Отчет по заданию 3

«Ансамбли алгоритмов. Веб-сервер. Композиции алгоритмов для решения задачи регрессии.»

курс «Практикум на ЭВМ»
кафедра ММП ВМК МГУ

Михеев Борис, 317 группа

10 декабря 2021 г.

Формулировка задания

В данном задании требуется реализовать алгоритмы **случайный лес** и **градиентный бустинг** на языке `Python`, применить их для решения задачи регрессии по предсказанию цены на недвижимость с использованием соответствующего датасета, провести ряд экспериментов для исследования зависимости качества алгоритмов и времени их работы от используемых методов и их параметров. Также требуется создать веб-сервис с соответствующим функционалом взаимодействия с моделью и разместить проект в отдельном репозитории.

Постановка задачи

Имеется задачи регрессии, используемая метрика качества - RMSE :

$$\text{RMSE} = \sqrt{\frac{1}{l}\sum_{i=1}^{l}(y_i - a(x_i))^2}$$

где y_i - истинное значение целевой переменной для i-ого объекта, $a(x_i)$ - предсказание алгоритма для i-ого объекта, l - размер обучающей выборки.

Рассматриваются следующие ансамблевые методы: случайный лес и градиентный бустинг, являющиеся ансамблями решающих деревьев. Требуется применить их для соответствующей задачи регресии и исследовать зависимость качества алгоритмов и времени работы от таких гиперпараметров как число деревьев в ансамбле, число рассматриваемых признаков для одного дерева, максимальная глубина деревьев, темп обучения (в случае градиентного бустинга).

Предобработка данных

В исходных данных число объектов - 21613, число признаков - 21, включая столбец с целевой переменной (признак `price`). В датасете нет пропущенных значений. Все признаки имеют целочисленные типы (`int64`, `float64`) за исключением признака `date`, который имеет тип `object` (строка). Данный признак целесообразно преобразовать в тип `datetime` а затем в числовой формат, а именно разбить на отдельные признаки года, месяца, дня, дня недели. Так как в рассматриваемой задаче предсказывается цена дома, и признак date, вероятно, соответствует времени выставления дома на продажу, то данный вариант преобразования может иметь смысл, ибо некоторая сезонность или принадлежность конкретному временному периоду может иметь значение на рынке недвижимости и, соотвтетственно, будет влиять на цену. Стоит учесть и день недели, возможно в будни и выходные продажи идут по разному.

Также, разумеется, стоит исключить признак `id` как шумовой и не несущий никакой важной в задаче информации.

По числу уникальных значений признаков большинство из них можно отнести к вещественным, некоторые - к категориальным (по смыслу названия и числу уникальных значений: признаки `condition`, `view`, `waterfront`, `floors` и т. д., а также признаки, сгенерированные после преобразования признака `date`). В виду относительно небольшого числа категориальных признаков можно не проводить их специальную обработку, будем рассматривать их как вещественные. Подробная информация о числе значений признаков представлена в приложенном к работе `ipynb`-файле.

Также в данных не содержится признаков, близких к константным, т. е. со стандартным отклонением, близким к 0. Таким образом, нет признаков, независимых от целевой переменной. Подробная информация о среднем и стандартном отклонении признаков представленна в приложенном к работе `ipynb`-файле.

Разобъем исходную выборку на обучающую и валидационную в отношении 7/3 с перемешиванием объектов.

Случайный лес

Зависимость от числа деревьев

Так как в данном алгоритме предсказание проводится путем усреднения предсказаний базовых алгоритмов, при увеличении числа деревьев разброс модели будет уменьшаться, и алгоритм не будет сильно переобучаться. Проведем исследование при неограниченной глубине деревьев и числе признаков, по которым выбирается оптимальное разбиение в каждой вершине, равном $\frac{l}{3}$, где l - число признаков в выборке (рекомендованное значение для регрессии). Число деревьев переберем от 1 до 1000. Стоит отметить, что деревья в данном случае строятся не параллельно, а последовательно. Ошибку измерим на обучающей и отложенной выборке.

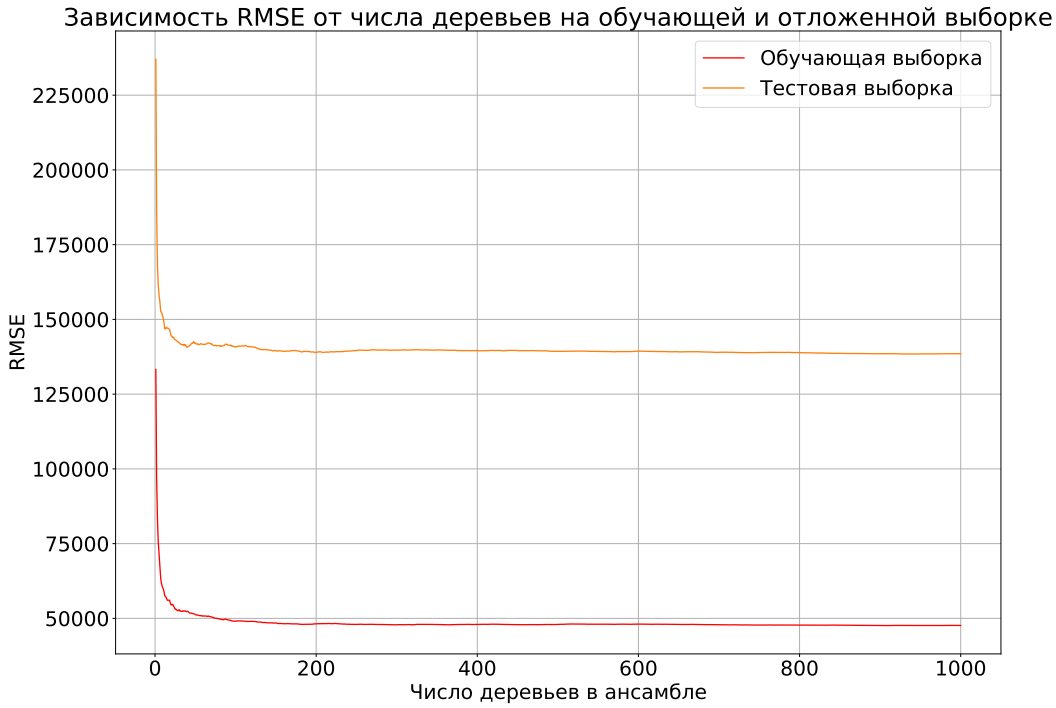

Рис. 1: Зависимость функции потерь RMSE от числа деревьев, алгоритм случайный лес

