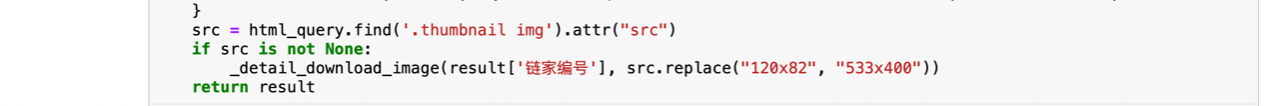
Moreover, the ***make\_deal\_list*** method is to build up the connection with the website and return the number of total pages and the current pages we have already accessed to.



After accessing the website successfully, we can get the raw data. Then we change the structure of each data for easier interpretation. The following five methods: ***\_detail\_download\_image, \_detail\_base\_info, \_detail\_msg, \_detail\_agent, \_xiaoqu*** is to deal with the data, thereby making the data more clearly visible. Each of them is responded for a certain part. The one ***\_detail\_download\_image*** is to download the image of every house in the listings to help we find the features of the housing in the future. The ***\_detail\_msg*** is for the basic information like the price, the number of days between posting on the website and successful deal, the bargain times, the number of people who are interested in, and the number of different people who skim the house. We replace the unit and title for each data and only leave the quantitative data. Next, the ***\_detail\_agent*** is used to get the name of local agent who is dealing with the house, and we print a “agent name” with id of agent to show clearer about the agent when we are scraping the data. Finally, the ***\_xiaoqu*** method helps us find the geographic location of each house with latitude, longitude, and district name from the primitive data.



Then we come to the first main part that we integrate all the basic information together in a method called ***make\_deal\_detail***. In this case, we can store the data either in excel document or mongodb in a way that we can see every features of the data are assigned together in order. All the 33 features are all store in a python dictionary called ‘result’.



Then in the threading pool part, we define a method: ***thread\_task***,which is used to use 100 thread to increase the productivity of the whole program and sperate the work by the districts of Guangzhou. Also, it will store the dictionary with data into the mongodb by using the mongodb class define earlier in the program.

Finally, the method ***execute*** is defined. The filter series of method is included to select the features to get the whole number of pages by using 4 for loops. Within the loop, data dealing part is working to handle the data and store the data in the data base. Then we can have our database in mongodb and excel.



* 1. **Data process and Folium**

Now we are about to deal with these data to get the required parts. First of all, we tried to extract the information we need (the name of the district, price, longitude and latitude). Then we put them together sorting by the name of the districts. The longitude and latitude were in the same column, so we use ‘split’ to divide the longitude and latitude into two columns and drop the original column. Then we got a data frame with four columns. The core code and data frame are shown in the Fig.2.2.1.

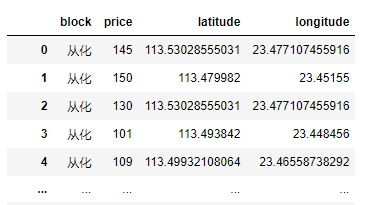
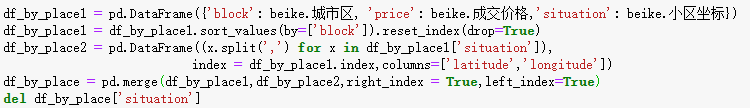


Fig.2.2.1 Data processing and results

Then we used ‘Folium’ for visualization. We marked the location of the houses in each district on the map, and changed the original icon. When putting the cursor over one of the markers, the price of the house will be shown. The Fig.2.2.2 shows the house information in Conghua District.

