МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
имени М.В.Ломоносова
Факультет вычислительной математики и кибернетики

Задание 3. Ансамбли алгоритмов для решения задачи регрессии. Веб-сервер

Подготовил:
студент 317 группы ВМК МГУ
Долгушев Глеб Дмитриевич

December 2024

Содержание

1 Введение

Цель данного задания — изучить устройство различных методов построения ансамблей (случайный лес и градиентный бустинг).

Более подробно представим содержание работы в виде следующих задач:

1. Провести анализ и предобработку выбранного набора данных для предсказания стоимости дома
2. Разбить выборку на обучение и контроль
3. Реализовать случайный лес и градиентный бустинг
4. Исследовать как влияют на работу реализованных методов на выбранном наборе данных различные параметры обучения, рассмотрев такие критерии как: значение функции ошибок **RMSE** на контрольной выборке и время обучения модели

2 Предобработка

2.1 Исследование целевой переменной

Перед началом предобработки данных было изучено, как распределена целевая переменная (Рис. 1). По гистограмме видно, что распределение является несимметричным, кроме того, оно является похожим на логнормальное. Чтобы проверить эту гипотезу, было также построено распределение логарифма целевой переменной (Рис. 2).

Рис. 1: Распределение целевой переменной в изучаемой выборке.

Рис. 2: Распределение логарифма целевой переменной в изучаемой выборке.

Действительно, полученный результат сильно схож с нормальным распределением, обрезанным с одного конца.

Однако, чтобы убедиться окончательно в необходимости преобразования целевой переменной или отсутствия таковой, обучим две базовые модели. За бэйзлайн был выбран градиентный бустинг над решающими деревьями с простым набором параметров (деревья глубиной 10 и размерностью подвыборки признаков, равной половине размерности всего признакового пространства). Далее, одна модель была обучена предсказывать непреобразованную целевую переменную, другая - предсказывать прологарифмированную. После преобразования предсказаний первой модели был получен график Рис. 3.

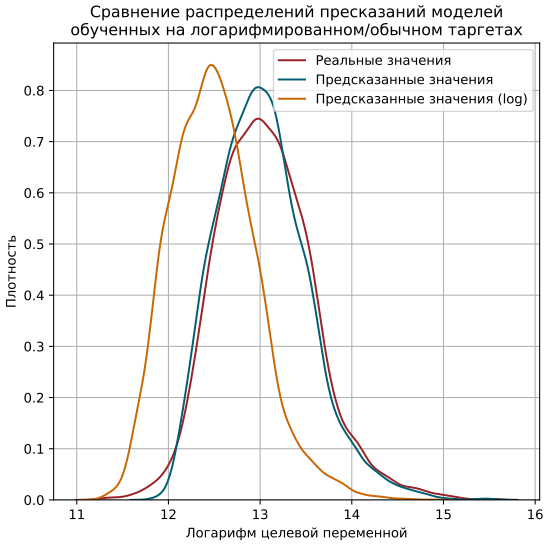

Рис. 3: Сравнение предсказаний базовых моделей на валидационной выборке.

Как можно заметить, выбор в качестве таргета преобразованной целевой переменной смещает распределе-

ние предсказаний относительно корректного. Таким образом, оптимальным оказалось решение оставить целевую переменную без преобразований.

2.2 Преобразования признаков

Первая проблема, которая была замечена при изучении данных признаков, - большое количество уникальных значений у `yr_built` и `yr_renovated`. Хорошим решением оказалось перевести значения данных признаков из конкретного года в десятилетие, в котором был построен или отремонтирован дом (для `yr_renovated` отсутствие ремонта было вынесено в отдельную категорию). Как можно видеть на Рис. 23, данный подход позволил получить значимые признаки с небольшим числом категорий.

Далее были преобразованы признаки `sqft_lot` и `sqft_lot15`, а именно прологарифмированы. После преобразования визуализация данных фичей (Рис. 22) становится более интерпретируемой.

2.3 Удаление шумовых признаков

Для выявления малозначимых или шумовых признаков был проведен анализ ящиков с усами для большинства категориальных признаков (подробнее на Рис. 23), а также была исследована корреляционная матрица признаков. В целях уменьшения переобучения признак `id` игнорировался, остальные более подробно рассмотрены ниже.

Отдельно были изучены свойства признака `date`. Для этого из него были выделены 2 новых признака: `year` и `month`. Их визуализация представлена на Рис. 4.

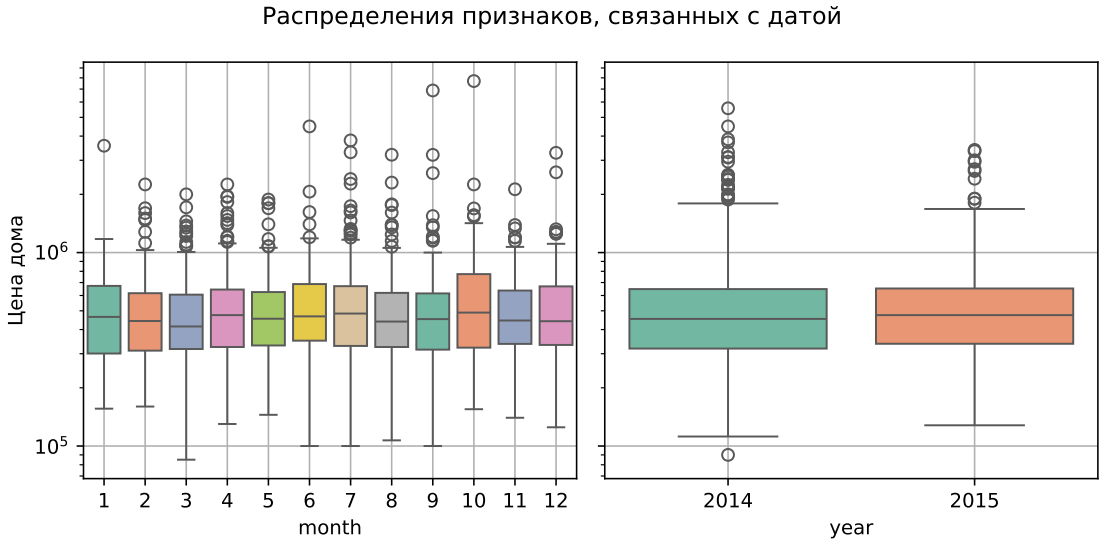

Рис. 4: Ящики с усами для выделенных из признака `date` признаков.

