

Лабораторная работа №3

Предсказание цен квартир с помощью линейной регрессии

Зыкин Леонид
470912

1 Постановка задачи

В данной лабораторной работе решалась задача предсказания цен на квартиры на основе их характеристик. Дано 100,000 записей с информацией о квартирах: площади помещений, этажность, год постройки, район, наличие коммуникаций и другие параметры. Требуется построить модель линейной регрессии, которая будет предсказывать цену квартиры с минимальной ошибкой RMSE.

2 Анализ данных

Исходный датасет содержит 19 признаков:

- Числовые признаки: площади (кухня, ванная, общая, дополнительная), этаж, максимальный этаж, год постройки, высота потолков, количество комнат и ванных комнат
- Категориальные признаки: наличие газа, горячей воды, центрального отопления, тип дополнительной площади (балкон/лоджия), название района

Целевая переменная — цена квартиры в рублях. Распределение цен имеет положительную асимметрию, что типично для рынка недвижимости.

3 Предобработка данных

3.1 Создание новых признаков

Для улучшения качества модели были созданы дополнительные признаки, которые могут лучше описывать взаимосвязи между параметрами квартиры и её ценой:

- **Соотношения площадей:** отношение площадей различных помещений к общей площади (например, доля кухни, ванной комнаты)
- **Возраст квартиры:** вычисляется как разность между текущим годом (2025) и годом постройки
- **Площадь на комнату:** общая площадь, делённая на количество комнат
- **Полиномиальные признаки:** квадрат и куб общей площади, квадрат возраста — для учета нелинейных зависимостей

- **Взаимодействия признаков:** произведения важных признаков друг на друга (например, количество комнат на общую площадь, этаж на площадь)
- **Логарифмические преобразования:** логарифм от площадей для работы с признаками, имеющими большой разброс значений

3.2 Кодирование категориальных признаков

Категориальные признаки были преобразованы в числовой формат с помощью one-hot encoding. Это позволяет модели учитывать влияние каждого значения категориального признака независимо. Для признака района также были созданы взаимодействия с важными числовыми признаками (площадь, количество комнат, год постройки, этаж, высота потолков), так как цена может по-разному зависеть от этих параметров в разных районах.

3.3 Отбор признаков

После создания всех признаков были применены следующие методы отбора:

1. **Отбор по дисперсии:** удалены признаки с дисперсией меньше 0.01, так как они практически не меняются и не несут полезной информации
2. **Удаление мультиколлинеарности:** удалены признаки с корреляцией больше 0.95, чтобы избежать избыточности и улучшить устойчивость модели

3.4 Масштабирование

Все числовые признаки были стандартизированы с помощью StandardScaler (приведение к нулевому среднему и единичной дисперсии). Это важно для линейной регрессии, так как признаки имеют разные масштабы (например, площадь измеряется в квадратных метрах, а год постройки — в годах).

4 Модель

В качестве модели использовалась обычная линейная регрессия из библиотеки scikit-learn. Линейная регрессия подходит для данной задачи, так как позволяет интерпретировать влияние каждого признака на цену, а также быстро обучается на больших объемах данных.

5 Реализация

Основной код предобработки данных представлен ниже:

```

1 def preprocess_data(df , is_train=True , onehot_encoder=None ,
2                     scaler=None , columns_to_drop=None ,
3                     feature_names=None):
4     df = df .copy()
5
6     # Creating new features
7     df [ 'area_ratio' ] = df [ 'kitchen_area' ] / (df [ 'total_area' ] + 1e-6)
8     df [ 'bath_ratio' ] = df [ 'bath_area' ] / (df [ 'total_area' ] + 1e-6)
9     df [ 'other_ratio' ] = df [ 'other_area' ] / (df [ 'total_area' ] + 1e-6)