Из графиков видно, что с ростом числа деревьев функция ошибки на обучении и валидации монотонно убывает, после примерно 200 деревьев выходит на асимптоту. Значение ошибки на валидации ожидаемо больше, чем на обучении. Оптимальным числом базовых алгоритмов с точки зрения ошибки RMSE на валидации оказалось значение 238 с RMSE = 134345.616, на обучении - 250 с RMSE = 48163.959 . Стоит отметить, что на контрольной выборке после достижения оптимума функция ошибки начинает незначительно возрастать, что свидетельствует о переобучении, но затем выходит на асимптоту. При большем количестве деревьев ошибка незначительно уменьшается в силу уменьшения разброса модели. В качестве оптимального значения числа алгоритмов для использования в дальнейшем можно рассмотрть значение 250.

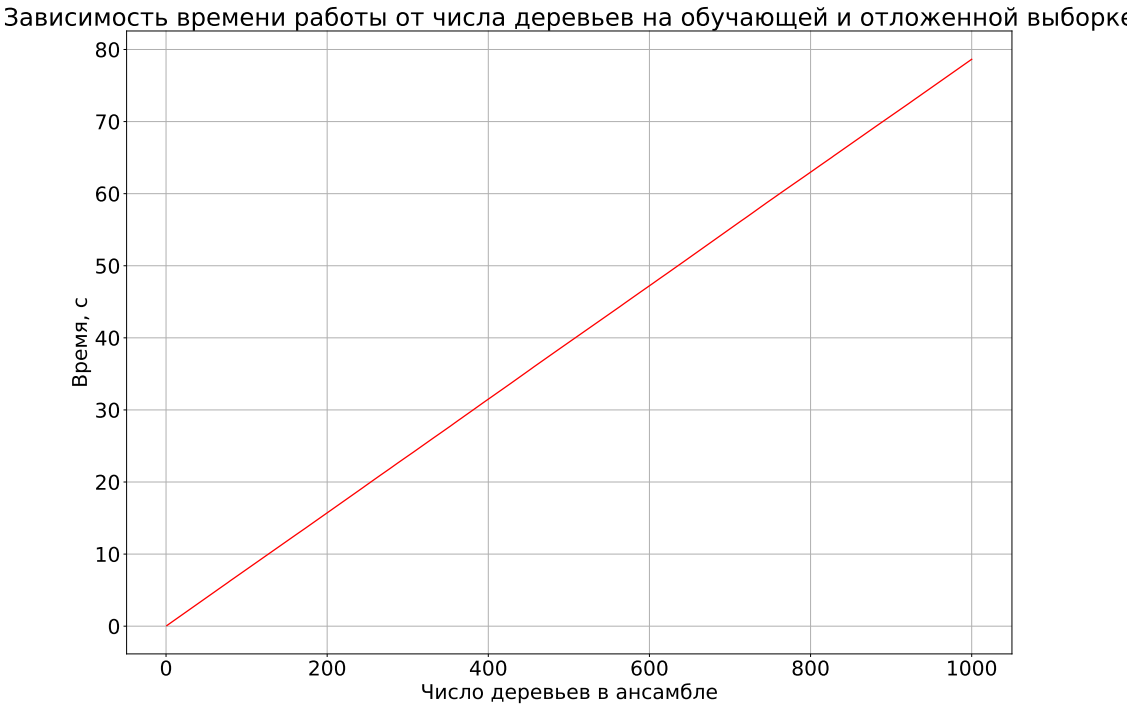

Рис. 2: Зависимость времени работы от числа деревьев, алгоритм случайный лес

Время работы алгоритма растет линейно с ростом числа деревьев, что логично, так как рассматриваемый метод является ансамблевым.

Зависимость от числа признаков для одного дерева

Исследуем зависимость значения RMSE и времени работы от размерности подвыборки признаков для одного дерева. Число деревьев примем равным 250 как оптимальное на основании предыдущих экспериментов, максимальную глубину деревьев возьмем неограниченной. Количество признаков переберем от 1 до 22. Стоит отметить, что соответствующий набор признаков выбирается случайно для каждого дерева.