Как можно видеть на графиках, данные признаки не несут в себе никакой существенной информации (колебания цены минимальны), поэтому было решено не учитывать их при обучении модели (как и признак `date` в целом).

Далее рассмотрим корреляционную матрицу. Из Рис. 5 были сделаны следующие выводы:

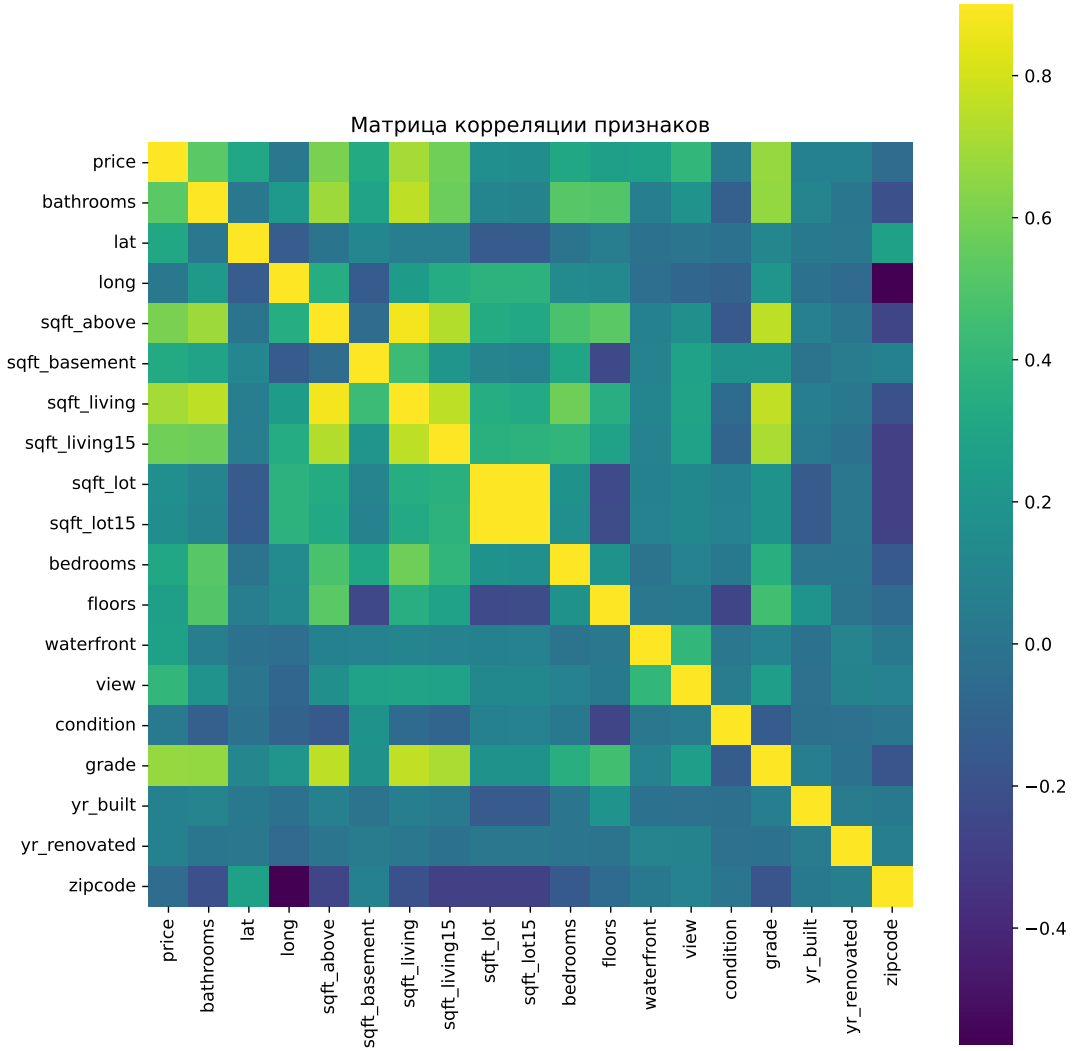

Рис. 5: Корреляционная матрица преобразованных признаков

1. Признаки в парах (`sqft_lot`, `sqft_lot15`), (`sqft_living`, `sqft_living15`), (`sqft_above`, `sqft_living`) очень сильно скоррелированы, поэтому из каждой пары оставлен для рассмотрения только 1 признак (кроме пары (`sqft_above`, `sqft_living`), так как было сделано предположение, что различие может оказаться важным в некоторых ситуациях).

