Статистическая выборка по ценам на офисную недвижимость и построение многофакторной регрессии

Исходные данные:
Файл формата MS EXCEL: «Офисы Продажа 2021г.xlsx»

ЗАДАНИЕ:
Задание состоит в изучении статистической выборки по ценам на офисную недвижимость и построении многофакторной регрессии. А именно зависимости удельной стоимости недвижимости (удельная стоимость, руб./кв.м = цена предложения, руб./общая площадь, кв.м) от конкретных ценообразующих факторов – общей площади и расстояния до центра города.

1 Выборка объектов:
Только класса С, рекомендуется очистить данные в том числе от выбросов.

2 Исследование:
Построение многофакторной регрессии

3 Используемый стек технологий:
LibreOffice,
Python+librarys:
pandas,
numpy,
matplotlib,
seaborn,
scipy,
...и другие сторонние библиотеки.

РЕШЕНИЕ:

1 Выборка объектов исследования
Воспользуемся библиотекой для обработки и анализа данных — pandas.
Первоначальная задача — формирование объекта типа pandas.DataFrame, для получения объекта воспользуемся методом read_excel():

File: selection.py:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from scipy import stats

#load .xlsx to pd.DataFrame
df_edit = pd.read_excel('Офисы Продажа 2021г.xlsx',sheet_name='Sheet0')
```

Далее определяем наименования столбцов для всех необходимых расчётов при формирования окончательной статистической выборки, а также учитываем тот факт, что необходимы объекты только класса С:

```
#selection columns
df = df_edit[['id','Класс','Общая площадь','Цена', 'Расстояние до центра, м.']]
df = df.loc[df['Класс']=='C']
```

Исключаем значения NaN(строки с пустыми значениями) с помощью метода dropna():

```
# apply dropna()
df = df.dropna()
df = df.reset_index(drop=True)
```

Запускаем несколько тестов для проверки, что метод отработал правильно:

```
def test_function():
    """
    tests function after dropna()
    """
    assert round(float(df.loc[df['id'] == 2562125]['Общая площадь']),2) == round(1058.900024, 2)
    assert round(float(df.loc[df['id'] == 2562125]['Цена']),2) == round(25000000, 2)
    assert round(float(df.loc[df['id'] == 2562125]['Расстояние до центра, м.']),2) ==
round(7358.726948,2)
    #assert df.loc[df['id'] == 2562125]['Класс'] == 'C'

    assert round(float(df.loc[df['id'] == 2362259]['Общая площадь']),2) == round(151.000000,2)
    assert round(float(df.loc[df['id'] == 2362259]['Цена']),2) == round(4000000,2)
    assert round(float(df.loc[df['id'] == 2362259]['Расстояние до центра, м.']),2) ==
round(6348.352448,2)

    assert round(float(df.loc[df['id'] == 2359419]['Общая площадь']),2) == round(50.200001, 2)
    assert round(float(df.loc[df['id'] == 2359419]['Цена']), 2) == round(1400000, 2)
    assert round(float(df.loc[df['id'] == 2359419]['Расстояние до центра, м.']), 2) ==
round(12991.190962, 2)