```

```

10     df['extra_area_ratio'] = df['extra_area'] / (df['total_area'] + 1e
11         -6)
11     df['floor_ratio'] = df['floor'] / (df['floor_max'] + 1e-6)
12     df['age'] = 2025 - df['year']
13     df['area_per_room'] = df['total_area'] / (df['rooms_count'] + 1)
14     df['bath_per_bathroom'] = df['bath_area'] / (df['bath_count'] + 1e
15         -6)
15     df['kitchen_per_total'] = df['kitchen_area'] / (df['total_area'] +
16         1e-6)
16
17     df['total_rooms'] = df['rooms_count'] + df['bath_count']
18     df['area_per_total_room'] = df['total_area'] / (df['total_rooms'] +
19         1)
20
20     # Polynomial features
21     df['total_area_sq'] = df['total_area'] ** 2
22     df['total_area_cub'] = df['total_area'] ** 3
23     df['age_sq'] = df['age'] ** 2
24
25     # Feature interactions
26     df['rooms_area_interaction'] = df['rooms_count'] * df['total_area']
27     df['floor_area_interaction'] = df['floor'] * df['total_area']
28     df['ceil_height_area'] = df['ceil_height'] * df['total_area']
29     df['rooms_floor_interaction'] = df['rooms_count'] * df['floor']
30     df['year_area_interaction'] = df['year'] * df['total_area']
31
32     # Logarithmic transformations
33     df['log_total_area'] = np.log1p(df['total_area'])
34     df['log_kitchen_area'] = np.log1p(df['kitchen_area'])
35     df['log_extra_area'] = np.log1p(df['extra_area'] + 1)
36
37     # Additional ratios
38     df['kitchen_bath_ratio'] = df['kitchen_area'] / (df['bath_area'] +
39         1e-6)
40     df['total_extra_ratio'] = (df['total_area'] + df['extra_area']) /
41         (df['total_area'] + 1e-6)
42     df['floor_ratio_sq'] = df['floor_ratio'] ** 2
43     df['area_per_room_sq'] = df['area_per_room'] ** 2
44
44     # ... (categorical encoding, feature selection, scaling)

```

Листинг 1: Функция предобработки данных

Функция удаления мультиколлинеарности:

```

1 def remove_multicollinearity(X, threshold=0.95):
2     corr_matrix = X.corr().abs()
3     upper_triangle = corr_matrix.where(
4         np.triu(np.ones(corr_matrix.shape), k=1).astype(bool)
5     )
6
7     to_drop = [column for column in upper_triangle.columns
8                 if any(upper_triangle[column] > threshold)]
9
10    if to_drop:

```

```

11         X = X.drop(columns=to_drop)
12
13     return X, to_drop

```

Листинг 2: Removing multicollinear features

Основной цикл обучения:

```

1 # Loading data
2 data = pd.read_csv(train_path)
3 test_data = pd.read_csv(test_path)
4
5 # Preprocessing training data
6 X, y, onehot_encoder, scaler, columns_to_drop, feature_names = \
7     preprocess_data(data, is_train=True)
8
9 # Training model
10 model = LinearRegression()
11 model.fit(X, y)
12
13 # Preprocessing test data
14 X_test, _, _, _, _ = preprocess_data(
15     test_data,
16     is_train=False,
17     onehot_encoder=onehot_encoder,
18     scaler=scaler,
19     columns_to_drop=columns_to_drop,
20     feature_names=feature_names
21 )
22
23 # Making predictions
24 predictions = model.predict(X_test)

```

Листинг 3: Model training

6 Результаты

После обучения модели на полном обучающем датасете (100,000 записей) и применения её к тестовым данным получены предсказания цен. Модель показала стабильные результаты с RMSE около 476,000 рублей, что составляет примерно 2.9% от средней цены квартиры.

Основные характеристики предсказаний:

- Средняя предсказанная цена: около 16.5 миллионов рублей
- Стандартное отклонение: около 5.9 миллионов рублей
- Диапазон предсказаний соответствует диапазону цен в обучающих данных

7 Выводы

В ходе выполнения лабораторной работы была построена модель линейной регрессии для предсказания цен на квартиры. Ключевые моменты:

1. Тщательная предобработка данных с созданием новых признаков значительно улучшила качество модели
2. Удаление мультиколлинеарных признаков помогло сделать модель более устойчивой
3. Взаимодействия между категориальными и числовыми признаками позволили учесть неоднородность влияния параметров в разных районах
4. Линейная регрессия показала хорошие результаты для данной задачи, обеспечив приемлемую точность при простоте интерпретации

Модель готова к использованию и может быть применена для предсказания цен на новые объекты недвижимости.