2. Все выделенные признаки, кроме `long`, `yr_built`, `yr_renovated`, `condition`, `zipcode` имеют неплохую корреляцию с таргетом

3. Признаки `yr_built`, `yr_renovated`, `condition`, `zipcode` имеют низкую корреляцию из-за своей структуры (это категориальные признаки, для которых категории не имеют определенного на них порядка)

Отдельно проанализирована значимость признака `zipcode`. На Рис. 20 можно видеть зависимость значения данной фичи от расположения дома на карте. Можно сделать вывод, что, скорее всего, данный признак отвечает за почтовый код района или нечто похожее. Отдельно построив зависимость цены дома от его местоположения (Рис. 21), было подтверждено, что это важный фактор для предсказания целевой переменной. Таким образом, признак `zipcode` предоставляет нам возможность использовать готовое разбиение города на районы, что может сильно улучшить качество регрессии и обобщающую способность алгоритма (анализ только координат приведет к ухудшению работы модели).

2.4 Разбиение выборки

Статистические данные на выбранных для обучения модели числовых и категориальных признаках представлены на Рис. 22 и Рис. 23 соответственно.

Для проведения экспериментов выборка была перемешана и разбита на тренировочную и контрольную (валидационную) в соотношении 7:3. Категориальные признаки были закодированы при помощи `OneHotEncoder`. Итоговое количество признаков после кодировки: 144.

3 Эксперименты

Все дальнейшие эксперименты подразумевают обучение модели на созданной тренировочной выборке и оценку на валидационной. Изучаемые величины - значение критерия качества **RMSE** и время работы алгоритма (то есть время его обучения).

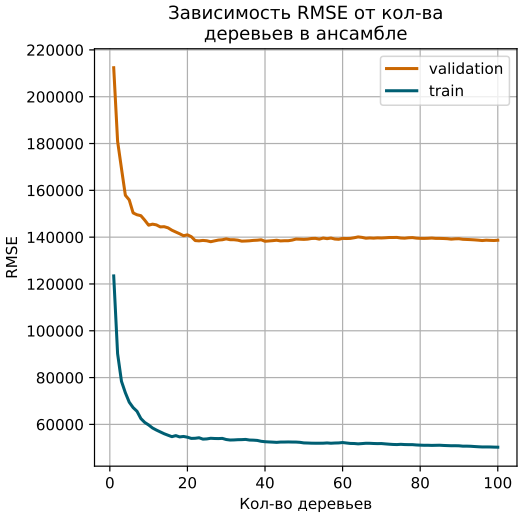

Рис. 6: Зависимость RMSE от кол-ва деревьев в ансамбле для алгоритма случайный лес.

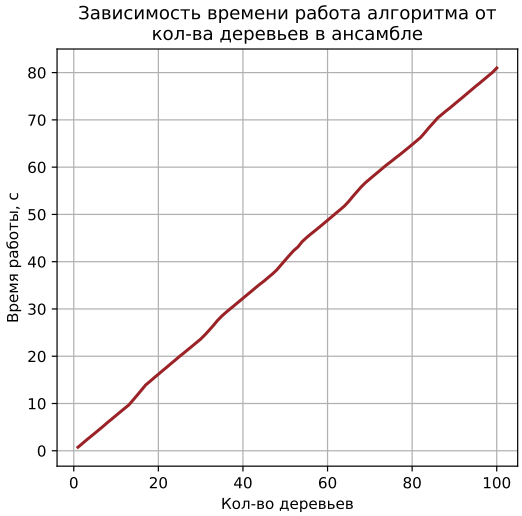

Рис. 7: Зависимость времени обучения от кол-ва деревьев в ансамбле для алгоритма случайный лес.

3.1 Исследование поведения алгоритма случайный лес

В данном пункте исследуется зависимость выбранных выше величин от того, с какими параметрами обучен случайный лес.

Предоставим краткое описание алгоритма. Случайный лес подразумевает обучение нескольких решающих деревьев на выборках, бутстрапированных из изначального набора данных. Предсказание осуществляется усреднением предсказаний всех решающих деревьев леса.

3.1.1 Количество деревьев в ансамбле

Дизайн данного эксперимента был построен следующим образом: так как в процессе обучения случайного леса на N-ой эпохе алгоритм представляет собой случайный лес из N решающих деревьев, то удобнее всего получать желаемый результат из истории обучения модели (таким образом получится избежать обучения нескольких алгоритмов).

Результаты эксперимента представлены на Рис. 6 и Рис. 7.

По графикам видно, что оптимальное значение **RMSE** на валидационной выборке достигается при включении в ансамбль приблизительно 30 деревьев, хотя сильного переобучения не наблюдается. Время работы алгоритма растет линейно (обучение каждого решающего дерева занимает приблизительно одно и то же время).

3.1.2 Размерность подвыборки признаков для одной вершины дерева

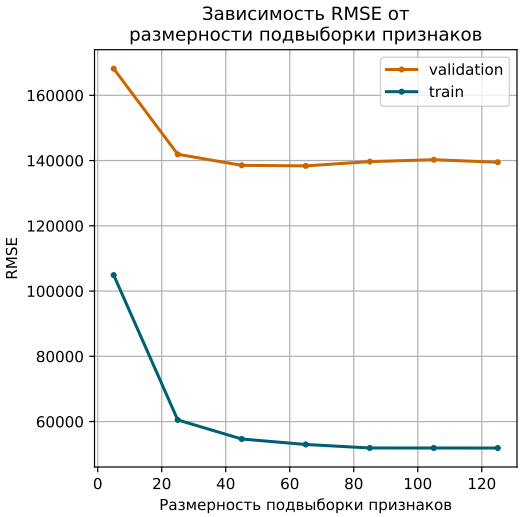

Рис. 8: Зависимость RMSE от размерность подвыборки признаков для одной вершины дерева для алгоритма случайный лес.