test_function()
```

Просчитываем и добавляем дополнительный столбец с результатами расчётов удельной стоимости:

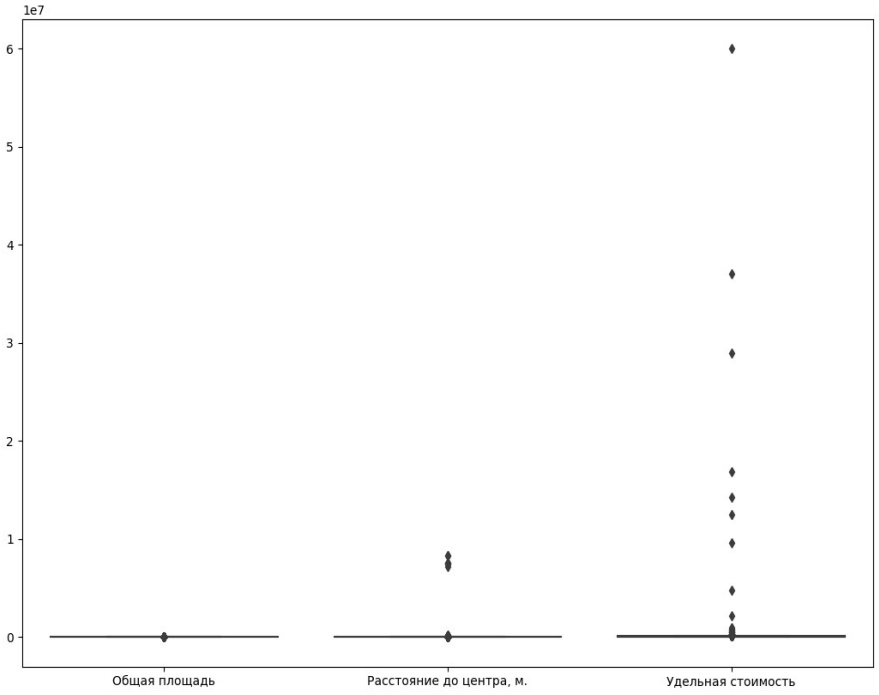
```
# add column 'Удельная стоимость'
df['Удельная стоимость'] = df['Цена']/df['Общая площадь']
```

Формируем график для просмотра выбросов значений, для этих целей воспользуемся графиком следующего типа:

Диаграмма размаха (англ.box-and-whiskers diagram or plot, box plot)— график, использующийся в описательной статистике, компактно изображающий одномерное распределение вероятностей.

Такой вид диаграммы в удобной форме показывает медиану (или, если нужно, среднее), нижний и верхний квартили, минимальное и максимальное значение выборки и выбросы. Несколько таких ящиков можно нарисовать бок о бок, чтобы визуально сравнивать одно распределение с другим; их можно располагать как горизонтально, так и вертикально. Расстояния между различными частями ящика позволяют определить степень разброса (дисперсии) и асимметрии данных и выявить выбросы.

