

Государственное бюджетное профессиональное  
Образовательное учреждение Московской области  
«Физико-технический колледж»

## **Аналитический отчет**

# **«Модель оценки цены квартиры на вторичном рынке»**

Работу выполнили:

Студент группы ИСП-22

Бусыгин Егор

Проверил:

Преподаватель М.О.

Коновалов Игорь

Долгопрудный, 2024

# **ВВЕДЕНИЕ**

## **Актуальность задачи**

Оценка недвижимости - важная составляющая девелоперского бизнеса. Информация, о реальной цене квартиры исходя из рынка, интересна для покупателей продавцов, застройщиков, агентов и др.

Эту информацию можно использовать по-разному, в частности, у бизнеса есть потребность использовать такую модель для лидогенерации клиентов. Потенциальный клиент заходит на сайт, хочет оценить свою квартиру, вводит параметры: квартира, на 3 этаже, 3-х комнатная, дизайнерский ремонт. Затем нажимает кнопку «узнать цену», и, прежде чем дать ответ, мы хотим запросить у клиента номер телефона или адрес почты.

Для этого нужна модель оценки цены квартиры по Московскому региону: Москва, Новая Москва, Московская область.

## **Постановка задачи**

**Цель:** собрать данные и провести разведочный исследовательский анализ данных (EDA) для построения модели, которая будет оценивать цену квадратного метра недвижимости в Московском регионе (Москва, Новая Москва, Московская область).

## **Задачи:**

- Используя открытые источники и личный опыт, составить список параметров, значительно влияющих на цену квадратного метра жилой площади.

- С учётом выявленных выше факторов произвести парсинг данных по квартирам на продажу, используя различные парсеры. Данные получаем, используя различные сайты с объявлениями о продаже недвижимости: Циан, Авито, ДомКлик и др.

- Произвести подготовку данных для анализа: проверка на пропуски, выбросы и ошибки. Обработать выявленные аномалии (удалить / заполнить)

- Проведите Исследовательский Анализ Данных (EDA). Постройте распределение основных параметров; визуализируйте взаимосвязи между ними; определите признаки, оказывающие наиболее сильное влияние на целевую переменную.

# СБОР И АНАЛИЗ ДАННЫХ: ОСНОВНЫЕ МЕТОДЫ И ИНСТРУМЕНТЫ

## 1.1 Сбор данных

Основные методы сборки данных были с помощью кода на таком языке программирования как Python. Для парсинга я использовал такие библиотеки как: `clanparser` и `DomClick`. Эти инструменты заходили на веб-страницы Циана и ДомКлика и создавали отдельные `csv` файлы, которые я далее объединил в один файл.

## 1.2 Обработка данных

Для обработки данных я использовал тот же Python, но уже в таком сервисе как Google Colab. Библиотеки, которые я использовал: `Pandas`, `NumPy`, `Matplotlib`, `Seaborn`, `Scikit-learn` Сервис предоставляет среду исполнения кода, визуализацию и удобный формат файла `.ipynb`

Перед непосредственным анализом, я удалил дубликаты, потому что многие квартиры были в размещении на нескольких сайтах и при парсинге некоторые данные могли собираться дважды. Таким образом датасет был 7577 строк.

После удаления дубликатов, мы можем увидеть лишние данные, но для их удаления, надо проверить, что они из себя представляют. Одной колонкой был материал здания. В ней было более 50% значений: -1. Это значит, что тут данные сайт не предоставил, поэтому я принял решение удалить эту колонку.

house_material_type	count
-1	6190
Монолитно-кирпичный	478
Монолитный	443
Монолитно-кирпичный, монолитный	216
Панельный	107
Панельный, монолитный	54
Кирпичный	27
Монолитно-кирпичный, монолитный, кирпичный	26
Монолитно-кирпичный, кирпичный	14
Монолитный, кирпичный	12
Блочный	5
Панельный, монолитно-кирпичный	2
Панельный, монолитно-кирпичный, монолитный, блочный	1
house_material_type	1

Рисунок 1. Содержание колонки "house\_material\_type"

Далее я удалил колонки, которые по моему мнению не имели влияния на анализ и модель. Эти колонки были: `author_type`, `url`, `phone`, `house_number`, `residential_complex`, `street`, `author`, `deal_type`, `house_material_type`, `accommodation_type`. После удаления этих колонок у меня всего осталось 17 строк и 7577 строк в датасете.