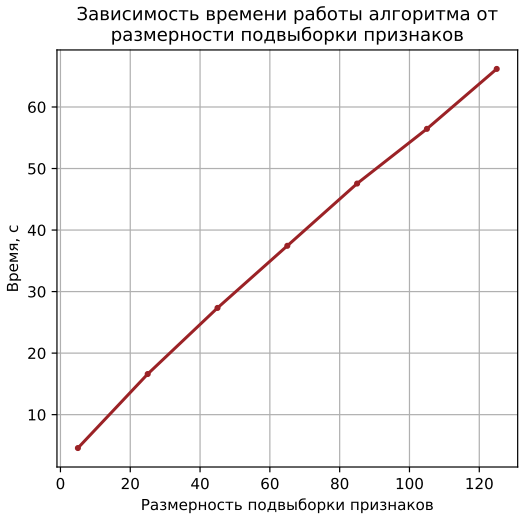

Рис. 9: Зависимость времени обучения от размерность подвыборки признаков для одной вершины дерева для алгоритма случайный лес.

В процессе данного эксперимента модель обучалась с разными значениями

параметра `max_features`, который отвечает за размерность случайной подвыборки признаков, которая рассматривается при поиске лучшего разбиения на каждом узле дерева. Результаты представлены на Рис. 8 и Рис. 9.

Сохранилась линейная зависимость времени обучения (так как при поиске наилучшего разбиения решающее дерево проходит по всем признакам из случайной подвыборки размера `max_features`). Кроме того, отлично проиллюстрирована возможность переобучения модели при большом значении параметра. Данное явление обусловлено тем, что в процессе эксперимента глубина дерева ограничена (`max_depth = 20`). Таким образом, при учете всех признаков в каждом дереве мы получаем, что все элементы ансамбля выбирают одни и те же, наиболее оптимальные с точки зрения разбиения пространства, признаки (причем ограниченное количество), становятся похожими и не могут учесть более тонкие зависимости. Если значения параметра `max_features` мало, то каждый из элементов ансамбля может нести в себе только очень общую зависимость целевой переменной от признаков, что опять же ухудшает качество работы случайного леса.

3.1.3 Максимальная глубина дерева

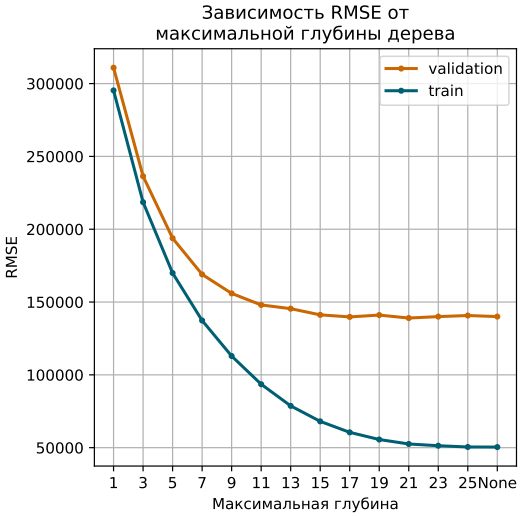

Рис. 10: Зависимость RMSE от максимальной глубины дерева для алгоритма случайный лес.

Рис. 11: Зависимость времни обучения от максимальной глубины дерева для алгоритма случайный лес.

В процессе данного эксперимента модель обучалась с разными значениями параметра `max_depth`, который отвечает за максимальную глубину элементов ансамбля. Результаты представлены на Рис. 10 и Рис. 11 (None означает дерево неограниченной глубины).

Из графиков видно, что улучшение качества на валидационной выборке прекращается после достижения значения глубины 15, но не наблюдается значительного переобучения даже при неограниченной глубине деревьев. Такой результат опять же обусловлен тем, что существует оптимальная сложность модели, после которой качество перестает улучшаться или начинает ухудшаться.

Отсутствие заметного переобучения может говорить о чистоте данных (малое число выбросов) и хорошей предварительной обработке признаков. Время обучения также увеличивается с ростом сложности модели.

3.2 Исследование поведения алгоритма градиентный бустинг

В данном пункте исследуется зависимость выбранных выше величин от того, с какими параметрами обучен градиентный бустинг.

Предоставим краткое описание алгоритма. Градиентный бустинг подразумевает последовательное обучение нескольких решающих деревьев на выборках, бутстрапированных из начального набора данных, каждое из которых предсказывает ошибку построенного на предыдущем шаге алгоритма на обучающей выборке. Предсказание после N-ой эпохи обучения осуществляется суммированием предсказаний N первых решающих деревьев.

3.2.1 Количество деревьев в ансамбле

Дизайн данного эксперимента аналогичен схожему для изучения поведения случайного леса (максимальная глубина дерева выбрана равной 10, так как бустинг - это ансамбль из слабых алгоритмов). Результаты представлены на Рис. 12 и Рис. 13.

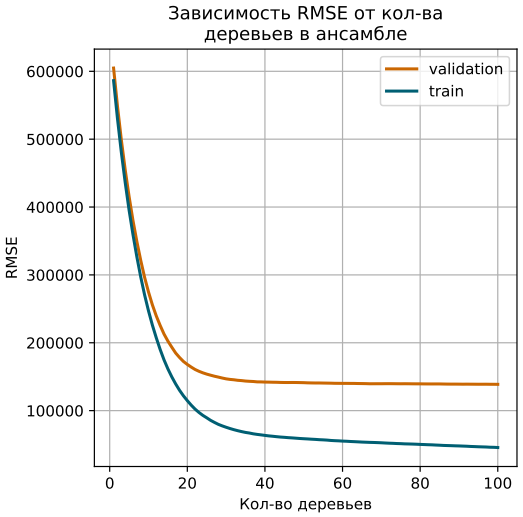

Рис. 12: Зависимость RMSE от кол-ва деревьев в ансамбле для алгоритма градиентный бустинг.