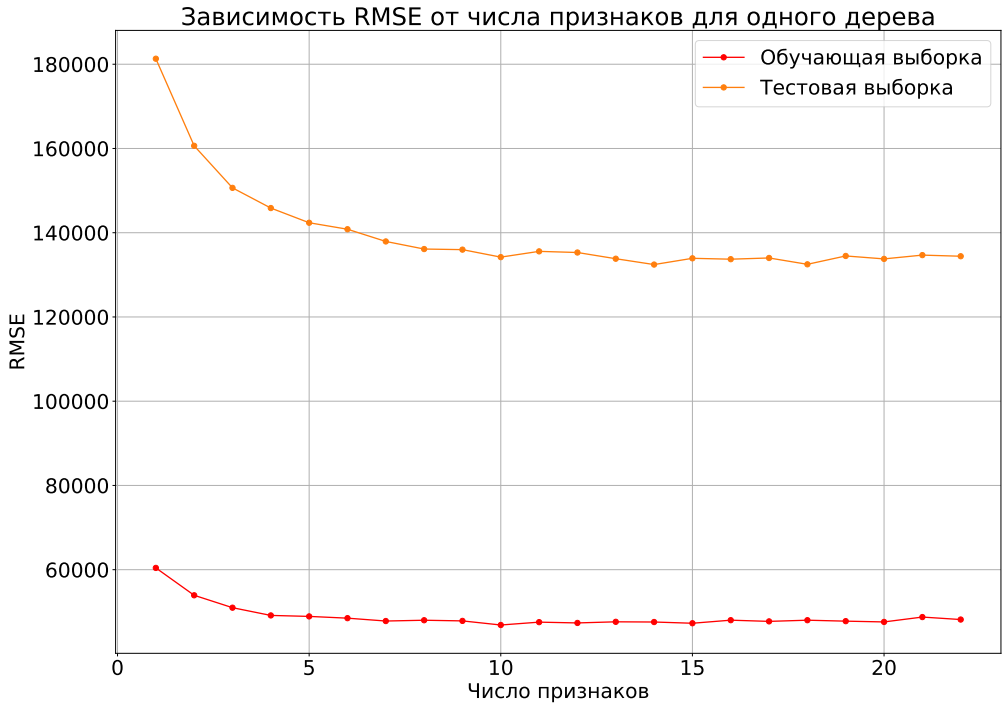

Рис. 3: Зависимость функции потерь RMSE от числа признаков для одного дерева, алгоритм случайный лес

В целом ошибка убывает с ростом числа признаков. Оптимальным значением для валидации оказалось 14 с RMSE = 132439.063, для обучения - 10 с RMSE = 46867.064. Присутствует некоторая немонотонность при определенных значениях числа признаков, при значениях, близких к числу признаков в выборке, заметно незначительное увеличение ошибки. Это может быть связано с тем, что в таком случае алгоритмы становятся менее рандомизированными, т. е. по сути более скоррелированными, что не способствует уменьшению разброса при использовании ансамблирования. Можно сказать, что по сути в таком случае просто используется бутстрап выборки. Примем за оптимальное число признаков, равное 14.

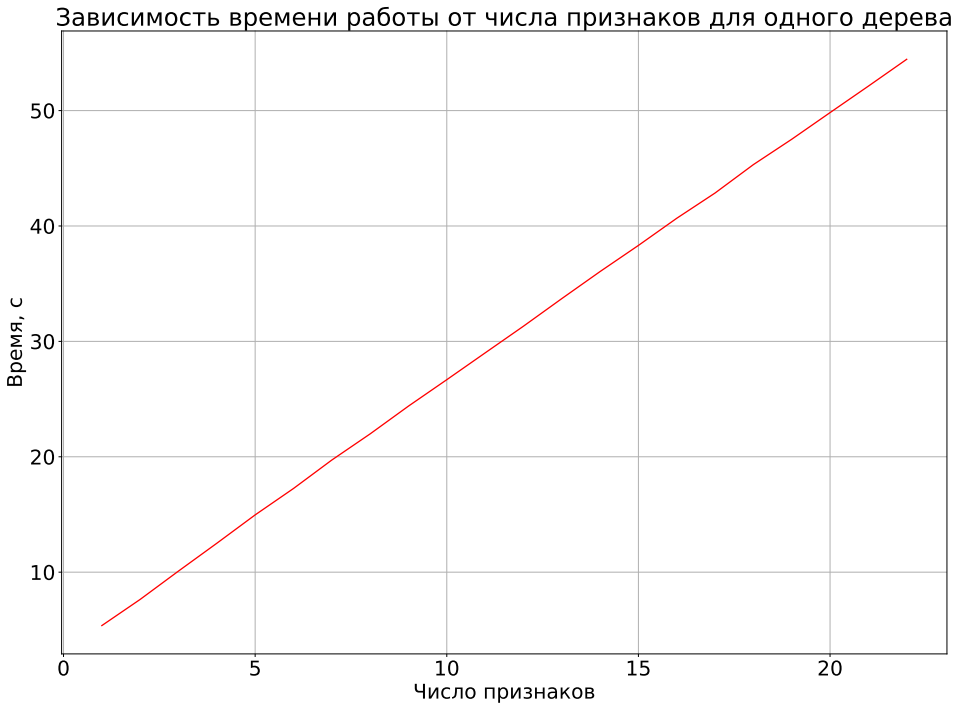

Рис. 4: Зависимость времени работы от числа признаков для одного дерева, алгоритм случайный лес

Время работы все так же изменяется линейно в зависимости от числа признаков.

Зависимость от максимальной глубины деревьев

Максимальная глубина деревьев отвечает за сложность модели, и чем она больше, тем более гибкой будет модель. Это позволит ей находить более сложные зависимости, но сделает ее более склонной к переобучению. Проведем эксперименты при следующих параметрах, выбранных как оптимальные по результатам предыдущих экспериментов: число деревьев - 250, число признаков для одного дерева - 14. Значения максимальной глубины рассмотрим от 1 до 30, а также отдельно рассмотрим случай неограниченной глубины.