Далее я удалил все строки, в которых все столбцы не имеют значений. Это хорошая практика так как такие строки невозможно заполнить ни средним, ни каким другим значением. У меня осталось 7765 строк.

Я перевел цену из типа объект в тип флот, потому что с данными объекта невозможно работать, а тип флот я выбрал, так как цена может быть нецелой. Далее я вывел типы данных и количество ненулевых значений. Мы видим, что все данные, кроме цены являются объектами, что не есть хорошо, потому что с ними модель не может работать.

#	Column	Non-Null	Count	Dtype
0	location	7562	non-null	object
1	floor	7562	non-null	object
2	floors_count	7562	non-null	object
3	rooms_count	7562	non-null	object
4	total_meters	7562	non-null	object
5	price	7562	non-null	float64
6	year_of_construction	7562	non-null	object
7	object_type	7562	non-null	object
8	have_loggia	7562	non-null	object
9	parking_type	7562	non-null	object
10	heating_type	7562	non-null	object
11	finish_type	7562	non-null	object
12	living_meters	7562	non-null	object
13	kitchen_meters	7562	non-null	object
14	ceiling_height	7562	non-null	object
15	district	2982	non-null	object
16	underground	5286	non-null	object

dtypes: float64(1), object(16)

Рисунок 2. Типы данных и кол-во ненулевых значений в датасете

Далее я вывел график этих самых нулевых значений. В основном они были в столбцах метро и район.

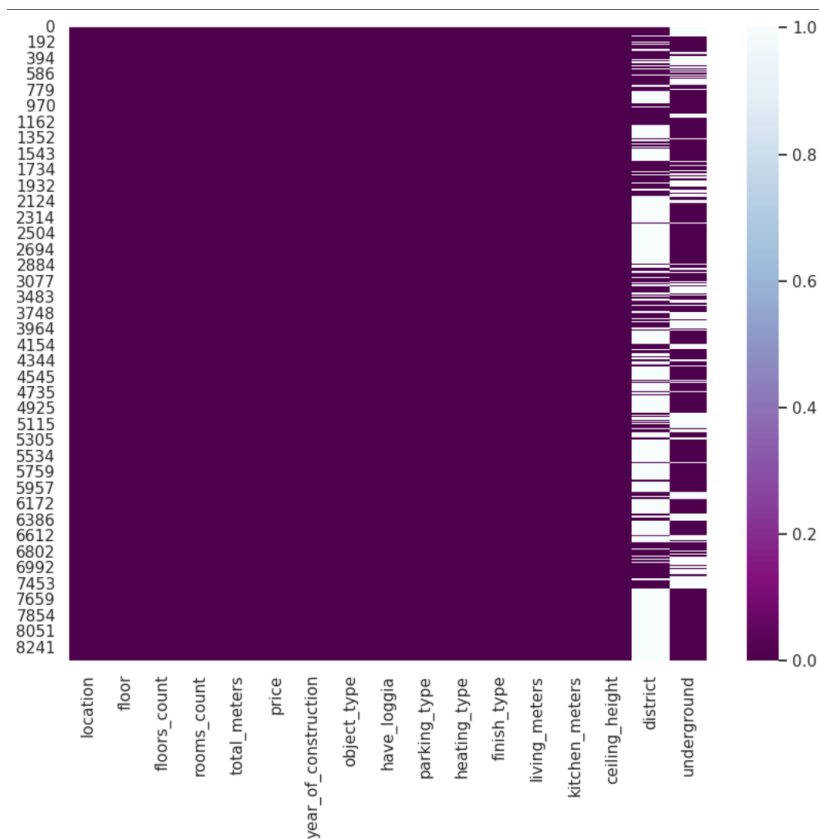
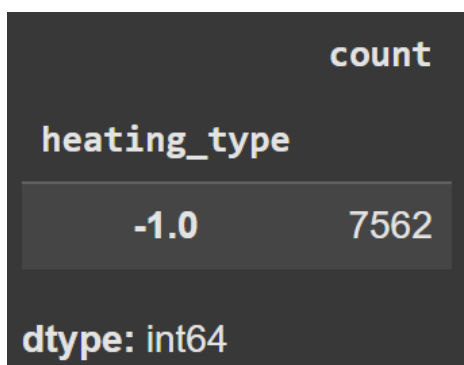


