ОТЧЕТ ПО РЕЗУЛЬТАТАМ РАБОТЫ И ХОДЕ ИССЛЕДОВАНИЯ ПО СОЗДАНИЮ МОДЕЛИ СТОИМОСТИ ЖИЛЬЯ

ЗАДАЧА 1

1. ПОСТАНОВКА ЗАДАЧИ

Цель проекта - разработка прототипа модели машинного обучения, которая будет предсказывать стоимость квартиры по исходным данным. В качестве источника исходных данных предлагается использовать данные сайта магнитогорской недвижимости www.citystar.ru.

2. ВЫБОР И ПОЛУЧЕНИЕ ИСХОДНЫХ ДАННЫХ

При изучении предложенного сайта было выявлено, что необходимые данные с главной страницы закодированы таким образом, что ни одна известная шифровка не может их раскодировать. Было принято решения провести парсинг данных с зеркала magnitogorsk-citystar.ru. Результаты тестовой прогонки данных через загруженные библиотеки оказались корректными и соответствуют данным с исходного сайта.

3. ВЫБОР МЕТОДА РЕШЕНИЯ

Для решения задачи парсинга были использованы стандартные и сторонние библиотеки для работы с http - запросами request и парсингом данных BeatifulSoup. Нашей задачей является создание базового прототипа с простым и быстрым решением для перспективы масштабирования и пополнения базы данных, поэтому были использованы минимальные ресурсы для сбора данных.

После сбора данных у нас получилась выборка размером 447x5.

4. ОПИСАНИЕ АЛГОРИТМА РЕШЕНИЯ

По завершении сбора данных далее стоит задача подготовки их к проведению моделирования. Для этого был проведен разведочный анализ полученных данных и составлен план по их обработке.

Данные были собраны и загружены в csv-файл - это экономит ресурсы времени и памяти компьютера. Далее был разработан план по предобработке данных.

На основе колонки flat_type были сгенерированы rooms и layout_type, обозначающие соответственно количество комнат и тип планировки. В некоторых строках отсутствовало значение количество комнат, поэтому там была поставлена заглушка для данной категории квартир. При изучении колонки layout_type было принято решение ее удаления в связи с большим количеством пропусков, около 2/3.

На основе floor_type были сгенерированы колонки floor и floor_type, обозначающие соответственно этаж расположения квартиры и этажность дома. Была проведена проверка аномалий, т.е. наличие объектов, где этаж квартиры был выше этажности дома.

На основе колонки square были сгенерированы колонки total_area, living_area, kitchen_area, обозначающие соответственно общую площадь квартиры, жилую площадь и площадь кухни. В данных имеется интересная закономерность: нулевые площади жилой зоны и кухни. Согласно официальной документации, нулевые площади жилой зоны и(или) кухни подразумевают собой квартиры-студии, но здесь имеет место быть явление нежелания пользователей заполнять все поля при подачи объявления, поэтому нулевые значения могут иметь любой характер. В связи с этим эти данные останутся без изменения, чтобы не исказить их поведение.

Колонку с целевой переменной привели к численному типу и проверили распределение значений. Удалили 3 выброса по нижней границе, по верхней границе оставили без изменений, т.к. данные значения логически укладываются в допустимый диапазон значений.

На основе колонки full_address были сгенерированы колонки district, street, house_number, обозначающие соответственно район, улицу и номер дома, по которым расположена квартира. Такое разделение позволит снизить дисперсию данных и обобщить их для повышения точности модели. Все значения в новых колонках были откорректированы и устранены неявные дубликаты. После данной процедуры была выявлена закономерность: имеются объекты, у которых отсутствует либо название улицы, либо номер дома. При детальном изучении оказалось, что это одни и те же объекты, которые далее были удалены из выборки.

В конце были заполнены пропуски в колонке district методом машинного обучения но основе имеющихся данных с помощью модели случайного леса, которая эффективно работает на небольших выборках.

Данные обработаны и готовы к моделированию.