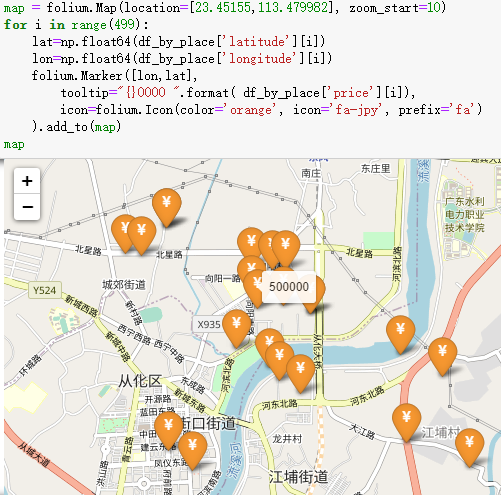


Fig.2.2.2 House locations and prices in Conghua district

* 1. **Data visualization by Google Charts**

We used Google charts to visualize and analyze the basic data, and then got some ideas about the factors that might affect the house prices.

**2.3.1 Total number of houses sold in different districts -- Doughnut Chart**

First, we traversed the data, and counted the number of each district, as shown in Fig.2.3.1.1. The Doughnut Chart is used to visualize the data of the total prices. From Fig.2.3.1.2, we can clearly see that Tianhe District has the largest percentage of the total number of house purchases, accounting for 18.2% of the total number, while Conghua district has the least, accounting for only 0.7%.

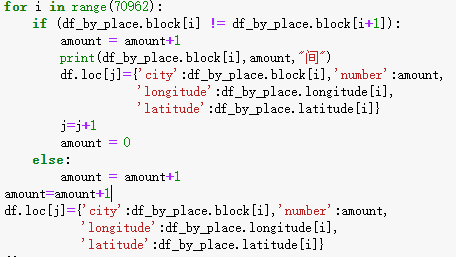


Fig.2.3.1.1 Count the number of houses

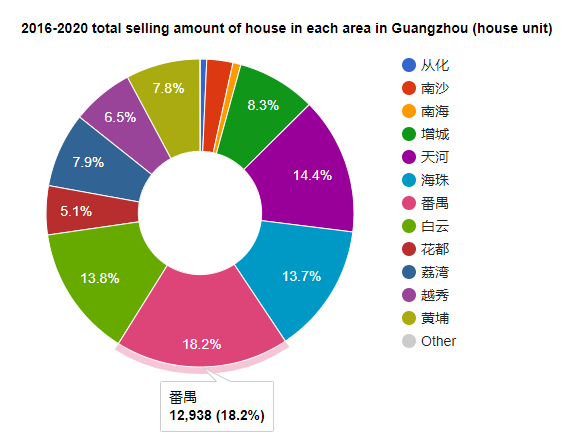


Fig.2.3.1.2 Visualize the number of houses

**2.3.2 Total selling prices of different areas -- Pie Chart**

Next, we counted the total selling prices in each district, and used pie chart to visualize the data, as shown in Figure 2.3.2.1. From the Fig.2.3.2.2, we can see that Tianhe District has the highest total house sales price, accounting for 22.5% of the whole Guangzhou City, while Conghua district still has the lowest total house-sale price, only for 0.4%.

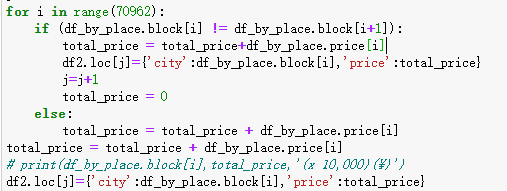


Fig.2.3.1.1 Count the total prices of houses

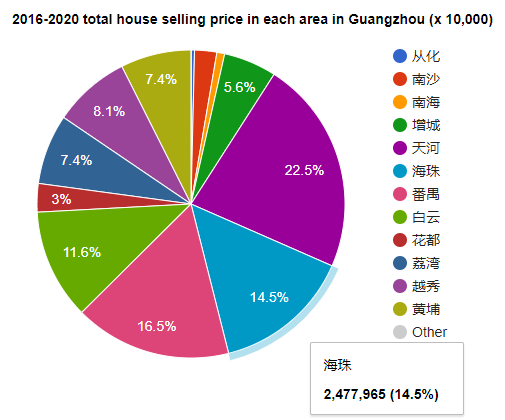


Fig.2.3.1.2 Visualize the total prices of houses

**2.3.3 Total house prices in different seasons - Bar Chart**

Next, we counted the total prices of the houses in different seasons of 2016-2020, and visualized the data with a bar chart. The core code in the Fig.2.3.3.1 takes 2016 as an example. From the bar chart in Fig.2.3.3.2, we can see that in the spring of 2016, the number of houses transaction is the least, only 348 million, while in the winter of 2020, the number is the largest, reaching 2650 million. What causes such a big difference in turnover? We'll explain that in the next section.

