Стоимость недвижимости

Кузнецов Г.И.

Введение

Разработка многофакторной модели для оценки стоимости недвижимости в заданном регионе на основе гибридных подходов с использованием языка программирования Python.

Задачи:

1. Собрать данные по стоимости недвижимости в выбранном регионе.
2. Выбрать признаки, влияющие на стоимость разных типов недвижимости в регионе.
3. Вдвинуть гипотезы о том, какие признаки наиболее и наименее влияют на стоимость разных типов недвижимости , проверить гипотезы.
4. Смоделировать модель прогнозирования стоимости разных видов недвижимости в выбранном регионе на основе разных методов обучения, сравнить эффективность моделей.
5. Дать рекомендации покупателям и продавцам недвижимости.

Сбор данных

Использовал CianParser и отфильтровал наиболее подходящие признаки.

Сохранил затем это в .csv файлик, чтобы всегда можно было продолжить работу, не парся заново!

```python
from cianparser import CianParser
import pandas as pd

parser = CianParser(location="Москва")

flats = parser.get_flats(
    deal_type="sale",
    rooms=(1, 2, 3),
    additional_settings={
        "start_page": 1,
        "end_page": 80,
        "only_flat": True,
        # "sort_by": "price_from_min_to_max"
    }
)

df = pd.DataFrame(flats)
df.to_csv("flats.csv", index=False, encoding="utf-8-sig")
```

Парсер

```python
features = [
    "price",              # цена - таргет
    "district",           # район
    "underground",        # метро
    "street",             # улица
    "house_number",       # номер дома
    "floor",              # этаж
    "floors_count",       # этажей в доме
    "total_meters",       # общая площадь
    "rooms_count",        # комнаты
    "residential_complex",  # название ЖК
    "url"                 # ссылка (пригодится)
]
df = df[features]
df.to_csv("flats_cut.csv", index=False, encoding="utf-8-sig")
```

Выбор признаков

Предобработка

В основном, ограничился тем, что почистил NaN-значения, пытаясь сохранить размер итак небольшого датасета.

```
df.isna().sum()
```
✓ 0.0s

price	30
district	99
underground	6
street	88
house_number	92
floor	0
floors_count	0
total_meters	0
rooms_count	0
residential_complex	567
url	0

dtype: int64

```
df.info()
```
✓ 0.0s

```
<class 'pandas.core.frame.DataFrame'>
Index: 1246 entries, 1 to 1458
Data columns (total 11 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   price               1246 non-null   float64
 1   district            1246 non-null   object
 2   underground         1246 non-null   object
 3   street              1246 non-null   object
 4   house_number        1246 non-null   object
 5   floor               1246 non-null   int64
 6   floors_count        1246 non-null   int64
 7   total_meters        1246 non-null   float64
 8   rooms_count         1246 non-null   int64
 9   residential_complex 1246 non-null   object
 10  url                 1246 non-null   object
dtypes: float64(2), int64(3), object(6)
memory usage: 116.8+ KB
```

Гипотезы

Выдвинул несколько базовых гипотез и проверил их, смотря на корреляцию и некоторые другие показатели.

