Московский государственный университет имени М.В. Ломоносова

Факультет вычислительной математики и кибернетики

Солдатов Владислав Денисович
317 группа

Отчёт
по курсу «Технологическая практика»
Ансамбли алгоритмов. Композиции алгоритмов для решения задачи регрессии.

Москва,
2022

Содержание

Аннотация

На настоящее время ансамбли алгоритмов, в частности такие семейства моделей как случайные леса и градиентный бустинг над деревьями, считаются одними из лучших по показываемому качеству в задачах с табличными данными. В процессе выполнения работы были созданы реализации этих алгоритмов для задачи регресии с функцией потерь, вычисляемой по формуле средневадратичного отклонения, а так же проведены эксперименты по исследованию и сравнению работы получаемых в ходе работы алгоритмов моделей на реальных данных. В данной работе представлен отчёт по проделанным экспериментам.

1 Введение

Практическая польза методов ансамблирования основывается на теоретических результатах, показывающих что при использовании такого подхода, в разложении ошибки на смещение и разброс, величина значения разброса уменьшается, а смещение остаётся на том же уровне. Алгоритм случайного леса «усредняет» ответы базовых алгоритмов-деревьев, то есть выдаёт предсказание по формуле $a_T(x) := \frac{1}{T}\sum_{t=1}^{T} b_t(x)$, где b_t – базовые алгоритмы, то есть единичные решающие деревья, каждый из которых обучен на некоторой подвыборке обучающей выборки, с использованием некоторого подмножества признаков. Для градиентного бустинга используется следующая стратегия: каждый новый базовый алгоритм настраивается на антиградиент функции потерь, в случае среднеквадратической ошибки равный разности целевой переменной и ответа, получаемого моделью при использовании всех ранее построенных деревьев. Далее полученный базовый алгоритм добавляется к остальным с весом, равным значению оптимального градиентного шага, умноженным на величину темпа обучения. Предсказание алгоритма формируется по формуле $a_T(x) := \sum_{t=1}^{T} \alpha_t b_t(x)$, где α_t – соответствующий t-му базовому алгоритму вес. Полученные реализации алгоритмов на языке программирования *Python* используют функционал библиотек *numpy* [1], *sklearn* [3], *scipy* [4]. В связи с тем, что каждый базовый алгоритм в подходе случайного леса обучается независимо от остальных, было принято решение при обучении итоговой модели строить базовые алгоритмы параллельно, используя функционал для многопоточности и многопроцессности, предоставляемый библиотекой *joblib* [2]. Подобная возможность не была реализована для алгоритма градиентного бустинга из-за последовательности построения базовых алгоритмов в процессе обучения. Для проведения экспериментов по исследованию работы алгоритмов была использована выборка с данными о продаже недвижимости «**House sales in King County, USA**».

2 Предобработка данных

В табл. 1 представлена основная информация о признаках, предоставленных для построения прогноза цены дома. Видно, что идентификационный номер, уникальный для каждого проданного дома не несёт информации, которая могла бы быть использована для предсказания целевой переменной, поэтому его необходимо опустить для дальнейшего исследования. Упорядоченные категориальные и численные признаки было решено оставить без изменения, так как модели решающих деревьев не меняют поведение в зависимости от масштаба признаков. Неупорядоченные категориальные признаки, а именно наличие набережной и почтовый код зоны, в которой находится дом, было решено закодировать с помощью mean-target encoding'а со сглаживанием. Отказ от one-hot кодирования был произведён с учётом числа уникальных значений признаков: в случае почтового кода такой подход в несколько раз увеличил бы размерность данных и замедлил бы процесс обучения, а для бинарно-

	Число уникальных значений	Тип данных	Категориальный признак
Идентификационный номер	21436	Целое число	-
Дата продажи	372	Строка	Нет
Число спален	13	Целое число	Упорядоченный
Число ванных комнат	30	Вещественное число	Упорядоченный
Жилая площадь	1038	Вещественное число	Нет
Общая площадь	9782	Вещественное число	Нет
Число этажей	6	Вещественное число	Упорядоченный
Набережная	2	Целое число	**Да**
Вид	5	Целое число	Упорядоченный
Состояние дома	5	Целое число	Упорядоченный
Оценка	12	Целое число	Упорядоченный
Жилая площадь над землёй	946	Вещественное число	Нет
Площадь подвала	306	Вещественное число	Нет
Год постройки	116	Целое число	Нет
Год реновации	70	Целое число	Нет
Почтовый код	70	Целое число	**Да**
Широта	5034	Вещественное число	Нет
Долгота	752	Вещественное число	Нет
Средняя жилая площадь в округе	777	Вещественное число	Нет
Средняя площадь участка в округе	8689	Вещественное число	Нет

Таблица 1: Анализ признакового пространства задачи

го признака наличия набережной указанный алгоритм кодирования оставил бы данные в исходном состоянии, не внеся вклад в качество признакового описания.

Данные были разделены на обучающую и отложенную выборки в отношении 70:30, после чего к ним были применены вышеописанные преобразования, а затем выборки были переведены в формат массивов *numpy*.