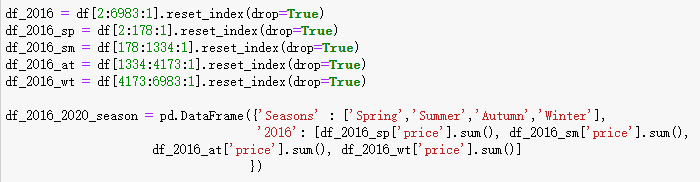


Fig.2.3.3.1 Process data for 2016

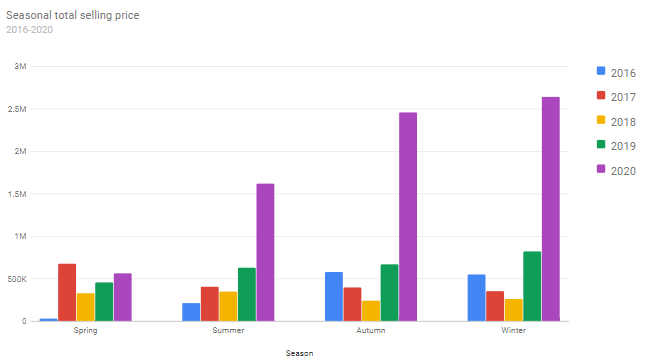
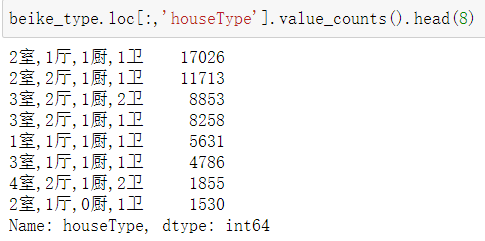
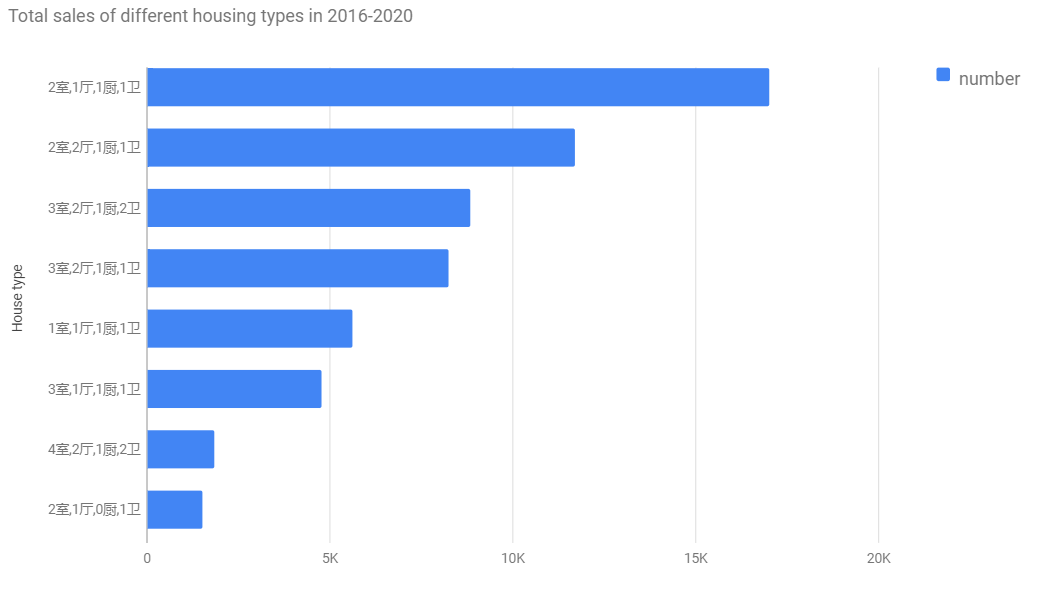