```
df_calc = df[['Общая площадь', 'Расстояние до центра, м.', 'Удельная стоимость']]
sns.boxplot(data=df_calc)
plt.show()
```



Для оценки выбросов посмотрим существенности различия средних значений двух выборок значения 75% процентиля и максимального значения:

print(df_calc.describe())

	Общая площадь	*Расстояние до центра, м.*	*Удельная стоимость*
count	*6269.000000*	*6.269000e+03*	*6.269000e+03*
mean	*360.874350*	*1.409743e+04*	*9.430648e+04*
std	*1185.410403*	*2.575604e+05*	*1.023342e+06*
min	*4.000000*	*0.000000e+00*	*6.986592e+01*
25%	*55.000000*	*1.557720e+03*	*4.186695e+04*
50%	*110.000000*	*3.259351e+03*	*6.000000e+04*
75%	***274.000000***	***5.904803e+03***	***8.000000e+04***
max	***58643.000000***	***8.268535e+06***	***6.005223e+07***

Вычислим z-оценку каждого значения в выборке относительно среднего значения выборки и стандартного отклонения с помощью модуля stats.zscore:

 # selection results
df_super=df_calc[(np.abs(stats.zscore(df_calc)) < 0.3).all(axis=1)

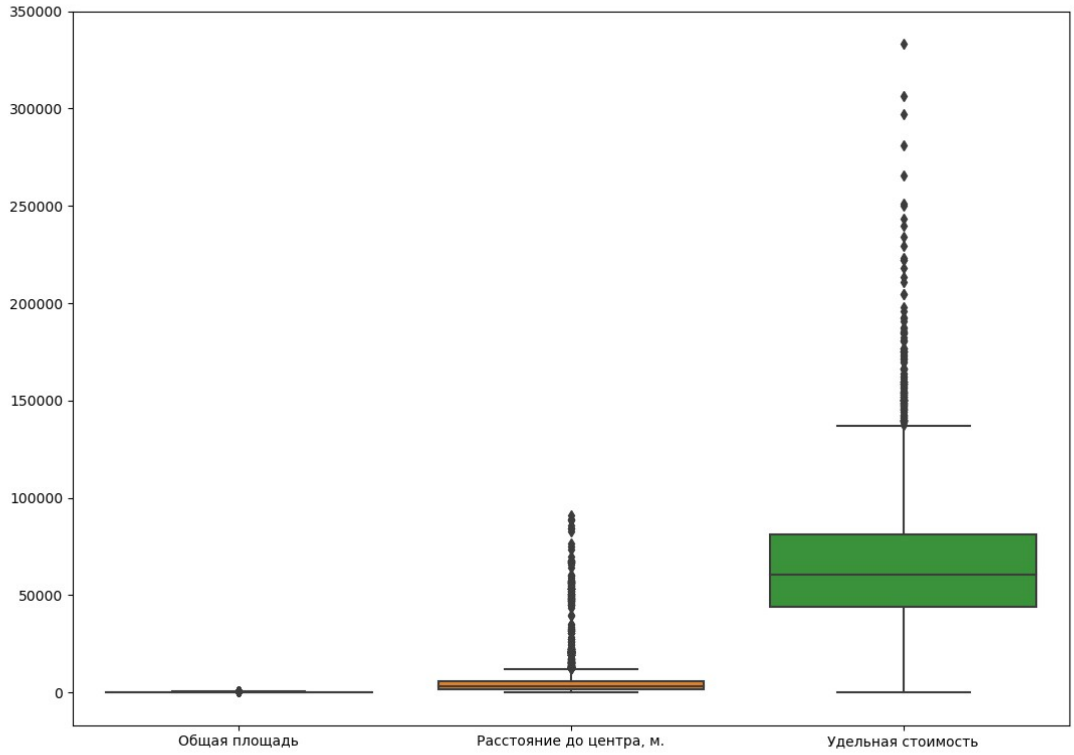
Посмотрим на результаты после исключения выбросов по данному методу и опять построим диаграмму:

print(df_super.describe())

	Общая площадь	*Расстояние до центра, м.*	*Удельная стоимость*
count	*5628.000000*	*5628.000000*	*5628.000000*

mean	*153.352861*	*5120.085177*	*65511.136545*
std	*148.049555*	*8317.579309*	*31991.489939*
min	*7.000000*	*0.000000*	*69.865915*
25%	*51.000000*	*1510.030426*	*43836.391105*
50%	*97.600002*	*3177.425194*	*60712.941176*
75%	***200.000000***	***5815.428614***	***81250.000000***
max	***716.000000***	***90877.973121***	***333333.333333***

sns.boxplot(data=df_super)
plt.show()



Большой разброс от медианы по значениям в исходных данных, при дальнейшем исключении выбросов статистическая выборка становиться очень мала, что негативно отражается на исследовании множественной регрессии.

Оставим полученные результаты и сохраним все в файл:
#save results to file
df_super.to_excel('output.xlsx')

<u>РЕЗУЛЬТАТ: output.xlsx — статистическая выборка</u>

2 Исследование

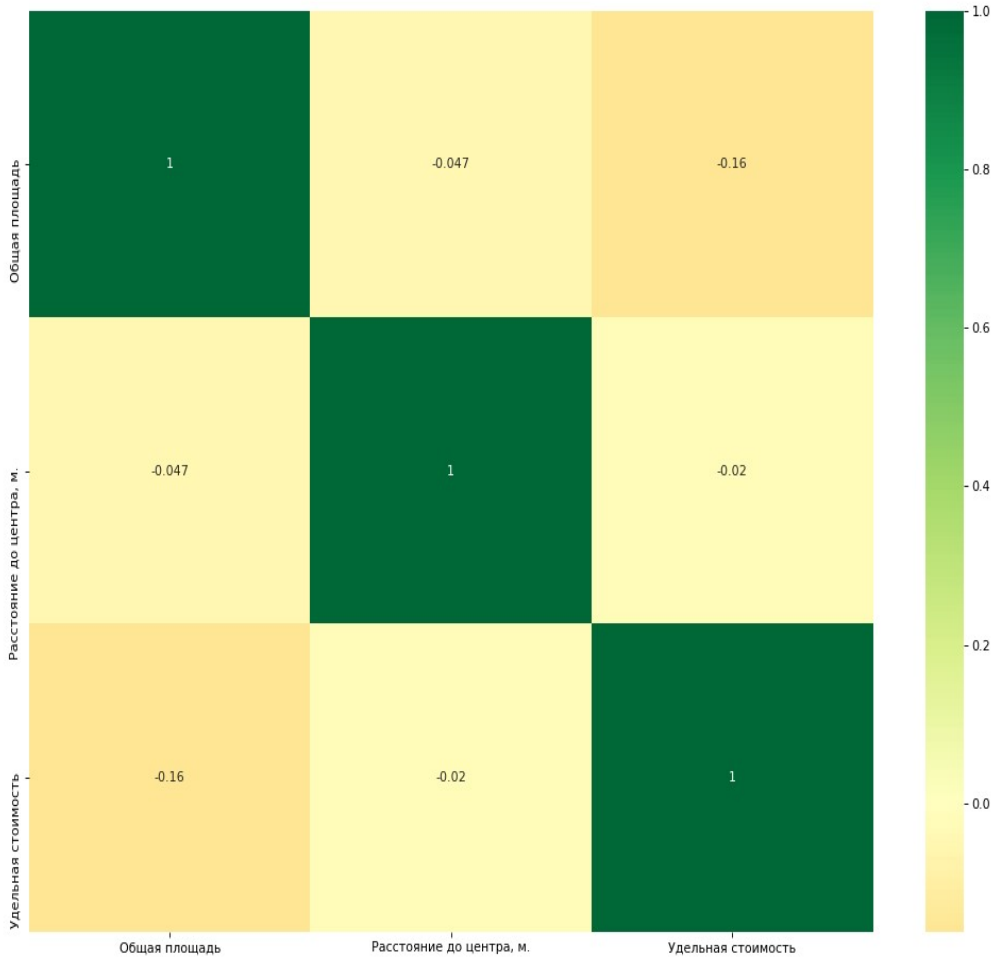
Разведочный анализ

Проведем разведочный корреляционный анализ для оценки тесноты A от B, используем для данных целей коэффициент корреляции.

Подсчитаем коэффициент средствами пакета анализа данных MS Excel:

	Столбец 1	Столбец 2	Столбец 3
Столбец 1	1		
Столбец 2	-0,047344807	1	
Столбец 3	-0,161534687	-0,019686738	1