3 Исследование работы случайного леса

В данной части работы было проведено исследование влияния на работоспособность алгоритма случайного леса различных гиперпараметров метода. Среди методов оценивания модели при фиксированном наборе гиперпараметров: время обучения и значение метрики **RMSE** на обучающей и контрольной выборках. Параметры, перебираемые в процессе исследования, включают: число используемых базовых алгоритмов, доля сэмплируемых для бутстрэпа объектов, доля используемых одиночным алгоритмом признаков, максимальная глубина дерева. В связи с параллельным построением деревьев, время считается отдельно для каждого обучающегося базового алгоритма, в его собственном потоке. Кроме того, из-за высокой скорости обучения и меньшего числа параметров алгоритма, по сравнению с градиентным бустингом в получившихся реализациях, выбор гиперпараметров осуществлялся полным перебором, в отличие от случая сравниваемого алгоритма, для которого, как будет видно далее, этот процесс осуществлялся жадным образом. Полученные таким образом опти-

		Время, с	Число деревьев	RMSE на обучающей выборке	RMSE на контрольной выборке
Доля признаков	$\frac{1}{3}$	76.16	27	55 095	156 267
	0.1	17.85	3	257 847	292 040
	0.75	162.31	996	44 287	**131 633**
	1.0	207.95	100	44 617	132 711
Доля объектов	auto	105.02	89	65 193	131 842
	0.3	55.06	83	95 479	138 551
	0.75	122.01	59	57 561	133 334
	1.0	162.31	996	44 287	**131 633**
Максимальная глубина	1	26.57	16	280 259	298 699
	3	43.53	28	189 063	210 624
	5	58.82	413	139 898	164 634
	Не ограничена	162.31	996	44 287	**131 633**

Таблица 2: Перебор гиперпараметров для случайного леса

мальные с точки зрения значения выбранной метрики на отложенной выборке гиперпараметры были проанализированы следующим образом: для каждой компоненты были просмотрены все возможные значения, при фиксированных значениях остальных параметров. Итоговые оптимальные значения гиперпараметров отображены в табл. 2, где значение *auto* для параметра доли объектов означает, что алгоритм выбирает число объектов, попадающих в обучающую выборку для базового алгоритма по формуле $1 - (1 - \frac{1}{l})^l$, где l – объём полной выборки. Число деревьев отражает то значение этой величины, для которой достигается наилучшее качество на отложенной выборке.

3.1 Размерность признакового пространства

На графике 1 отображена зависимость метрики RMSE от числа деревьев, использованных для предсказания, и от времени, затраченного на обучение. Видно, что с увеличением рассматриваемого параметра, улучшается качество, показываемое моделью на обучающей выборке, и лишь разница между значениями 0.75 и 1.0 заметна слабо. При этом, так же увеличивается и разрыв в качестве на обучающей и валидационной выборках. Кроме того, по графику зависимости метрики от времени обучения, можно заметить, что с увеличением размерности признакового пространства, увеличивается и затрачиваемое на обучение время. Это подтверждается графиком зависимости времени обучения от числа деревьев для различных значений доли используемых признаков на рис. 2.

Согласно представленным выше значениям (табл. 1), полученным в результате перебора, при значении 1.0 рассматриваемого гиперпараметра, лучшее значение метрики на отложенной выборке достигается при числе деревьев равном 100, после чего с увеличением количества базовых алгоритмов до 1000, качество на валидации только ухудшается, что может свидетельствовать о сильном переобучении. В противоположность этому, для значения 0.75, качество на контрольной выборке улучшалось вплоть до 996-го добавленного дерева. Кроме того, модель, обученная с таким значением гиперпараметра доли используемых признаков, показывает лучшее время обучения, чем при значении равном 1.0. Благодаря всему вышесказанному, можно сделать вывод, что выбор данного значения рассматриваемого гиперпараметра был сделан верно.