Рисунок 3. График нулевых значений по колонкам.

Следующим шагом я постарался перевести типы данных в числовые. Если квадратные метры и цена могут быть не целыми, то такие параметры как комнаты, этаж, этажи в здании могут быть только целыми, поэтому я перевел их в целые числа. Строками я оставил метро, локацию, тип объекта.

После перевода я вывел целевую переменную для моделирования и анализа. Эта переменная была цена за квадратный метр. Я взял во всех строках цену и разделил на общую площадь. Далее я добавил это значение в отдельную колонку.

После получения целевой переменной у нас все еще был достаточно грязный датасет. Такая колонка как «heating\_type» имела все значения -1, что означает в этом столбце не было данных, поэтому я решил ее удалить.



count	
heating_type	
-1.0	7562

dtype: int64

Рисунок 4. Информация по столбцу "heating\_type"

Далее я проверил столбец наличия лоджии и балкона. Там были строки и числа наличия лоджии или балкона, так как лоджия мало чем отличается от балкона, я решил объединить эти значения и получил наличие число наличия выделенного помещения в доме.

have_loggia	count
-1	3229
1 лоджия	1933
1 балкон	1854
2 лоджии	195
1 лоджия, 1 балкон	178
2 балкона	120
3 лоджии	18
3 балкона	12
2 лоджии, 1 балкон	8
1 лоджия, 2 балкона	7
4 лоджии	4
2 лоджии, 2 балкона	3
4 балкона	1

Рисунок 5. Данные в параметре "have\_loggia"

have_loggia	count
0	5162
1	2032
2	322
3	38
4	8

Рисунок 6. Полученные данные после обработки

После первоначальной чистки я решил вывести график зависимости цены от года постройки здания. И получил аномалии в годе постройки зданий. После чего я принял почистить данные года постройки и убрал все данные, которые старше 1950 года. После чистки осталось 4195 строк.