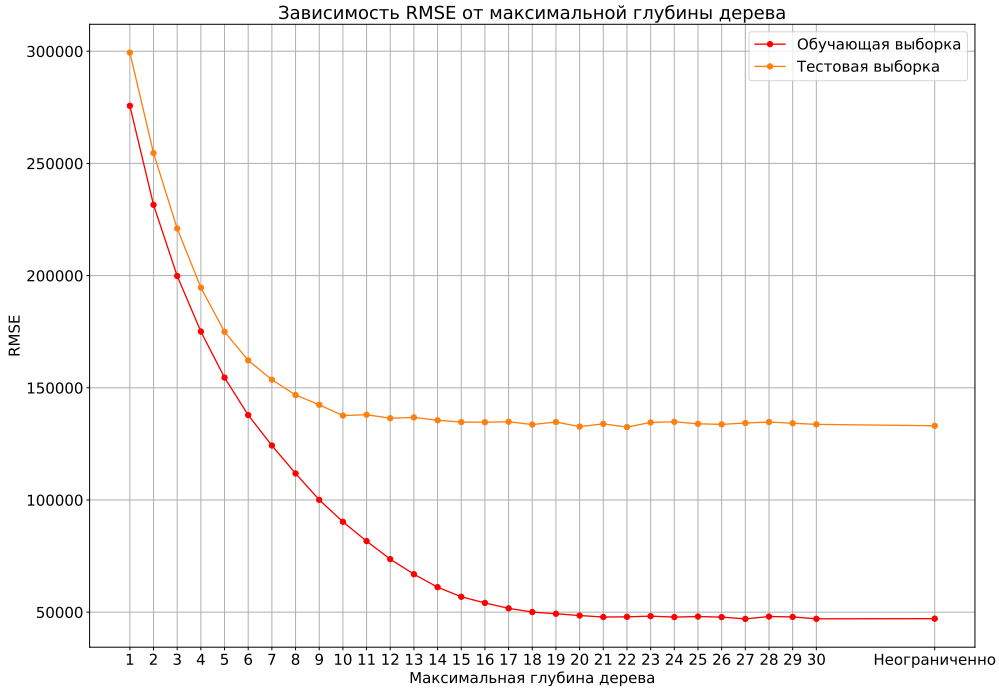

Рис. 5: Зависимость функции потерь RMSE от максимальной глубины деревьев, алгоритм случайный лес

В целом значение ошибки уменьшается с ростом гиперпараметра, после определенного значения выходит на асимптоту, причем для валидационной выборки это происходит при меньших значениях. Для малых значений глубины значение ошибки достаточно велико, модель недостаточно гибка. Оптимальным значением для валидации оказалось 22 с RMSE = 132482.988, для обучения - 27 с RMSE = 46999.098, хотя выход на асимптоту происходит раньше, примерно при 9 для валидации, при 19 для обучения. При больших значениях глубины наблюдается немонотонность. Возможно это связано с тем, что с ростом глубины деревьев модель сильнее переобучается. При неограниченной глубине значение ошибки близко к оптимальному. В целом ограничение на глубину имеет смысл также с точки зрения времени обучения.

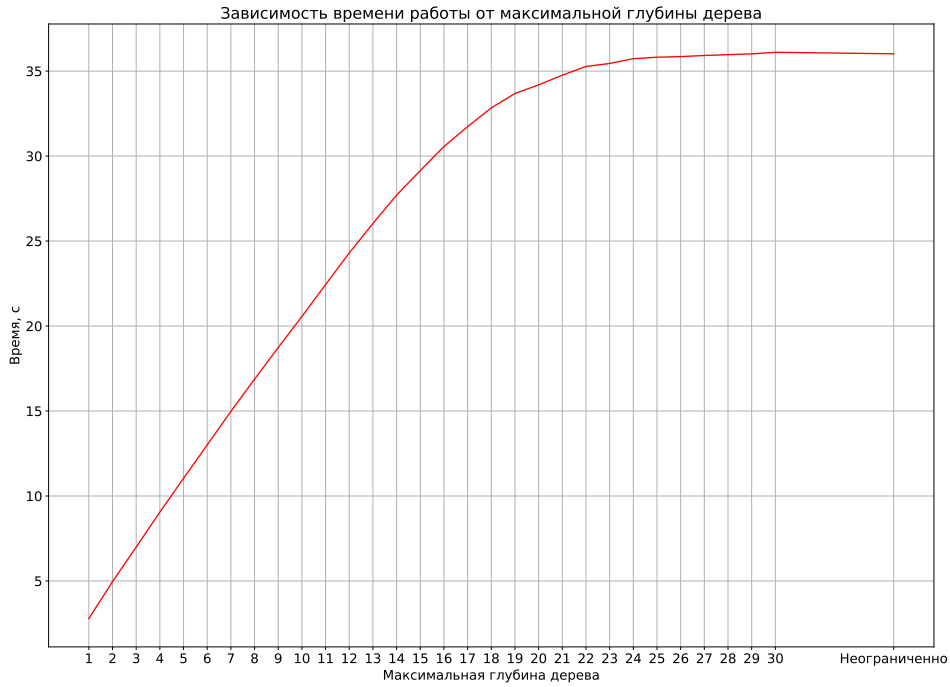

Рис. 6: Зависимость времени работы от максимальной глубины деревьев, алгоритм случайный лес

Время работы изменяется относительно линейно, а затем при больших значениях глубины выходит на асимптоту. Для неограниченной глубины время работы близко ко времени для глубины 30.

Градиентный бустинг

Зависимость от числа деревьев

В данном алгоритме базовые алгоритмы уже не являются независимыми, каждый следующий алгоритм старается исправить ошибки предыдущего. Поэтому при большом числе деревьев модель склонна к переобучению. Проведем эксперименты при неограниченной глубине деревьев и числе признаков для каждого дерева, равном трети от числа признаков в исходном датасете. Число деревьев переберем от 1 до 1000.