1. Чем больше общая площадь квартиры (total_meters), тем выше её цена.
2. Квартиры в центральных районах (district) стоят дороже, чем в спальных.
3. Близость к метро (underground) увеличивает стоимость квартиры.
4. Наличие жилого комплекса (residential_complex) повышает цену объекта.
5. Этаж (floor) почти не влияет на цену, если дом не высокий.
6. Название улицы (street) и номер дома (house_number) не влияют на цену.
7. Количество комнат (rooms_count) напрямую связано с ценой, но не линейно.

```python
# Гипотеза 1
corr_result = df[['total_meters', 'price']].dropna()
corr_coef, corr_p = pearsonr(corr_result['total_meters'], corr_result['price'])
print(f"1: {corr_coef}, {corr_p}")

# Гипотеза 2
anova_df = df[['district', 'price']].dropna()
district_groups = [g['price'].values for _, g in anova_df.groupby('district')]
anova_f_district, anova_p_district = f_oneway(*district_groups)
print(f"2: {anova_f_district}, {anova_p_district}")

# Гипотеза 3
anova_df2 = df[['underground', 'price']].dropna()
underground_groups = [g['price'].values for _, g in anova_df2.groupby('underground')]
anova_f_underground, anova_p_underground = f_oneway(*underground_groups)
print(f"3: {anova_f_underground}, {anova_p_underground}")

# Гипотеза 4: ANOVA по residential_complex
anova_df4 = df[['residential_complex', 'price']].dropna()
res_groups = [g['price'].values for _, g in anova_df4.groupby('residential_complex')]
anova_f_res, anova_p_res = f_oneway(*res_groups)
print(f"4: {anova_f_res}, {anova_p_res}")

# Гипотеза 5: Корреляция floor и price
corr_result_floor = df[['floor', 'price']].dropna()
corr_coef_floor, corr_p_floor = pearsonr(corr_result_floor['floor'], corr_result_floor['price'])
print(f"5: {corr_coef_floor}, {corr_p_floor}")

# Гипотеза 6: ANOVA для street и house_number
anova_df6a = df[['street', 'price']].dropna()
street_groups = [g['price'].values for _, g in anova_df6a.groupby('street')]
anova_f_street, anova_p_street = f_oneway(*street_groups)

anova_df6b = df[['house_number', 'price']].dropna()
house_groups = [g['price'].values for _, g in anova_df6b.groupby('house_number')]
anova_f_house, anova_p_house = f_oneway(*house_groups)
print(f"6: {anova_f_street}, {anova_p_street}, {anova_f_house}, {anova_p_house}")

# Гипотеза 7: rooms_count и price (ANOVA + корреляция)
anova_df7 = df[['rooms_count', 'price']].dropna()
rooms_groups = [g['price'].values for _, g in anova_df7.groupby('rooms_count')]
anova_f_rooms, anova_p_rooms = f_oneway(*rooms_groups)
corr_coef_rooms, corr_p_rooms = pearsonr(anova_df7['rooms_count'], anova_df7['price'])
print(f"7: {corr_coef_rooms}, {corr_p_rooms}")
```

Гипотезы: обоснование

Гипотеза 1:
Чем больше общая площадь (total_meters), тем выше цена → corr = 0.735, p ≈ 2.8e-212
Сильная положительная корреляция, связь очень значима — площадь сильно влияет на цену.

Гипотеза 2:
Квартиры в разных районах (district) отличаются по цене → F = 5.31, p ≈ 4.28e-51
Различия между районами статистически значимы, район влияет на цену.

Гипотеза 3:
Близость к метро (underground) влияет на цену → F = 3.90, p ≈ 1.38e-52 Метро оказывает значимое влияние на цену, по выборке.

Гипотеза 4:
Жилой комплекс (residential_complex) повышает цену → F = 5.75, p ≈ 2.28e-91
Наличие и тип ЖК значимо влияют на стоимость жилья.

Гипотеза 5:
Этаж (floor) почти не влияет на цену → corr = -0.099, p ≈ 0.00044
Связь слабая, но значимая, влияние небольшое и скорее обратное.

Гипотеза 6:
Название улицы (street) и номер дома (house_number) не влияют
street: F = 9.05, p ≈ 3.52e-149
house_number: F = 1.16, p ≈ 0.031
Улица заметно влияет на цену, а номер дома — незначительно, но чуть выше порога (0.05).

Гипотеза 7:
Количество комнат (rooms_count) влияет на цену, но не линейно → corr = 0.30, p ≈ 2.2e-27
Есть умеренная положительная связь, значимая, но она не строго линейная.