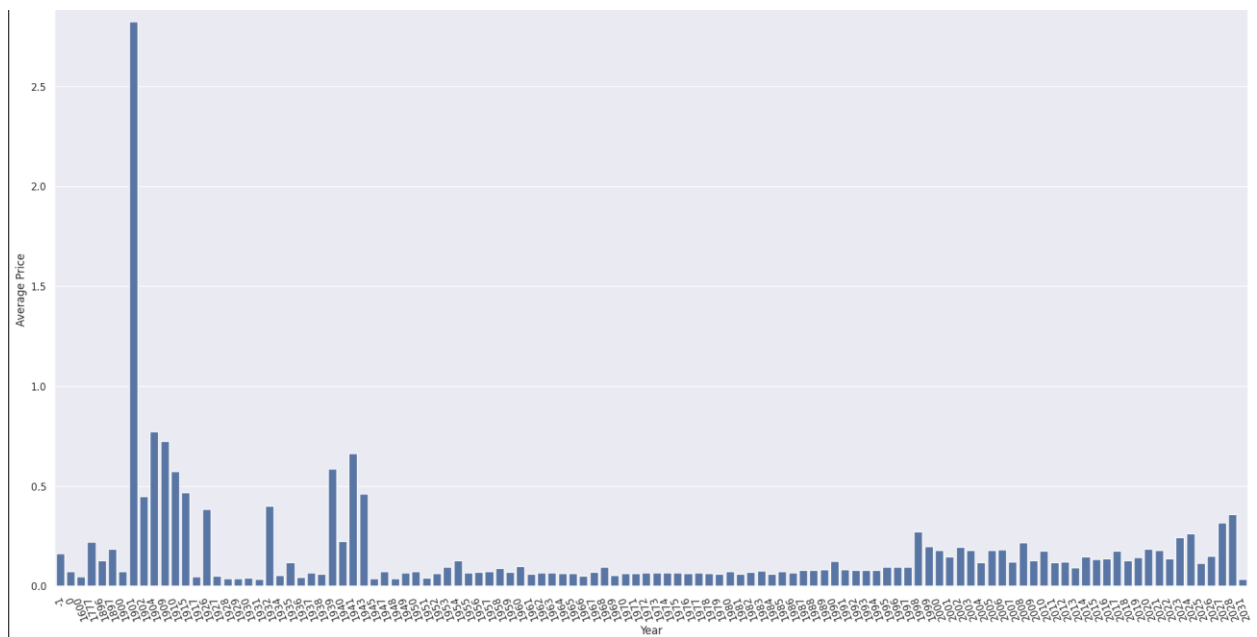


Рисунок 7. Зависимость цены от года постройки.

Далее я проверил колонку парковки и района. Потому что там оставались грязные данные, которые модель бы не переварила. В районах были вообще аномальные данные, которые надо было удалить, тем более там более 50% неопределенных. А парковки мы обработаем дальше, но их можно было оставить и строками, но лучше все же закодировать такие данные.

	count
parking_type	
-1	3324
Наземная	2819
Подземная	833
Открытая	453
Многоуровневая	133

Рисунок 8. Данные парковок.

district	
Неопределен	4580
Дмитров	178
Клин	104
Пресненский	89
Волоколамск	68
...	...
Арзамас	1
Шумерля	1
мкр. Предмestье Глазково	1
Новоалтайск	1
мкр. Пашковский	1
427 rows × 1 columns	

Рисунок 9. Данные по району

Далее мы обработаем колонки жилой площади, высоты потолка и площади кухни. Там есть -1, что значит сайт не предоставил данные по этому параметру. Все -1 мы превратим в нули, а числовые значения оставим. Таким образом будет видно, какие данные у нас есть. А средним значением заполнять не стоит, потому что по личному опыту люди часто смотрят на высоту потолка, размер кухни и жилую площадь, проще говоря, эти параметры очень важные. После такой обработки можно и закодировать остальные категориальные признаки и вывести матрицу корреляции.