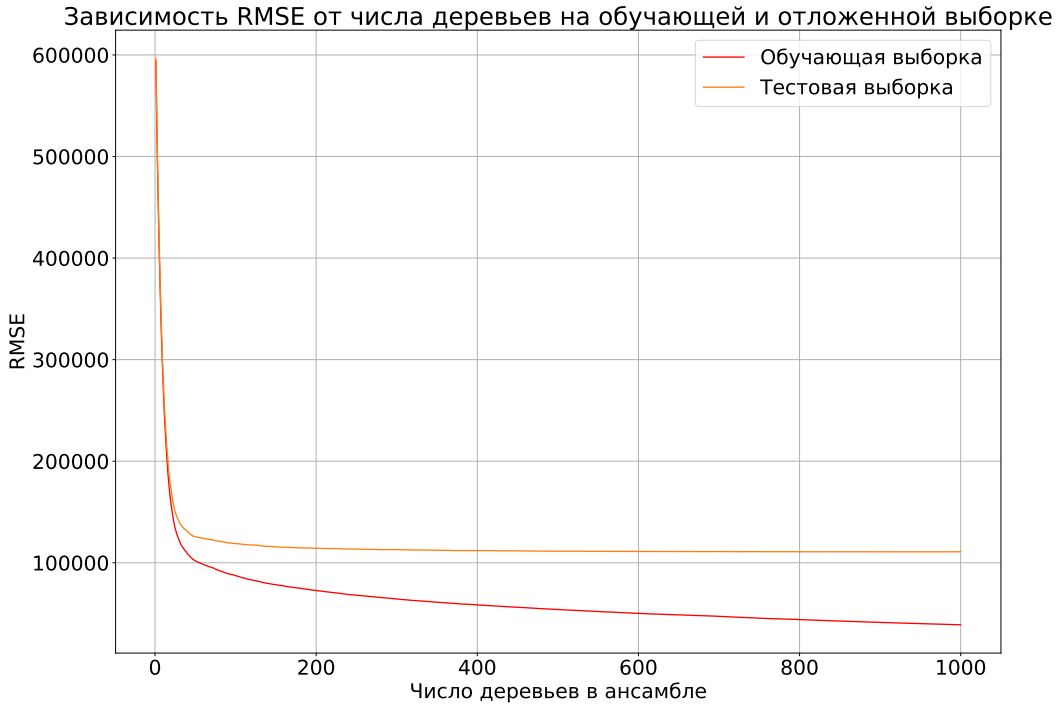

Рис. 7: Зависимость функции потерь RMSE от числа деревьев, алгоритм градиентный бустинг

Значение функции ошибки монотонно убывает с ростом числа деревьев, ошибка на валидации больше ошибки на обучении. На обучающей выборке функция ошибки стремится к нулю, для валидации выходит на асимптоту. В целом значение RMSE меньше, чем для случайного леса, однако при малом числе алгоритмов ошибка достаточно велика. Оптимальным числом для валидации оказалось 961 с RMSE = 110734.518, для обучения - 1000 с RMSE = 38969.845, хотя выход на асимптоту для валидации происходит раньше. Так как в рассматриваемом методе каждый следующий алгоритм пытается исправить ошибки предыдущего, т. е. они не являются независимыми, и предсказания умножаются в процессе на темп обучения, то вообще имеет смысл рассматривать зависимость ошибки от числа деревьев в совокупности с различными значениями темпа обучения, что будет сделано в последующих пунктах. В качестве оптимального значения для использования в дальнейшем возьмем 400.

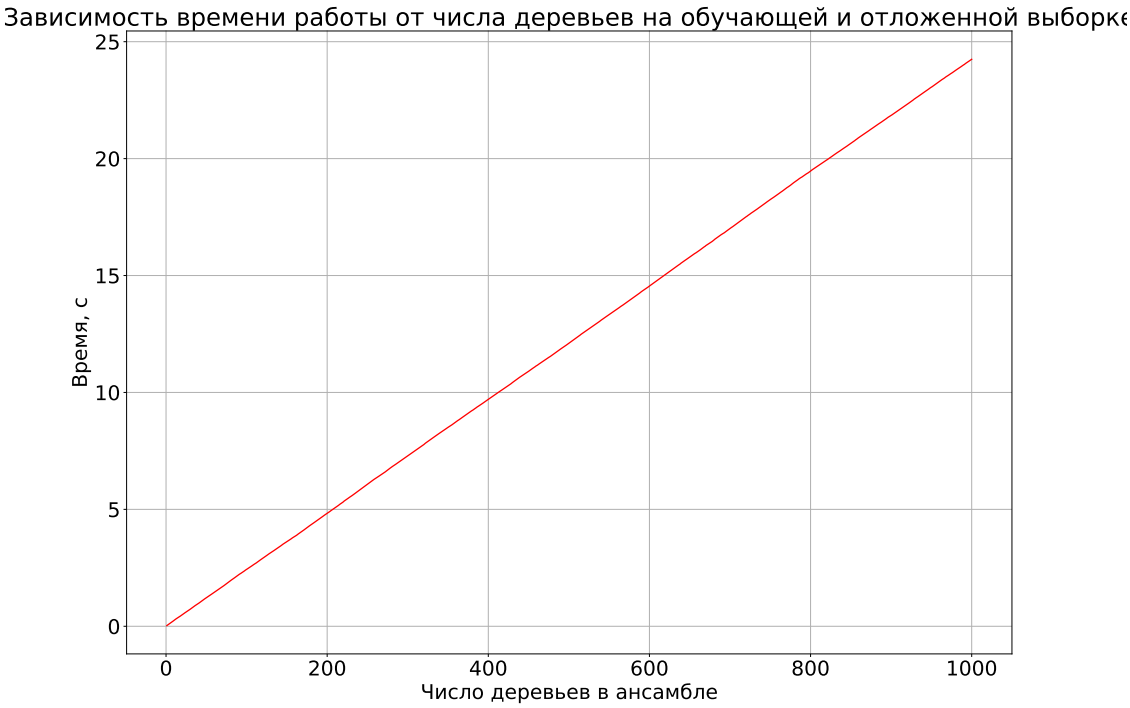

Рис. 8: Зависимость времени работы от числа деревьев, алгоритм градиентный бустинг

Время обучения все так же зависит линейно от числа деревьев, но обучение происходит быстрее, чем для случайного леса.

Зависимость от числа признаков для одного дерева

Проведем эксперименты при 400 деревьях и неограниченной глубине.