5. ОПИСАНИЕ МОДЕЛИ

Для обучения модели была проведен препроцессинг признаков - обработка категориальных и численных признаков кодировщиками для масштабирования данных для корректного обучения линейных и деревянных моделей. Для линейных моделей была проведена проверка на мультиколлинеарность, которая показала

отсутствие корреляций между признаками и наличие корреляции между признаками и целевой переменной, что говорит о положительном влиянии на выбранную метрику.

Далее согласно классической схеме было проведено обучение 3х моделей: линейной, случайно леса и бустинга. Линейная модель является безлайном: она показывает работоспособность модели и значение метрики из коробки при наличие линейной зависимости между признаками. Далее было проведено обучение модели случайного леса, которая на практике показывает хорошие результаты на выборках небольших размеров. В конце было проведено обучение бустинга, которые показал результат немного хуже леса, т.к. бустинг ищет нелинейные зависимости в данных и на практике показывает хороший результат на выборках больших размеров.

6. ОПИСАНИЕ КАЧЕСТВА МОДЕЛИ / ПОЛУЧЕННЫХ РЕЗУЛЬТАТОВ

В качестве тестируемой модели была выбрана модель случайного леса. Для нее был проведен поиск гиперпараметров по сетке кросс-валидации. Были использованы только 2 параметра - количество деревьев и глубина каждого дерева, т.к. именно эти гиперпараметры дают прирост качества метрики. Тестируемая модель показала ошибку в среднем 610000 т.р. Это значение большое, но нужно учитывать, что данных было мало - около 427 строк, и модель может уловить не все зависимости в них. Для улучшения качества модели нужно увеличить количество данных, провести генерация новых признаков и переобучить модель.

7. ОПИСАНИЕ РЕЗУЛЬТАТОВ ТЕСТИРОВАНИЯ МОДЕЛИ

Было создано веб-приложение на основе REST API и фреймворка FLASK. Приложение содержит сохраненную модель, препроцесинг данных, пайплайн обработки полученных входных данных в виде json-файла и вывода результата предсказания по ним. Проводить тестирование приложения можно несколькими способами: через терминал bash, программу postman, внутри проекта по работе с данными. Все способы прошли проверку и показали идентичный результат. По итогу было проведено тестирование приложение внутри рабочей тетради в jupyter lab. Там было прописана функция, которая на вход принимала параметры пользователя и выводила на экран предсказанную стоимость квартиры. Для сравнения можно сопоставить полученные результаты с актуальным значением выбранных объектов. Как показали резульаьы сравнения, отклонение от реальной цены составило не более 70000 т.р., в среднем 40000т.р., что является отличным показателем для модели с полу-базовым подходом.

8. ВЫВОДЫ

Работа по созданию прототипа модели предсказания стоимости квартиры показала хороший результат. Был проведен парсинг всех имеющихся данных по квартирам в г.Магнитогорск, качественная предобработка признаков, корректное моделирование и обучение нескольких моделей с перебором гиперпараметров по сетке. Результат

проведенной работы показал отличный результат по сравнению со значением метрики. На основе полученных результатов можно предложить несколько рекомендаций:

- собрать больше данных с разных источников;
- провести дополнительный feature engineering;
- переобучить модель и провести перебор большего количества параметров с несколькими функциями потерь.

ЗАДАЧА 2

На основе данных, собранных в задании 1 разбить исходную выборку на кластеры методами кластерного анализа. Цель задания определить район продажи квартиры на основе данных о ценах, площадях и количестве комнат. Сравнить полученный результат с реальным расположением квартир по районам. В качестве района рассматривать Ленинский, Орджоникидзевский, Правобережный, Орджоникидзевский (Левый берег) и Ленинский (левый берег).

Для кластерного анализа был применен метод Kmeans. Были созданы кластеры по районам, затем полученные районы по кластерам были сопоставлены с актуальными и посчитана точно совпадения - 39%.

Разбить данные на кластера так, чтобы каждому кластеру соответствовал только один район - невыполнимая задача. Проблема заключается в том, что алгоритмы кластеризации стремятся группировать данные на основе схожих признаков, и они не знают о наличии определенных уникальных значений районов, которые не были определены в список выбранных для кластеризации признаков. Кластеризация, как правило, используется для выявления общих паттернов в данных и различных группировок для снижения дисперсии, а не для точного соответствия предсказаний с реальными данными в рамках жестких требований.