Fig.2.3.3.2 Total prices of the houses in different quarters of 2016-2020

**2.3.4 Total sales of different types of houses -- bar chart**

We made statistics on the total sales of different house types from 2016 to 2020, as shown in Fig.2.3.4.1, and visualized the data with bar chart. It can be seen from Fig.2.3.4.2 that the demand of 2 bedrooms is the largest, followed by 3 bedrooms.

Fig.2.3.4.1 count the total number of different number of houses

Fig.2.3.4.2 Total sales of different housing types in 20165-2020

Different types of house represent the housing needs of different groups of people, and the demand varied. At present, the majority of people who have housing needs are newlyweds (two adults) and three members of a family (two adults and one child), followed by five members of a family (two parents, two couples, one child), and a family of many people (2 adults and more than one child), and it’s the reason that leads to this difference.

* 1. **Raise the questions**

As it can be seen from the figures above, Panyu District has the largest amount of house transactions, but only ranks the second place for total sales, with a difference of 6% from Tianhe District, which has reached 10 billion. How does such a big difference come about? We supposed that it might have something to do with the floor space. The larger the area, the higher the selling price. We counted the data and found that there are many apartment buildings in Panyu District with one bedroom, one living room, one kitchen and one bathroom dominated, while two bedrooms, one living room, one kitchen and one bathroom is the most common housing type in Tianhe District. The housing type in Tianhe District determines that the price of the house there is higher than that in Panyu, which leads to the phenomenon above.

Apart from the area that houses locate, we guessed the housing type is also relative to the price per square meter of the house, so we discussed this in chapter 3. The data we’ve retrieved contains page views and the last recorded price, which is different from the listed price. To verify our suppose that whether the last recorded price will be influenced by the page views, we wrote a program and discussed it in chapter 4.

1. **Analyze the influence of different types of houses on price by Scikit-Learn**

Since the number of bedrooms, living rooms, kitchens and bathrooms might influence the price of a house, we wrote a program by Scikit-Learn in order to discover the price difference due to the difference in the number of rooms’ kinds.

The first thing we have done is Data Cleaning. In data.csv, there are some information on the sale of parking spaces (‘车位’) in Fig.3.1, so we need to drop those data to ensure that only house sales are kept in data.

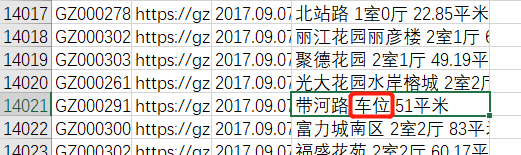
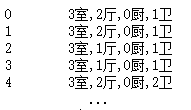


Fig.3.1 It contains the sale data of parking spaces in data.csv

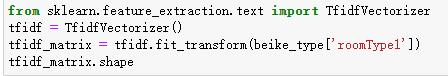
The keyword ‘lambda’ is used to create anonymous functions which can be used wherever function objects are required.[1] Therefore, applying this lambda expression, which can make the data separated by commas, is the way to format the column named ‘房屋户型’ which is shown in Fig.3.2.

Fig.3.2. Use lambda function to insert commas between different type of rooms



The next step was to use TF-IDF and K-Means to give each row a label that can represent the type of the house. TF-IDF stands for Term Frequency Inverse Document Frequency of records. It can be defined as the calculation of how relevant a word in a series or corpus is to a text.[2] Here we use TfidfVectorizer to generate the TF-IDF value directly.[3] in Fig.3.3.

Fig.3.3. Change the words to array by TF-IDF



The array - ‘tfidf\_weight’, contains the weight of each line of ‘houseType’. We use K-Means to cluster this array, which is to divide those house types into 50 parts. In this case, we add the labels of each point to the data frame in Fig.3.4.