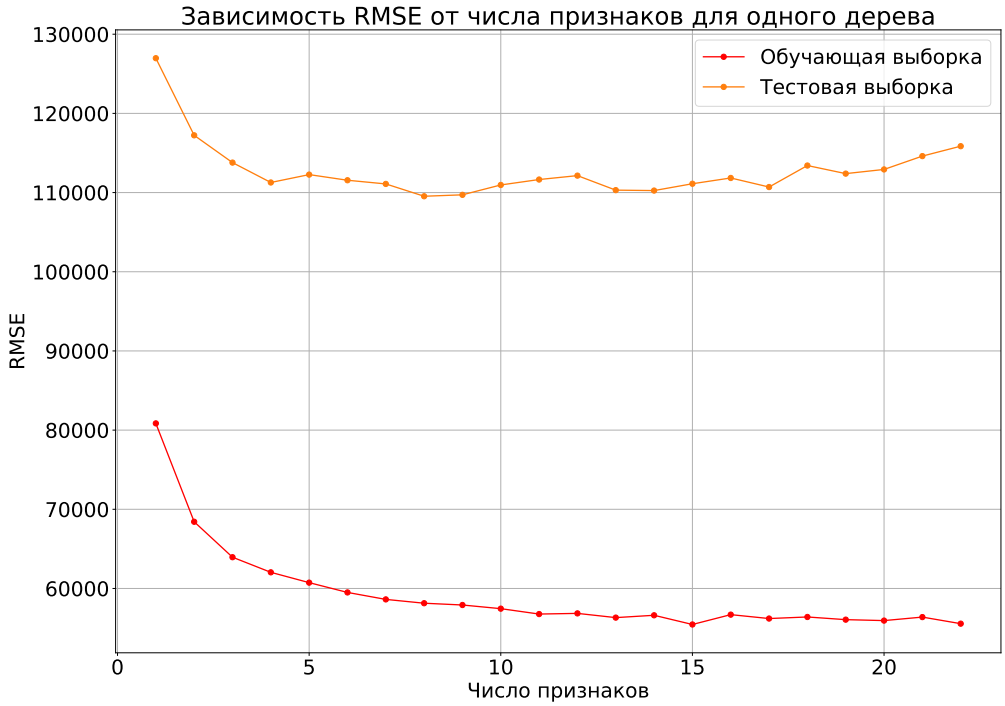

Рис. 9: Зависимость функции потерь RMSE от числа признаков для одного дерева, алгоритм градиентный бустинг

Наблюдается немонотонность, отличная от случайного леса. На обучении в целом значение функции ошибки убывает с ростом числа признаков, график для валидации нестабилен, виден рост ошибки при числе признаков, близком к количеству признаков в исходном датасете. Заметны при этом явные оптимумы для обоих случаев: 8 для валидации с RMSE = 109534.856 и 15 для обучения с RMSE = 55452.092. В качестве оптимального количества рассмотрим значение 8.

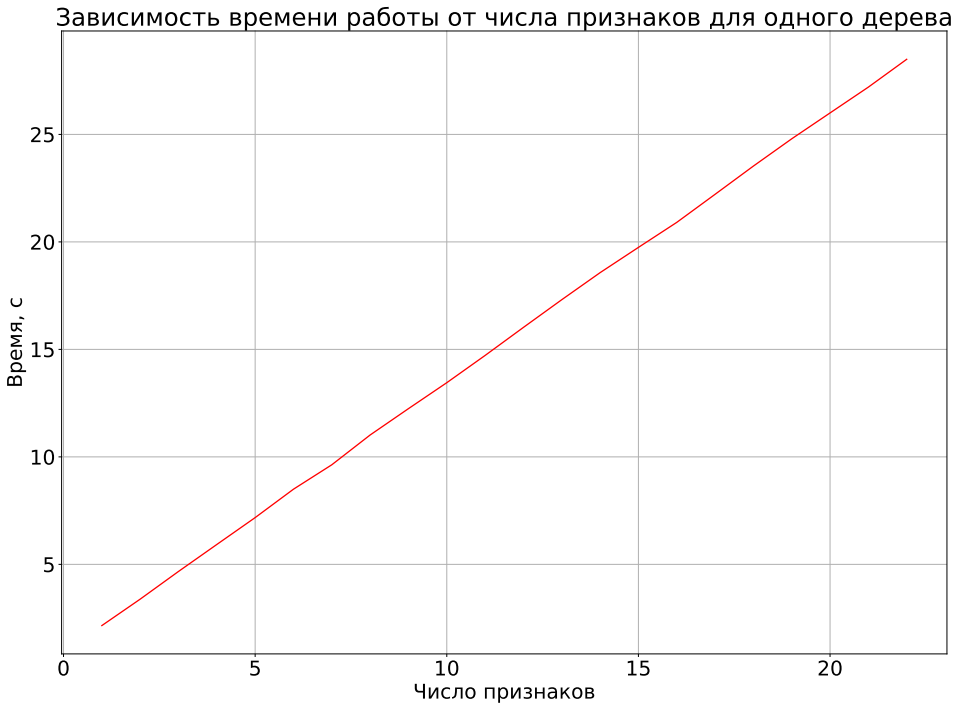

Рис. 10: Зависимость времени работы от числа признаков для одного дерева, алгоритм градиентный бустинг

Зависимость времени работы аналогично почти линейна.

Зависимость от максимальной глубины деревьев

Исследуем зависимость при следующих параметрах: число деревьев - 400, количество признаков для одного дерева - 8. Также рассмотрим случай неограниченной глубины.

