




Подробное описание шагов исследования.

Проект в рамках всероссийской научно-технологической программы по решению проектных задач в области искусственного интеллекта и смежных дисциплин «Сириус.ИИ»






Шаг 1

Обработка отсутствующих значений.

Взглянув на строки с пропущенными значениями, было принято решение сначала **интерполировать часть данных на основе уже известных**, а после **убрать те строки, в которых есть пропущенные значения**, потому что в ином случае (после объединения `train.csv` и `macro.csv`) просто **удалив все строки с пропущенными значениями** не остаётся строк с хоть какими-то значениями в принципе.



Шаг 2

Обработка лишних значений.

Посмотрев на корреляцию между полями и целевой переменной, можно понять, что к признакам, которые можно убрать, не повлияв на целевую переменную, относятся:

- `trc_sqm_500` (0.000374),
- `divorce_rate` (0.000385),
- `build_year` (0.002161),
- `cafe_sum_3000_max_price_avg` (0.002200)
- `balance_trade` (0.003161)

Шаг 3

Выявление аномалий.

Если посмотреть на данные, то можно увидеть аномалии и выбросы. Есть несколько причин того, с чем они могут быть связаны, самые вероятные — опечатки при записи данных, разные экономические события (например кризис на рынке). Для их обработки было принято решение **удалить строки с аномальными значениями и выбросами.**

Удаление строк с аномальными значениями и выбросами

```
data_cleaned = all_data_interpolated[~std_outliers].copy()  
data_cleaned
```

	id	timestamp	full_sq	life_sq	floor	max_floor	material	build_year	num_room	kitch_sq	...	provision_retail_space_modern_sqm	turnover_catering_per_cap	theaters_viewers_per_1000_cap	seats_theather_rfmin_per_100000_cap	museum_visitis_per_100_cap	bandwidth_per_100000_cap
8278	8281	2013-06-01	63	22.000000	15.0	12.428571	1.000000	2005.857143	2.285714	6.714286	...	271.0	9350.0	627.0	0.43939	1440.0	...
8289	8292	2013-06-03	82	43.666667	11.0	9.000000	1.666667	1966.666667	2.333333	8.000000	...	271.0	9350.0	627.0	0.43939	1440.0	...
8293	8296	2013-06-03	38	19.000000	17.0	17.000000	1.000000	1986.000000	1.000000	8.000000	...	271.0	9350.0	627.0	0.43939	1440.0	...
8295	8298	2013-06-03	14	14.000000	1.0	16.352941	1.058824	1985.087719	1.058824	8.235294	...	271.0	9350.0	627.0	0.43939	1440.0	...
8301	8304	2013-06-04	57	57.000000	13.0	14.411765	1.235294	1982.350877	1.235294	8.941176	...	271.0	9350.0	627.0	0.43939	1440.0	...
...
25519	25522	2014-11-29	61	38.000000	4.0	9.000000	1.000000	1972.000000	3.000000	7.000000	...	271.0	10311.0	627.0	0.44784	1440.0	...
25520	25523	2014-11-29	38	36.000000	13.0	17.000000	1.000000	1965.000000	1.000000	1.000000	...	271.0	10311.0	627.0	0.44784	1440.0	...
25523	25526	2014-11-29	45	18.000000	5.0	5.000000	1.000000	1965.000000	2.000000	5.000000	...	271.0	10311.0	627.0	0.44784	1440.0	...
25524	25527	2014-11-29	27	27.000000	21.0	22.000000	6.000000	2015.000000	1.000000	10.000000	...	271.0	10311.0	627.0	0.44784	1440.0	...
25525	25528	2014-11-29	72	44.000000	7.0	16.000000	1.000000	1982.000000	3.000000	10.000000	...	271.0	10311.0	627.0	0.44784	1440.0	...

7927 rows x 391 columns

Шаг 4

Сбалансированность.