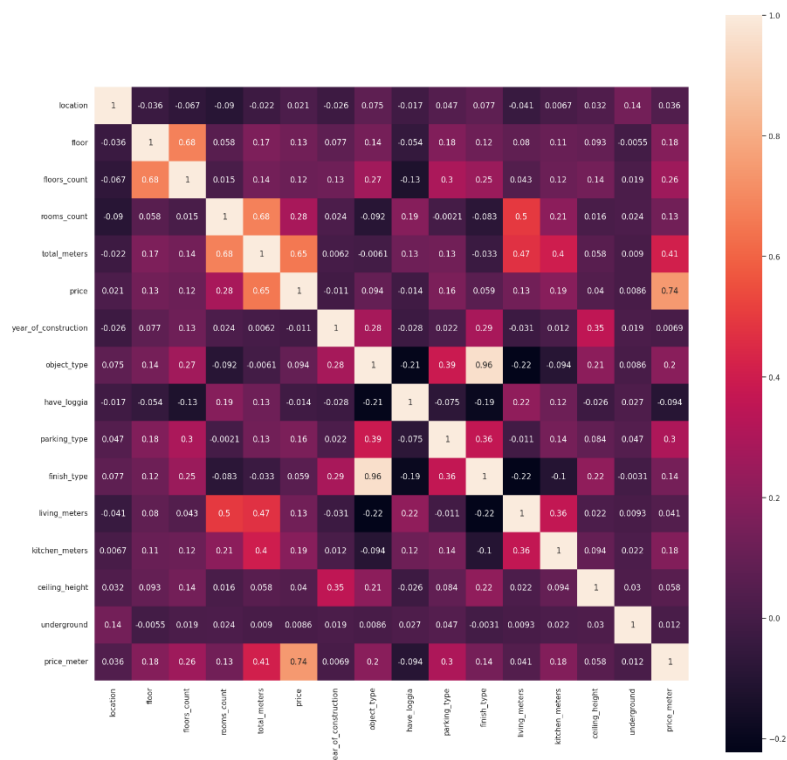


Рисунок 10. Матрица корреляции

Из матрицы видно, что на нашу целевую переменную логично влияет количество метров всего, так как чем больше метров, тем больше цена и больше цена за метр. Так же влияет количество комнат и парковка. Кухня влияет на удивление сильно, даже больше, чем жилая площадь (living\_meters).

Далее посмотрим график цены за площадь. И сразу увидим аномалии. Площадь квартир превышающих 150 квадратных метров может быть только у домов или у объединённых квартир. Оба варианта нам не подходят. Придется удалить данные строки. А после обработки у нас останется всего 3 тысячи строк.



Рисунок 11. Цена против площади

После обработки, я решил проверить высоту потолков. Там тоже оказались аномальные значения для квартир. Высота потолка не может быть 20 и более метров, если это не дом или не объединенная квартира. Пришлось удалить такие аномалии.

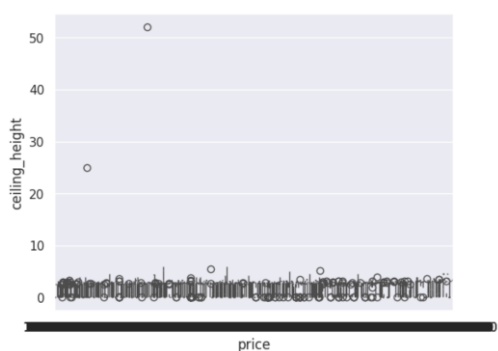


Рисунок 12. Отношения высоты потолка к цене

После обработки таких данных у нас остались колонки метро и парковки, которые тоже следует закодировать для удобной работы. Так как наличие парковки и наличие метро можно обозначить единицей, а отсутствие нулем, то такой путь самый оптимальный. По моему мнению какое метро

именно у дома не особо имеет значение. Так как этот параметр перекрывает район.

После работы с датасетом, я решил проверить гипотезы на графиках Power BI.