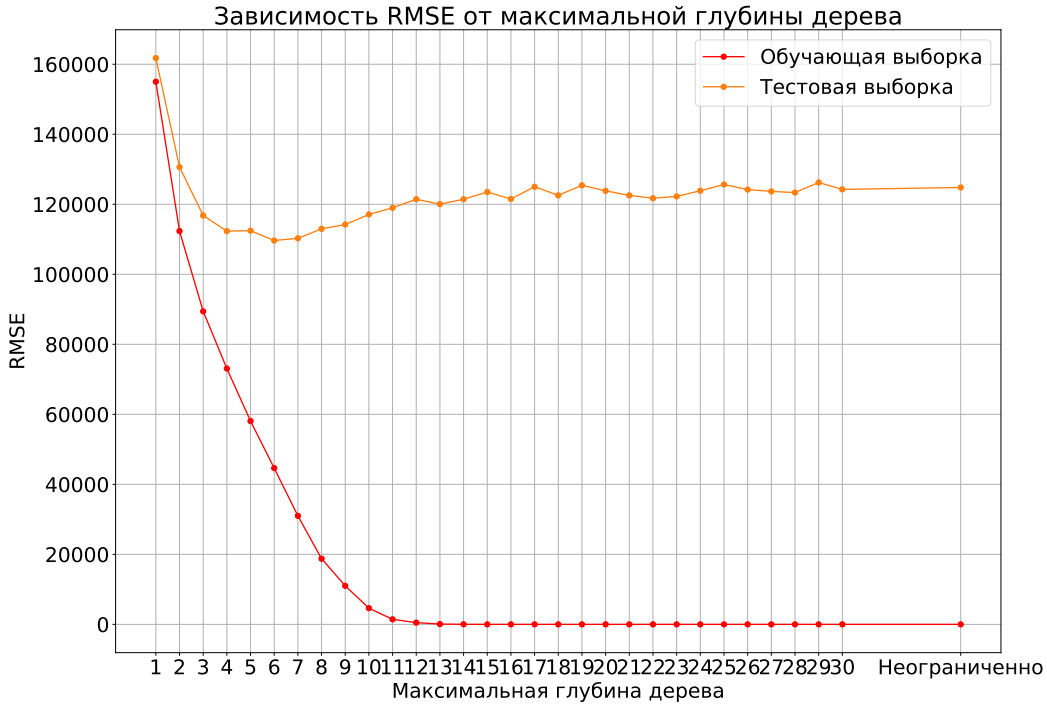

Рис. 11: Зависимость функции потерь RMSE от максимальной глубины деревьев, алгоритм градиентный бустинг

Заметим, что на обучающей выборке при больших значениях глубины модель сильно переобучается, достигается нулевое значении функции ошибки. Для валидационной выборки сначала наблюдается убывание, затем начинается немонотонное возрастание графика, что свидетельствует о переобучении в случае глубоких деревьев. Оптимальная глубина для валидационной выборки - 6 с RMSE = 109648.362, для обучающей - 20 с RMSE = 0.00011.

Рис. 12: Зависимость времени работы от максимальной глубины деревьев, алгоритм градиентный бустинг

Зависимость для времени достаточно странная, сначала время линейно возрастает, далее наблюдается резкое уменьшение и выход на асимптоту. Сложно сказать, с чем связано такое поведение, возможно со случайностью выбора признаков, но скорее всего с переобучением, так как время измерялось на обучающей выборке. Стоит отметить, что подобная картина наблюдается при неоднократных повторных запусках.

Зависимость от темпа обучения

Можно сказать, что данный параметр позволяет бороться с переобучением, определяет степень «доверия» каждому следующему базовому алгоритму. Исследуем зависимость в совокупности с числом деревьев от 1 до 1000 при максимальной глубине, равной 6 и 8 признаках для каждой вершины.

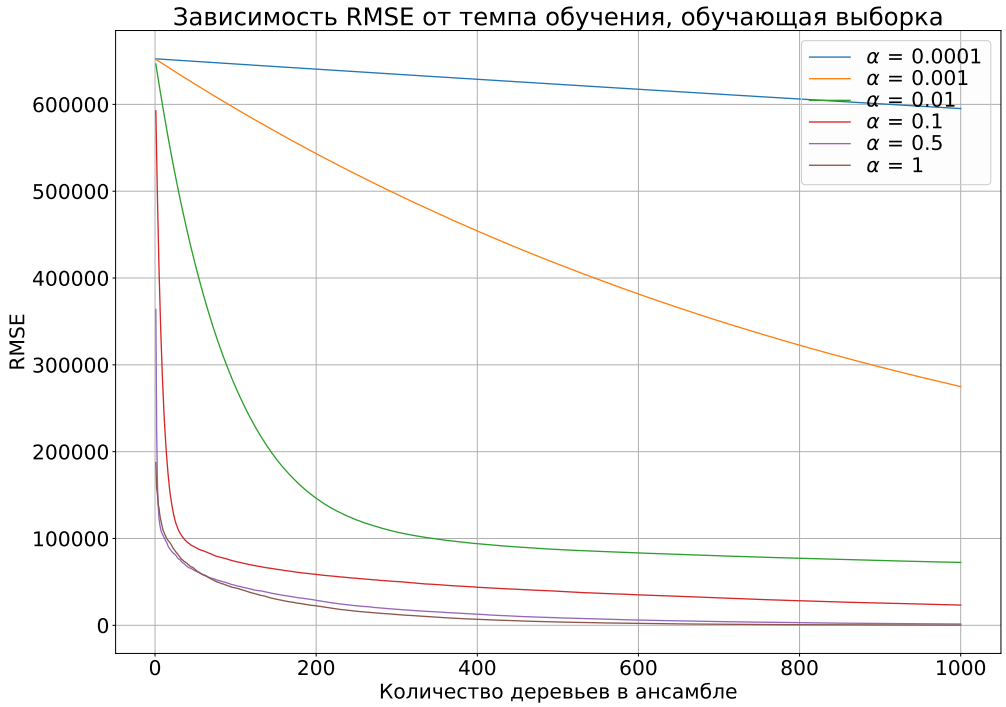

Рис. 13: Зависимость функции потерь RMSE от числа деревьев и темпа обучения, алгоритм градиентный бустинг, обучающая выборка