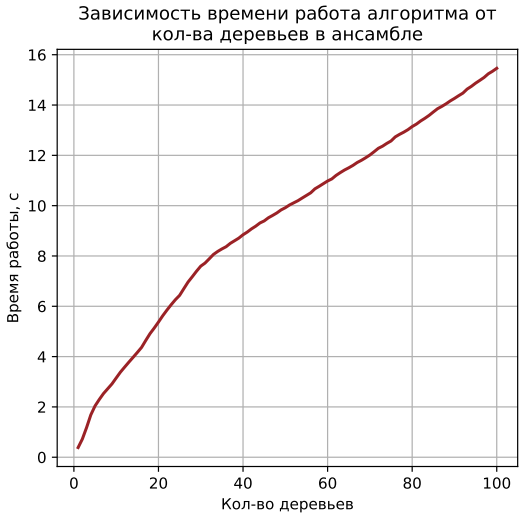

Рис. 13: Зависимость времени обучения от кол-ва деревьев в ансамбле для алгоритма градиентный бустинг.

Результаты схожи с полученными для случайного леса, но качество перестает улучшаться уже после включения 30 деревьев в ансамбль (а не после 50).

3.2.2 Размерность подвыборки признаков для одной вершины дерева

В процессе данного эксперимента модель обучалась с разными значениями параметра `max_features`, который отвечает за размерность случайной подвыборки признаков, которая рассматривается при поиске лучшего разбиения на каждом узле дерева. Результаты представлены на Рис. 14 и Рис. 15.

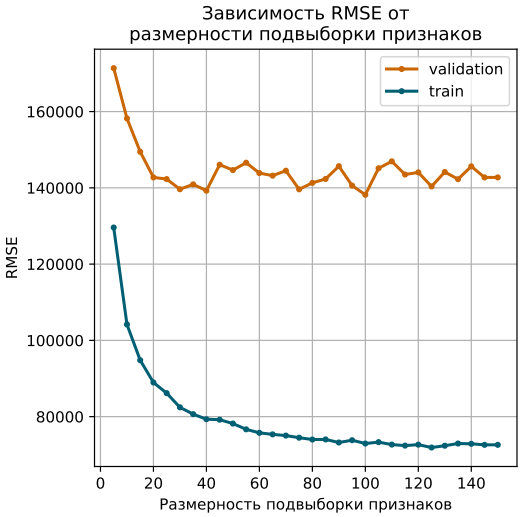

Рис. 14: Зависимость RMSE от размерность подвыборки признаков для одной вершины дерева для алгоритма градиентный бустинг.

Рис. 15: Зависимость времени обучения от размерность подвыборки признаков для одной вершины дерева для алгоритма градиентный бустинг.

Сохраняется подобие линейной зависимости времени обучения от сложности модели. В отношении же параметра `max_features` по сравнению с аналогичным результатом для случайного леса наблюдается значительное переобучение (при увеличении числа признаков качество ухудшается в большей мере). Кроме того, оптимальным оказывается размерность подвыборки признаков, приблизительно равная 30, что подтверждает необходимость использовать как элементы ансамбля в градиентном бустинге именно слабые регрессоры.

3.2.3 Максимальная глубина дерева

В процессе данного эксперимента модель обучалась с разными значениями параметра `max_depth`, который отвечает за максимальную глубину элементов ансамбля. Результаты представлены на Рис. 16 и Рис. 17 (None означает дерево неограниченной глубины).

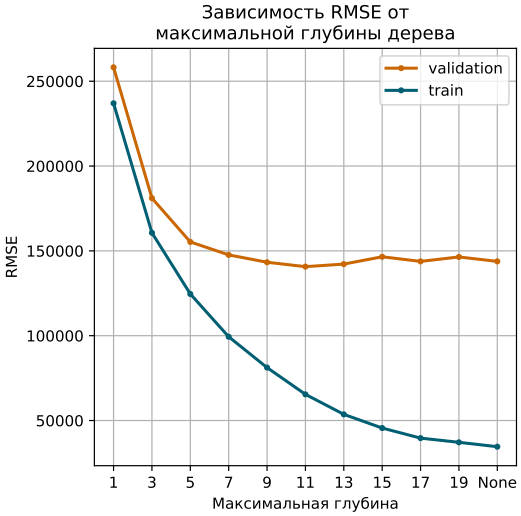

Рис. 16: Зависимость RMSE от максимальной глубины дерева для алгоритма градиентный бустинг.

Рис. 17: Зависимость времни обучения от максимальной глубины дерева для алгоритма градиентный бустинг.

Результаты полностью схожи с аналогичными, полученными для случайного леса, за исключением того, что значение оптимальной глубины решающих деревьев меньше (равно 11) и более заметна тенденция к переобучению деревьев большой глубины.

3.2.4 Коэффициент скорости обучения

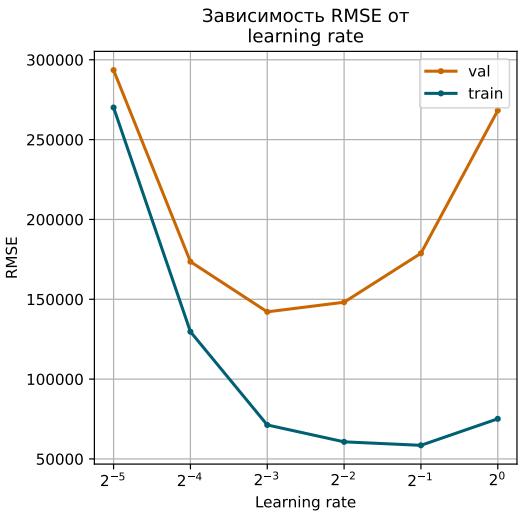

Рис. 18: Зависимость RMSE от `learning rate` для алгоритма градиентный бустинг.