Проверять сбалансированность данных стоит по целевой переменной — ценой недвижимости, так как перед нами стоит задача регрессии. Проведя анализ данных, используя коэффициент асимметрии, значение эксцесса и, отобразив распределение на графике, на котором видны пики, можно утверждать о несбалансированности целевой переменной. Для ее балансировки было принято решение убрать строки, из-за которых происходил дисбаланс целевой переменной

```
from scipy.stats import skew, kurtosis

plt.figure(figsize=(10, 4))
plt.subplot(1, 2, 1)
plt.hist(all_data_interpolated["price_doc"], bins=50, density=True, alpha=0.6, color='blue')
plt.title('Histogram of Data')

# Вычисляем коэффициент асимметрии и эксцесс
skewness = skew(all_data_interpolated["price_doc"])
kurt = kurtosis(all_data_interpolated["price_doc"])

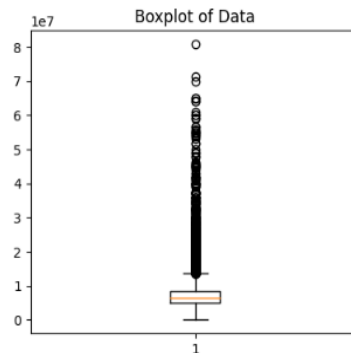
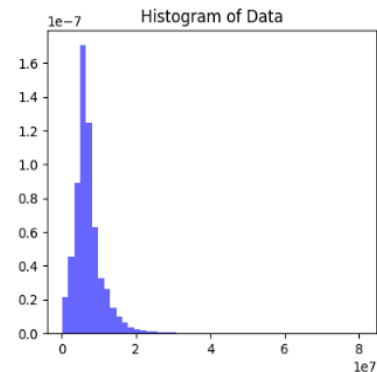
print("коэффициент асимметрии:", skewness)
print("эксцесс:", kurt)

plt.subplot(1, 2, 2)
plt.boxplot(all_data_interpolated["price_doc"])
plt.title('Boxplot of Data')

plt.show()

# Видны пики на графиках и высокие значения коэффициента асимметрии и эксцесса до балансировки целевой переменной
```

коэффициент асимметрии: 3.9482878974423765
эксцесс: 31.397085228301677

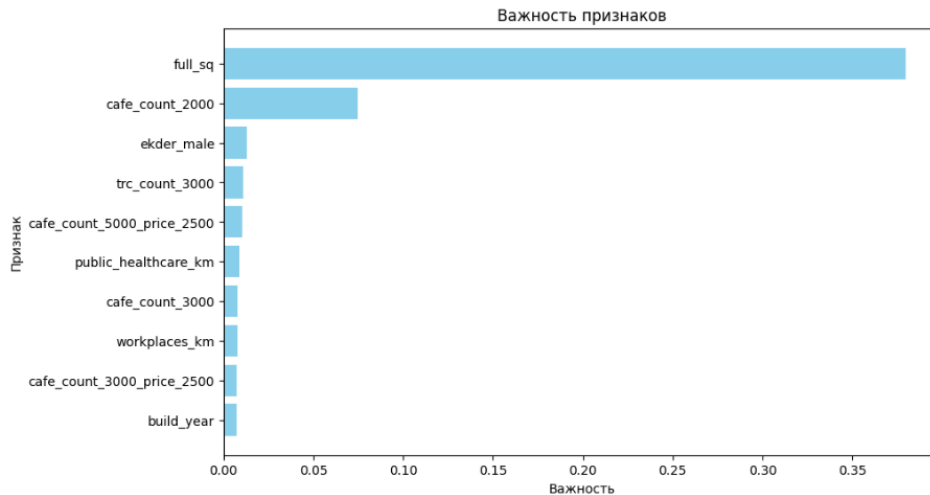


Шаг 5

Базовый отбор признаков.

Проанализировав данные через встроенную функцию, можно увидеть, что самый влиятельный на цену признак – **full_sq**, **cafe_count_2000**, **ekder_male**.

```
[38]: plt.figure(figsize=(10, 6))
plt.barh(importances_df['Признак'][:10], importances_df['Важность'][:10], color='skyblue')
plt.xlabel('Важность')
plt.ylabel('Признак')
plt.title('Важность признаков')
plt.gca().invert_yaxis()
plt.show()
```



[38]:

Шаг 6

Статистики

Согласно проведенному анализу недвижимости в прошлом и в настоящем, мы сделали вывод, что рынок недвижимости очень изменчив.

```
filtered_data_2014["floor"].value_counts()  
# В 2014 году чаще всего покупают недвижимость на 3 этаже или с 1 по 10 этажи, в Москве в 2023 большинство сделок также приходит на 1-10 этажи
```

3.0	1416
2.0	1355
5.0	1249
4.0	1183
1.0	977
7.0	875
6.0	848
9.0	818
8.0	809
12.0	592
10.0	586
11.0	585
14.0	453
13.0	402
16.0	326
17.0	302
15.0	293

Данные о современных показателях:

<https://blog.domclick.ru/novosti/post/kakie-kvartiry-na-vtorichnom-rynke-moskvy-rokupaayut-chashe-vsego-issledovanie-domklik?ysclid=lucxn45v2e969504679>