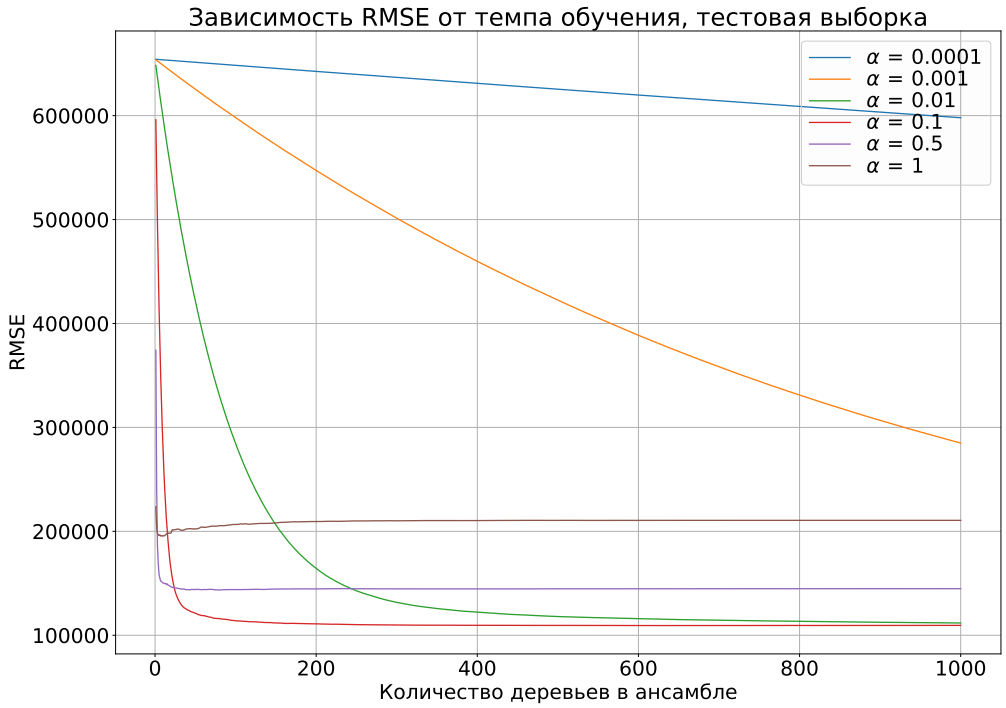

Рис. 14: Зависимость функции потерь RMSE от числа деревьев и темпа обучения, алгоритм градиентный бустинг, валидационная выборка

Заметно, что при малых значениях темпа обучения модель не успевает достичь оптимума, ей требуется большее число базовых алгоритмов. На обучающей выборке при любом темпе обучения присутствует монотонное убывание ошибки, с ростом темпа обучения значение RMSE уменьшается. Для валидационной выборки относительная монотонность есть лишь для некоторых значений параметра. При сравнительно больших значениях темпа обучения на валидации функция ошибки незначительно возрастает. При значениях 0.01 и 0.1 ошибка медленно убывает с ростом числа деревьев. Сначала с увеличением параметра до 0.1 значение RMSE уменьшается, далее ошибка растет. Вероятно это связано с тем, что модель начинает расходиться, «перелетать» точку оптимума.

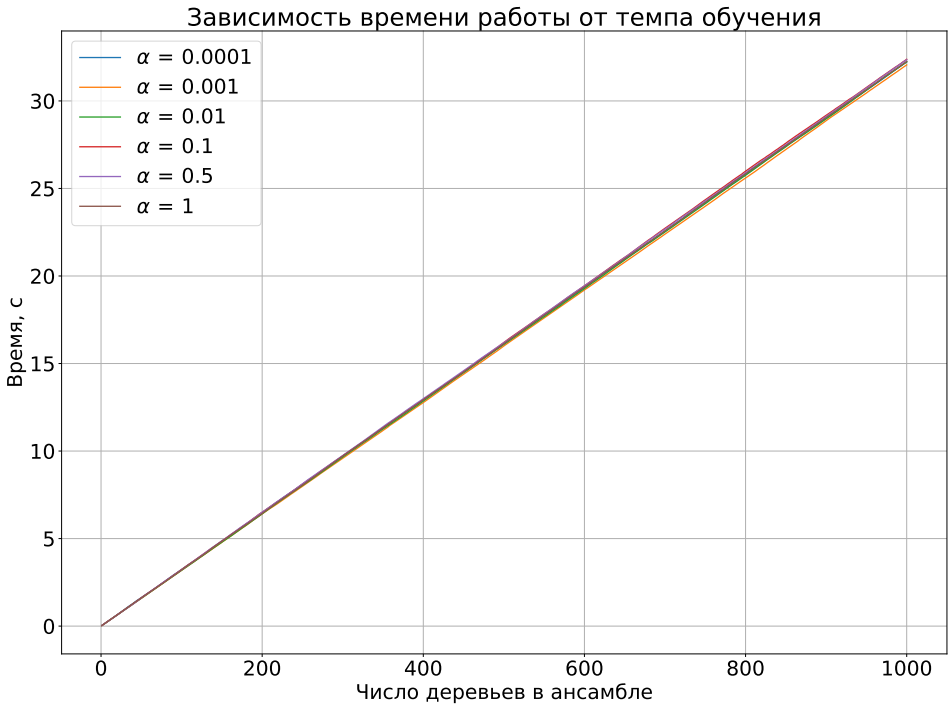

Рис. 15: Зависимость времени работы от числа деревьев и темпа обучения, алгоритм градиентный бустинг

Графики для времени практически сливаются в один, время работы почти не различается при различных значениях параметра. Таким образом, можно сказать, что время работы алгоритма не зависит от темпа обучения.

Выводы

В данной работе были рассмотрены алгоритмы **случайный лес** и **градиентный бустинг** в применении к задаче регрессии, проведено исследование зависимости их качества и времени работы от их гиперпараметров: числа базовых алгоритмов, числа признаков для одного дерева, максимальной глубины деревьев, темпа обучения (для градиентного бустинга). Использование градиентного бустинга оказалось предпочтительнее, так как данный алгоритм позволяет достичь лучших значений метрики качества RMSE, работает быстрее, чем случайный лес, а также имеет возможность более тонкой настройки ввиду особенностей алгоритма и наличия параметра темпа обучения в частности. Однако данный метод склонен к переобучению при большом числе деревьев, может очень долго сходиться при малых значениях темпа обучения и расходиться при больших значениях. Таким образом, градиентый бустинг требует более тщательного и продуманного подбора параметров. Случайный лес же является относительно простой моделью, не переобучается с ростом числа деревьев, однако является менее гибкой моделью и работает значительно дольше.

Список литературы

[1] К. В. Воронцов. Машинное обучение, курс лекций. http://www.machinelearning.ru/wiki/index.php?title=%D0%9C%D0%B0%D1%88%D0%B8%D0%BD%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D1%83%D1%87%D0%B5%D0%BD%D0%B8%D0%B5_(%D0%BA%D1%83%D1%80%D1%81_%D0%BB%D0%B5%D0%BA%D1%86%D0%B8%D0%B9,_%D0%9A.%D0%92.%D0%92%D0%BE%D1%80%D0%BE%D0%BD%D1%86%D0%BE%D0%B2)