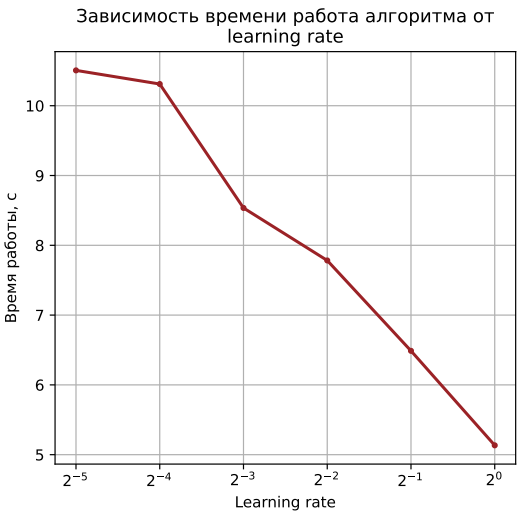

Рис. 19: Зависимость времени обучения от `learning rate` для алгоритма градиентный бустинг.

В процессе данного эксперимента модель обучалась с разными значениями

параметра `learning_rate`, который определяет шаг обучения, используемый для обновления модели на каждой эпохе бустинга. Результаты представлены на Рис. 18 и Рис. 19.

Данный параметр оказывает значительное влияние на качество работы алгоритма. При малом значении `learning_rate` каждое из решающих деревьев слишком незначительно обновляет предсказания модели. При большом значении, по аналогии с методом градиентного спуска происходят колебания вокруг оптимального решения.

4 Заключение

В результате работы был проведен всесторонний анализ выбранного набора данных, применены изученные методики построения новых признаков и выделения шумовых.

Также в ходе исследования было рассмотрено поведение алгоритмов случайный лес и градиентный бустинг в зависимости от параметров. Благодаря анализу полученных результатов было также изучено явления переобучения моделей, характеризующихся высокой сложностью. Кроме того, многократно было подтверждено, что для алгоритма градиентный бустинг желательно использовать слабые базовые алгоритмы.

5 Литература

Список литературы

[1] Онлайн учебник по машинному обучению от ШАД, URL: https://education.yandex.ru/handbook/ml

[2] Курс лекций по методам машинного обучения, К.В. Воронцов, URL: https://github.com/MSU-ML-COURSE/ML-COURSE-24-25/tree/main/slides/2_stream

6 Аппендикс

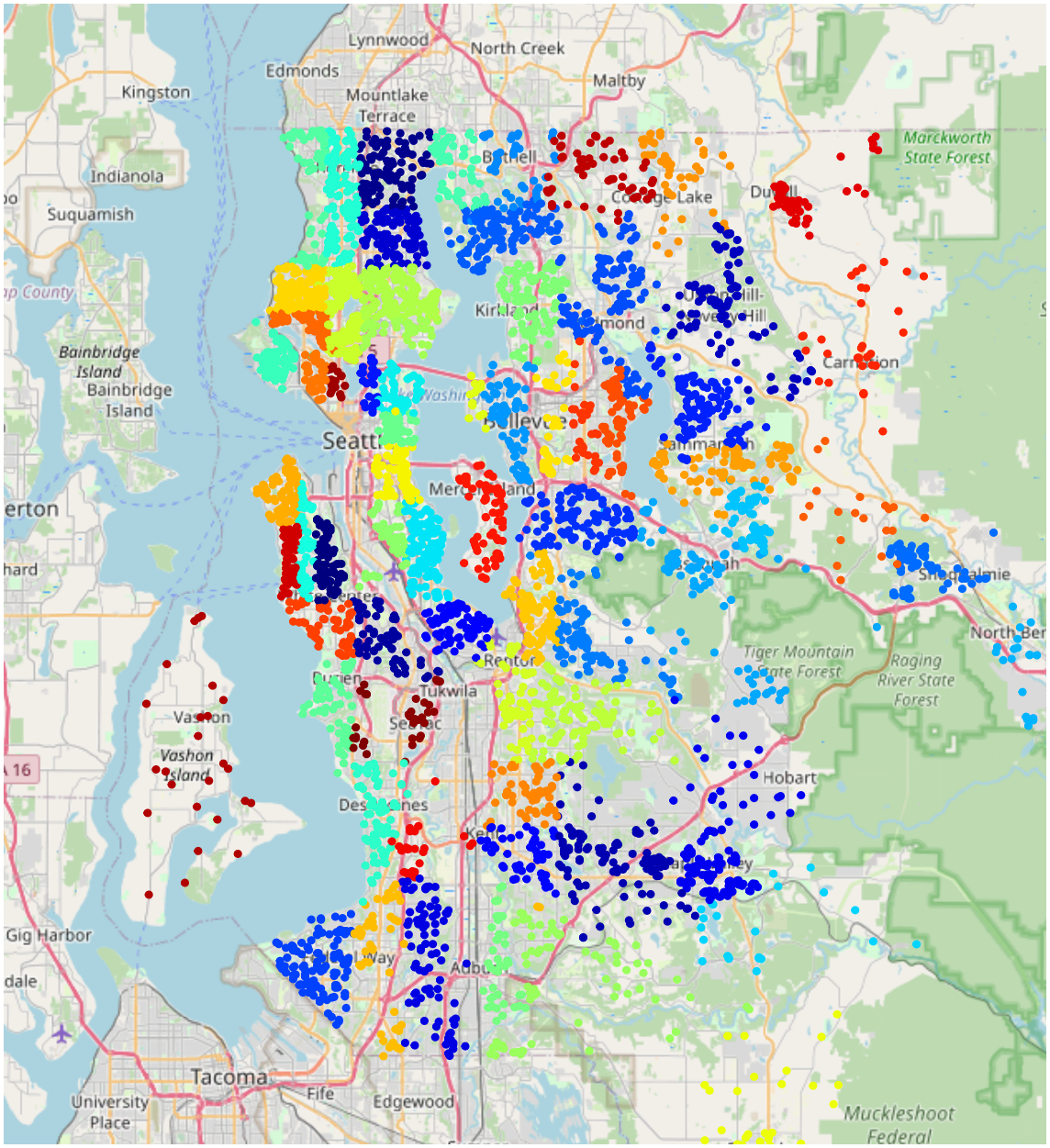

Рис. 20: Визуализация признака `zipcode`.