Тепловая карта средствами python выглядит следующим образом:



Как мы можем заметить все полученные значения близки к нули, что предположительно свидетельствует об отсутствии корреляционной зависимости статистических данных.

Множественная линейная регрессия

Воспользуемся пакетом анализа данных MS Excel для расчета линейной регрессии:

output_regression.xlsx — статистическая выборка + результаты расчетов линейной регрессии.

Простая линейная регрессия использует линейную функцию для прогнозирования значения целевой переменной **y**, содержащий в функции только одну независимую переменную **x1**,

$$y = b_0 + b_1 x_1$$

Множественная линейная регрессия

Регрессионный анализ используется для изучения связей между зависимой переменной и одной или несколькими независимыми, определения тесноты связи и математической зависимости.

Можем ли мы создать модель множественной линейной регрессии, которая прогнозирует удельную стоимость от двух независимых переменных: общей площади и расстояния до центра города? При условии, что статистическая выборка правильна и выбросы были определены правильно, на основе данной исходной выборке — ответ: нет.

Коэффициенты множественной корреляции рассчитываются по данной формуле:

$$R_{z,xy} = \sqrt{\frac{r_{xz}^2 + r_{yz}^2 - 2r_{xz}r_{yz}r_{xy}}{1 - r_{xy}^2}}$$

, где r_{xz} , r_{yz} , r_{xy} *- коэффициенты парной корреляции.*

Все представленные коэффициенты близки к нули, что свидетельствует об отсутствии какой либо регрессионной зависимостей.

Перспективы: Конечно, как и любая статистическая гипотеза, данная, требует тщательной проверки и тестирования результатов выполнения всех инструкций представленного кода, а также соответствию парадигмам математической статистики. Возможно применение принципов ООП для разбиения архитектуры приложения на 3 этапа: выборка данных, исключение выбросов и построение соответствующей модели. Реализация gui или web-интерфейса для выполнения данных задач. Относительно, исключения выбросов и выборки данных возможны другие способы реализации поставленной задачи для сравнения полученных результатов. Для тестового задания данное решение считаю исчерпывающим так как отсутствует коммерческая мотивация в виде требований заказчика в выполнении поставленных целей.

Спасибо, надеюсь на дальнейшее сотрудничество в области программирования и анализа данных.

Соискатель вакансии,

Аналитик данных

Беззубов Андрей

+79081698092