Fig.3.4. Add the ‘labels’ to the data frame



The reason we choose to compare the prices of second-hand houses in Yuexiu District is that it’s the place where Guangzhou Municipal Government located in. Then we extract the data we need and format it to form a new data frame which is shown in Fig.3.5.

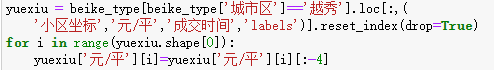


Fig.3.5. Get the required second-hand housing information in Yuexiu District

We obtain the longitude and latitude of the Guangzhou Municipal Government (113.264385, 23.129112) and find that a value difference of 0.01 in latitude or longitude represents 1 kilometer approximately from the website of amap.[4] We choose the houses within 1 kilometer from the government and calculate the average price per square meter of each label. ( Fig.3.6)

After that, we filter the labels that have sold each year and visualize the obtained data.

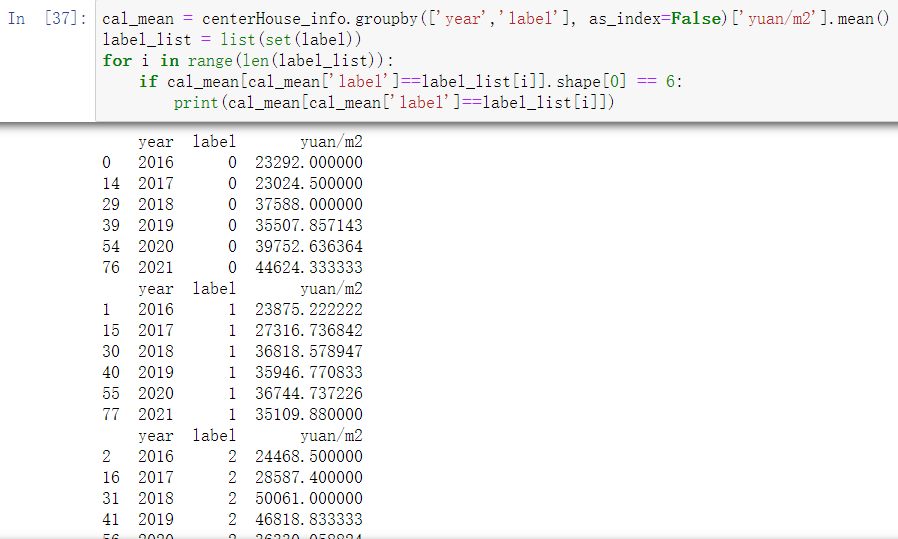
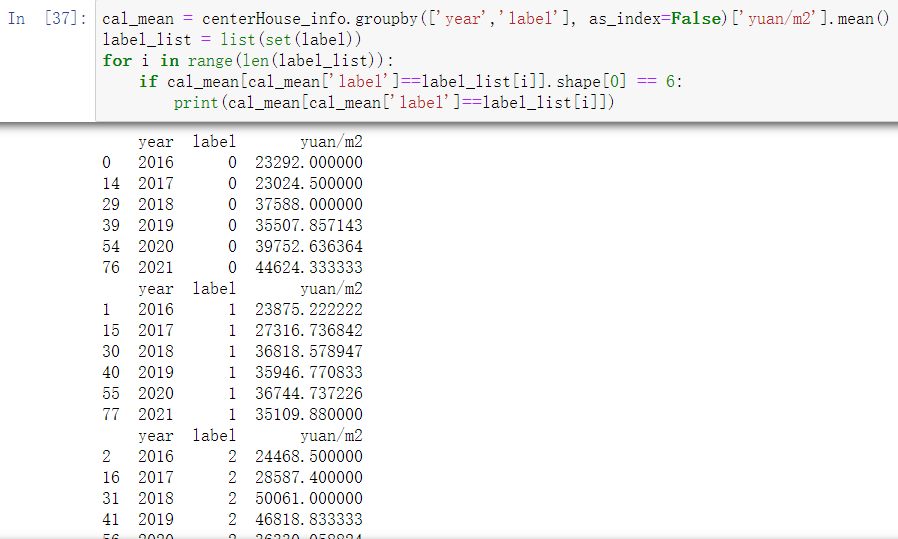


Fig.3.6. Find the houses near the government and calculate mean value of the price



We used a line chart in Fig.3.7 to show the per square meter prices of different labels in time order. We found that the most labels’ prices rose slightly over time. Only label2 and label13 were different, and the prices of label2 changed significantly over time, while label13 opposite to the other labels most time. Then let's have a look at what housing types that label2 and label13 represent.