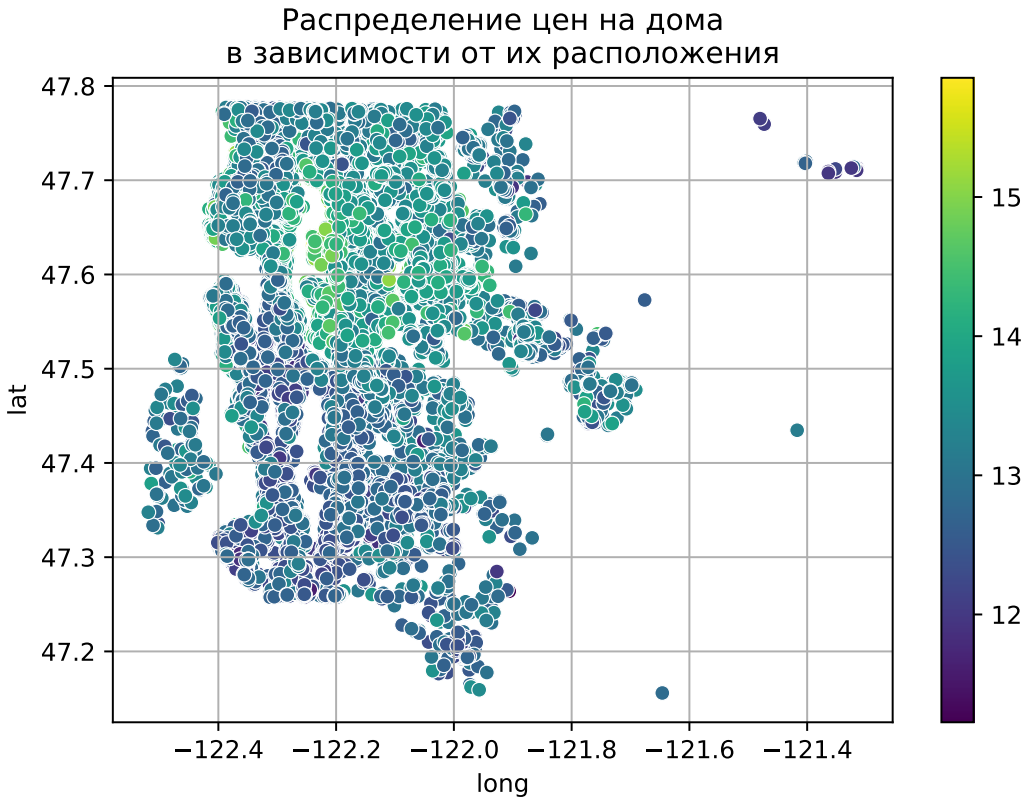

Рис. 21: Визуализация распределения значения целевой переменной в зависимости от расположения домов.