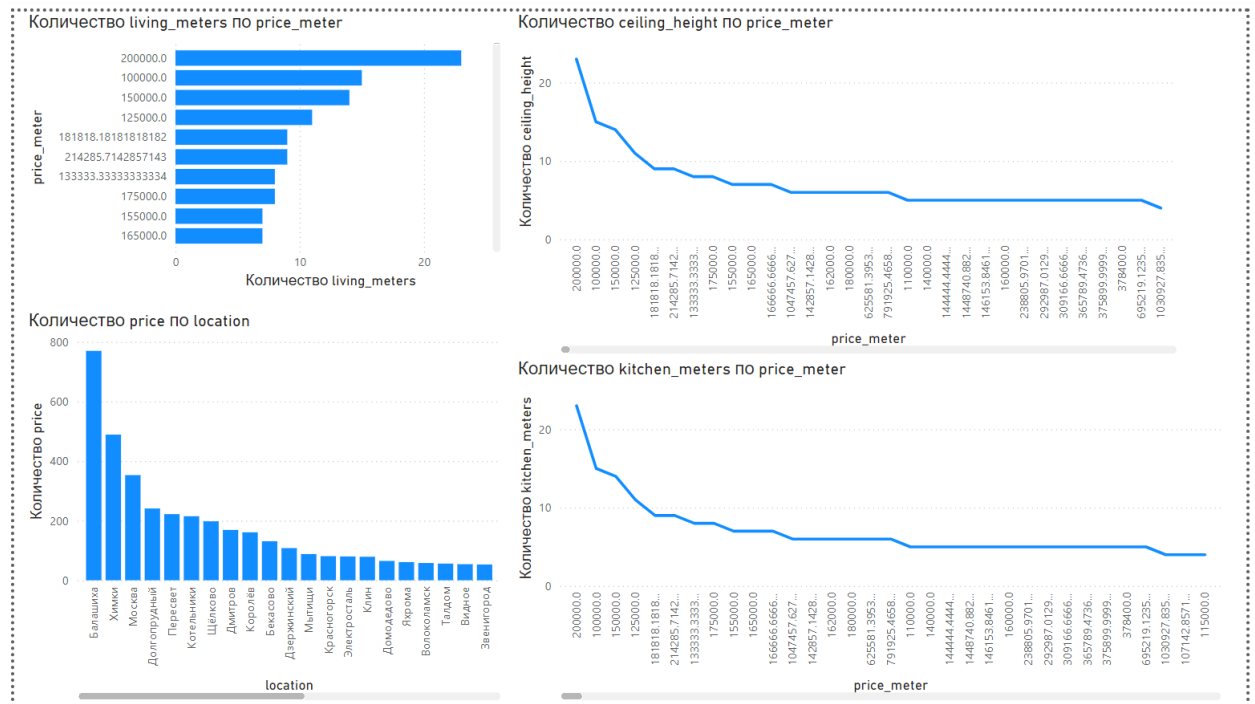


Рисунок 13. Графики Power BI

И так, что мы можем увидеть на графиках. Первым делом на левом верхнем графике видно, что цена за метр увеличивается пропорционально количеству жилой площади. Что по логике, верно, ведь чем больше площадь, тем больше цена, а в соответствии больше и цена за метр всей площади.

Два правых графика показывают, что цена за метр возрастает с ростом высоты потолка и количеством квадратных метров кухни. Такая корреляция вполне допустима. По личному опыту я могу сказать, что многим людям нужен высокий потолок, так как в таких квартирах комфортнее жить. А более просторная кухня позволяет комфортнее находиться в ней.

Нижний левый график показывает зависимость цены от места нахождения. Балашиха имеет самую большую цену. Это объясняется тем, что скорее всего там больше всего 4-х комнатных квартир. Так же больше всего в целом квартир из этого города.

На конец работы и анализа можно смело сказать, что у нас осталось чуть меньше половины строк, что относительно мало, но достаточно для работы.

## **ЗАКЛЮЧЕНИЕ**

### **2.1 Что можно улучшить.**

После работы, я решил подумать, что можно было бы исправить в анализе и работе. Опираясь на собственный опыт, можно легко понять, что район, а то и город сильно влияют на стоимость квартиры, поэтому колонку лучше кодировать, нежели удалять.

Так же мне стоило выяснить более подробно почему Балашиха находится на первом месте по цене. Явно это не является аномалией, но выяснить причину стоило бы

Очень много данных я удалил, когда очищал аномалии с годом конструкции здания. При этом аномалии 1600 года можно вполне спокойно удалить, а вот дома прошлого века, которые были построены около 1900 года, можно было бы и оставить, ведь они спокойно могли реконструироваться и быть пригодными для проживания.

### **2.2 Выводы.**

В выводе можно смело сказать, что на общую стоимость и на цену за квадратный метр влияют такие параметры: общая площадь, площадь кухни, район и наличие метро. С этими параметрами надо работать максимально аккуратно, ведь из-за сильной корреляции, наш анализ может стать некорректным, при ошибочных действиях