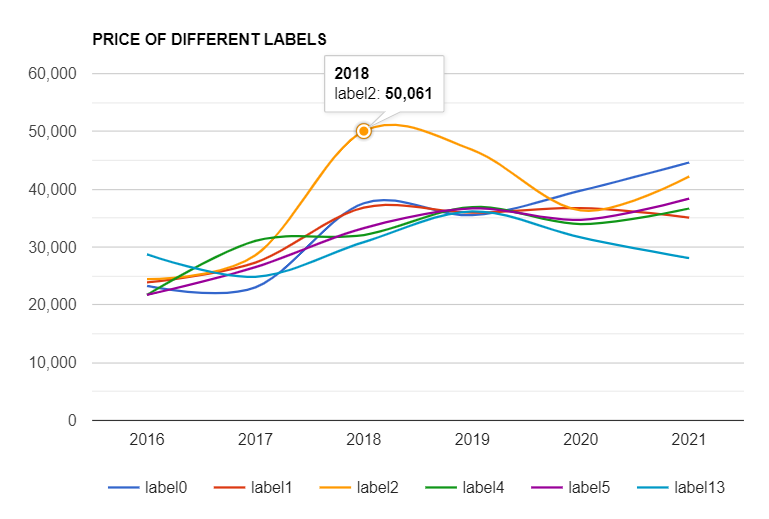


Fig.3.7. Find the houses near the government and calculate mean value of the price

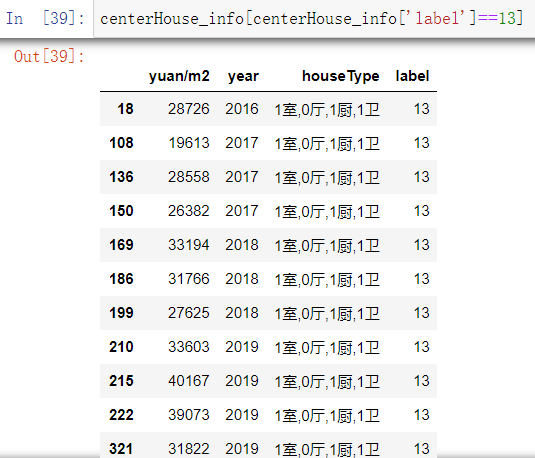
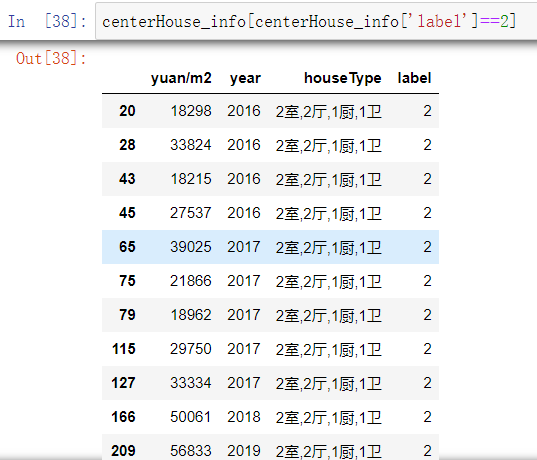


Fig.3.8. House types represented by different labels

We find that the biggest difference between the two types is the number of bedrooms in Fig.3.8. Label2 has two bedrooms and label13 has only one. Therefore, we can infer that one-bedroom houses are not so popular, while houses with two bedrooms are in great demand, so this trend was drove. This can also be confirmed by the sales figure 2.3.4.2 in the data.