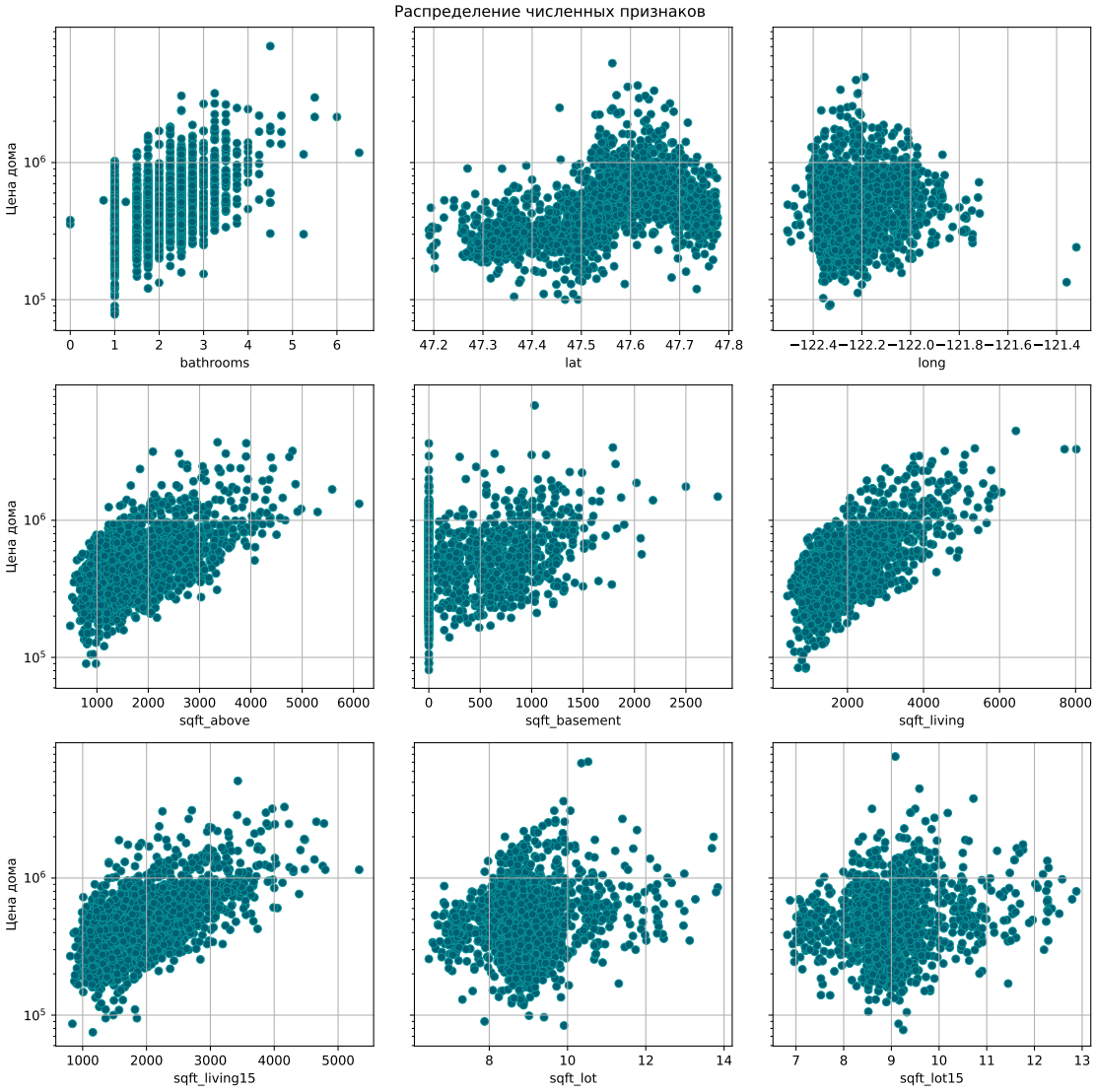

Рис. 22: Визуализация распределения численных признаков.

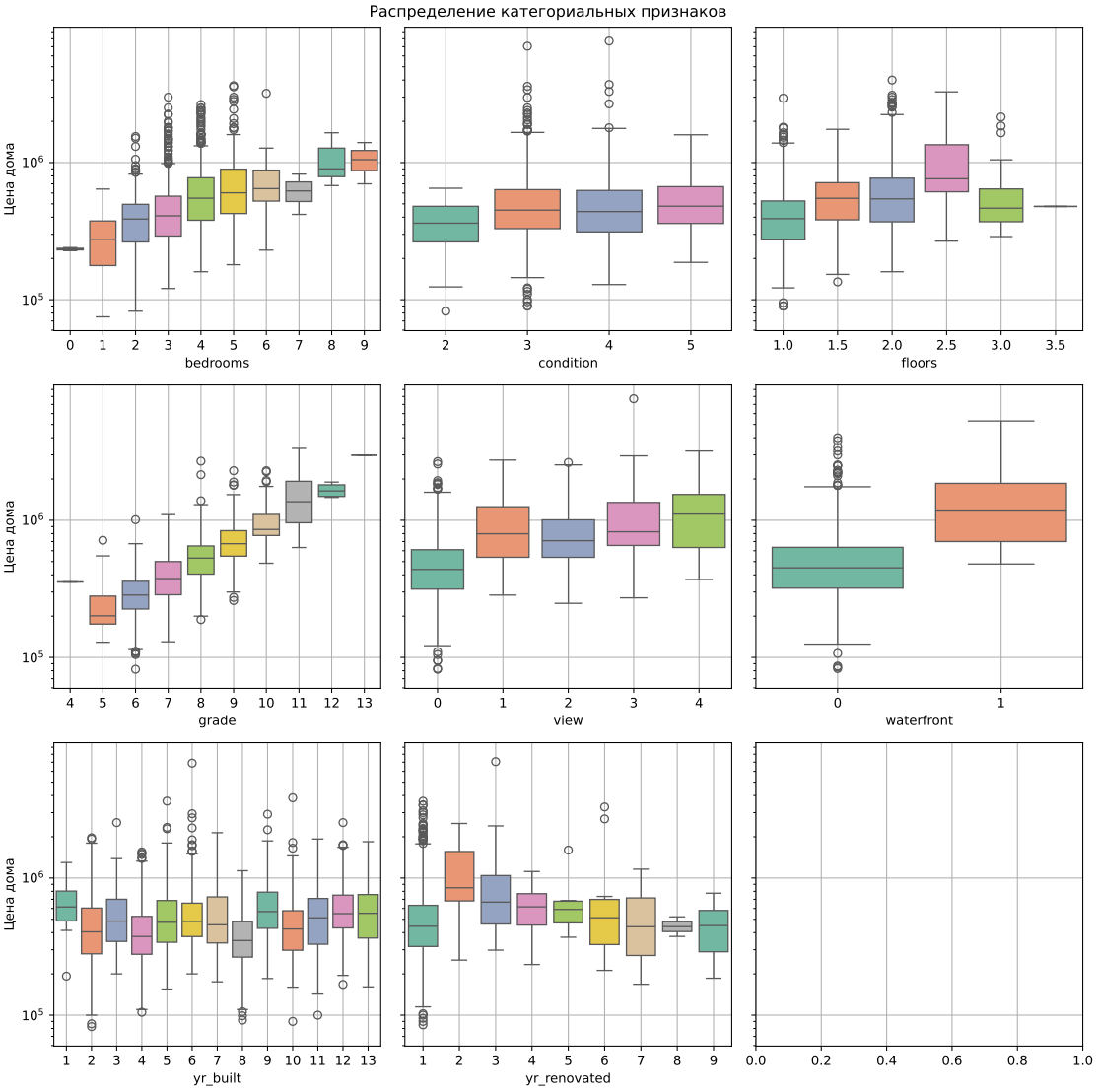

Рис. 23: Визуализация распределения категориальных признаков.