Моделирование

```python
X = df.drop(columns=['price', 'url'])
y = df['price']
```

```python
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

```python
models = {
    'Linear Regression': LinearRegression(),
    'Decision Tree': DecisionTreeRegressor(random_state=42),
    'Random Forest': RandomForestRegressor(random_state=42, n_estimators=100),
    'Gradient Boosting': GradientBoostingRegressor(random_state=42, n_estimators=100)
}
results = {}
```

```python
cat_features = ['district', 'underground', 'street', 'house_number', 'residential_complex']
num_features = ['floor', 'floors_count', 'total_meters', 'rooms_count']

preprocessor = ColumnTransformer([
    ('num', Pipeline([
        ('imputer', SimpleImputer(strategy='median')),
        ('scaler', StandardScaler())
    ]), num_features),
    ('cat', Pipeline([
        ('imputer', SimpleImputer(strategy='constant', fill_value='missing')),
        ('onehot', OneHotEncoder(handle_unknown='ignore'))
    ]), cat_features)
])
```

```python
for name, model in models.items():
    pipeline = Pipeline([
        ('preprocessor', preprocessor),
        ('regressor', model)
    ])
    pipeline.fit(X_train, y_train)
    preds = pipeline.predict(X_test)
    results[name] = {
        'RMSE': mean_squared_error(y_test, preds),
        'MAE': mean_absolute_error(y_test, preds),
        'R2': r2_score(y_test, preds)
    }
```

```python
for name, metrics in results.items():
    print(f"{name}: RMSE={metrics['RMSE']:.2f}, MAE={metrics['MAE']:.2f}, R2={metrics['R2']:.3f}")
```

```
Linear Regression: RMSE=1587469629853561.75, MAE=22972264.25, R2=0.769
Decision Tree: RMSE=2222047473859969.00, MAE=19837366.78, R2=0.676
Random Forest: RMSE=996846489456414.50, MAE=16049866.48, R2=0.855
Gradient Boosting: RMSE=757862874814894.62, MAE=15652485.62, R2=0.890
```

Рекомендации

На основе датасета, анализа гипотез и результатов моделирования, предоставил несколько рекомендаций.

Для покупателей:

1. Цена напрямую связана с площадью — выбирайте квартиры, которые соответствуют вашим финансовым возможностям с учётом этой зависимости. Модель показывает сильную корреляцию, значит переплата за лишние метры — частое явление.
2. Обращайте внимание на район — из анализа и ANOVA ясно, что цена существенно варьируется в зависимости от района, так что если хотите сэкономить, рассмотрите варианты в менее дорогих районах, но с учётом инфраструктуры.
3. Близость к метро повышает стоимость, поэтому если бюджет ограничен, ищите квартиры чуть дальше от станций.
4. Жилой комплекс заметно влияет на цену, поэтому если модель прогнозирует высокую стоимость, проверьте наличие ЖК — это может быть причиной.
5. Этаж практически не влияет на цену (слабая отрицательная связь), так что не стоит переплачивать за «любимые» этажи.
6. Количество комнат влияет, но связь не линейна — больше комнат не всегда значит дороже, смотрите на соотношение цена/комнаты и площадь.

Для продавцов:

1. Используйте площадь и район как ключевые аргументы для цены, так как модель и анализ показывают их главную роль.
2. Подчеркивайте наличие жилого комплекса и близость к метро в объявлениях, т.к. они увеличивают стоимость.
3. Не переоценивайте влияние этажа, улицы или номера дома — модель показывает, что эти факторы значительно меньше влияют на цену.
4. Если квартира с большим количеством комнат, объясните покупателям ценность через площадь и функциональность, так как модель выявила сложную нелинейную связь.
5. Используйте модель, чтобы прогнозировать адекватную цену и быстрее закрыть сделку, избегая завышений или недооценок.

Выводы

- собраны и подготовлены данные по стоимости недвижимости в выбранном регионе;
- выбраны и проанализированы ключевые признаки, влияющие на цену различных типов недвижимости;
- сформулированы и проверены гипотезы о влиянии признаков на стоимость объектов;
- построены и обучены модели прогнозирования стоимости недвижимости с использованием разных алгоритмов, проведено сравнение их эффективности;
- разработаны практические рекомендации для покупателей и продавцов, основанные на результатах анализа и моделирования.

Спасибо за внимание