1. **A Regression Practice**
   1. **Intro**

This is the summary of building a practice regression model in python using Scikit-Learn. At the beginning of the course, we learned about the basic of Scikit-Learn and have a broad and basic understanding of machine learning. Hence, we decide to use Scikit-Learn in this portion of our final group project.

How to buy a house with a price as low as possible? We noticed the price difference between the posting price and the actual dealing price then raised this question. (Fig.4.1)

Dealing price Posting price

| |

表格

描述已自动生成

Fig.4.1. Raw data of Guangzhou housing price

|

Page views

Moreover, I think that there is a relationship between how much less the dealing price is and number of times people view the house. I decide to build a regression model to implement the idea.

* 1. **Content**

After importing all the necessary libraries and the data of housing in Guangzhou, we isolate the elements we need from the dataset, which are ‘dealing\_price’, ‘posting\_price’, ‘watched’ (page views), and ‘posting\_minus\_dealing’ (price difference). (Fig.4.2)

After calculating the price difference:

表格

描述已自动生成

Fig.4.2. data frame includes the price difference

Then we reshape the ‘posting\_minus\_dealing’ and ‘watched’ columns as two arrays, array ‘x’ and array ‘y’. Using ***train\_test\_split()*** function to split the data into training and testing data; ***linearRegression()*** function to start model training.

However, the R^2 score is not as expected. (Fig.4.3)

文本

描述已自动生成

Fig.4.3. R^2 score of the linear regression model

So we decide to use a logistic regression to build a binary classification model. By turning ‘posting\_minus\_dealing’ to ‘0’ and ‘1’. (Fig.4.4)

That is, ‘0’ means posting price minus dealing price less than(<) 0, ‘1’ means posting price minus dealing price greater or equal to (>=) 0, to prepare for the classification model training.

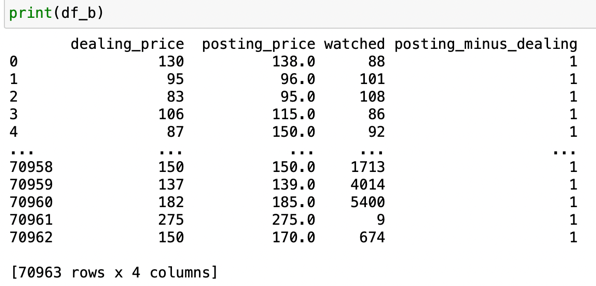


Fig.4.4 Prepare for logistic model training

Rebuild the ‘x’ and ‘y’ array, and Start model building by using LogisticRegression.fit() as shown in Fig.4.5

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

Fig.4.5. Logistic model training

Calculate the prediction probability and model R^2 score. (Fig.4.6)

图形用户界面, 文本, 应用程序

描述已自动生成

Fig.4.6. Prediction probability and R^2 score

1. **Conclusion**

From the research in the third chapter, although we do not directly see the relationship between the housing type and the price per square meter of the houses, we found that people prefer houses with two bedrooms, that is to say the prices of these houses has greater volatility than other houses, and will generally rise greatly under the rising trend of house prices. And the demand for houses one bedroom is not very high, with the fluctuation of house prices is very small, and even fell.

From the bar chart in the second chapter, we can see that the total amount of house purchase in 2020 is four to five times than that of previous years. What caused the sharp increase of house purchase sales in 2020? Would the difference in the total selling price be related to the season? These problems are need to be studied in further study.

1. **Reference**

[1] *“Python Lambda.” GeeksforGeeks, 27 Dec. 2019, www.geeksforgeeks.org/python-lambda/.*

[2] *“Understanding TF-IDF (Term Frequency-Inverse Document Frequency).” GeeksforGeeks, 22 Jan. 2021, www.geeksforgeeks.org/understanding-tf-idf-term-frequency-inverse-document-frequency/.*

[3] *“使用sklearn提取文本的tfidf特征.” 简书, www.jianshu.com/p/c7e2771eccaa.*

[4] *高德地图API, lbs.amap.com/tools/picker.*