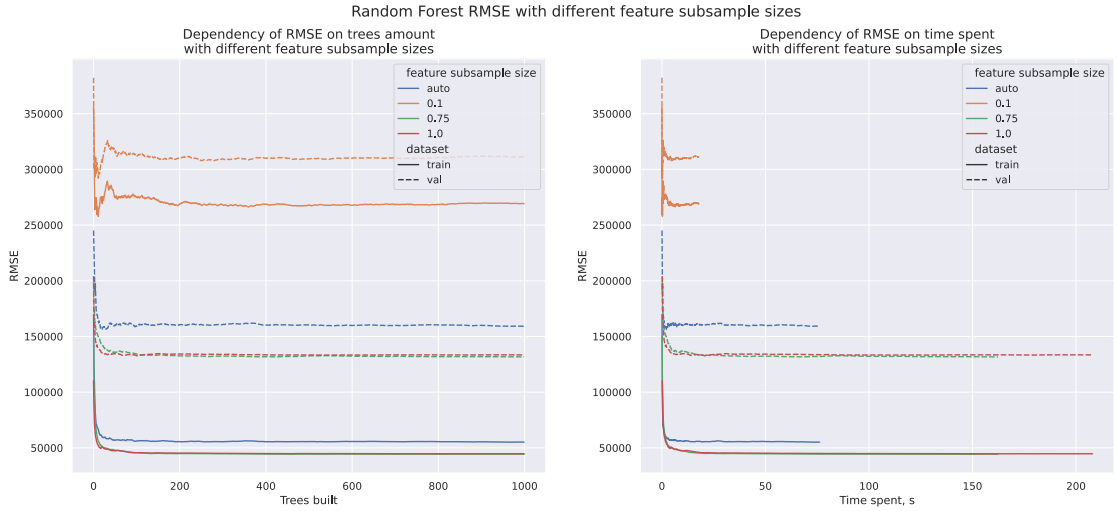


Рис. 1: Зависимость RMSE от числа деревьев и времени работы случайного леса для различных значений доли используемых признаков

3.2 Доля объектов, используемых при обучении

На рис. 3 отображены аналогичные предыдущему пункту зависимости, полученные при изменении размера подвыборки, сэмплируемой с возвращением из обучающей выборки. Видна похожая зависимость качества на обучающей выборке от объёма данных, использованных для построения базового алгоритма, а так же увеличение разрыва в качестве на обучении и валидации, отмеченные и для гиперпараметра рассмотренного выше. Несмотря на это, с увеличением значения числа объектов, использованных в обучении дерева, улучшается и качество на отложенной выборке, а так же заметна похожая на предыдущий пункт ситуация, в которой худшие по качеству алгоритмы достигают пика гораздо раньше полного обучения 1000 деревьев. Ситуация с временем обучения (рис. 4) абсолютно аналогична рассмотренной выше.

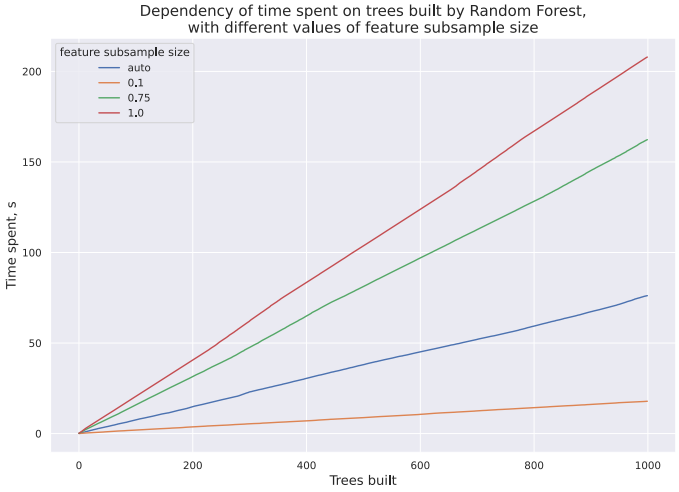


Рис. 2: Зависимость времени работы случайного леса от числа деревьев для различных значений доли используемых признаков

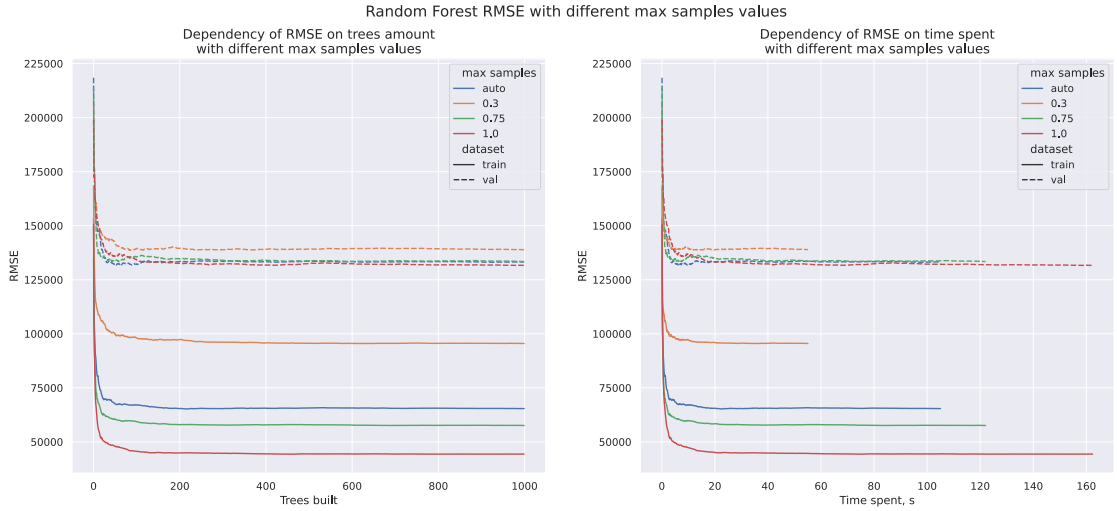


Рис. 3: Зависимость RMSE от числа деревьев и времени работы случайного леса для различных значений доли объектов, использованных при обучении

3.3 Максимальная глубина дерева

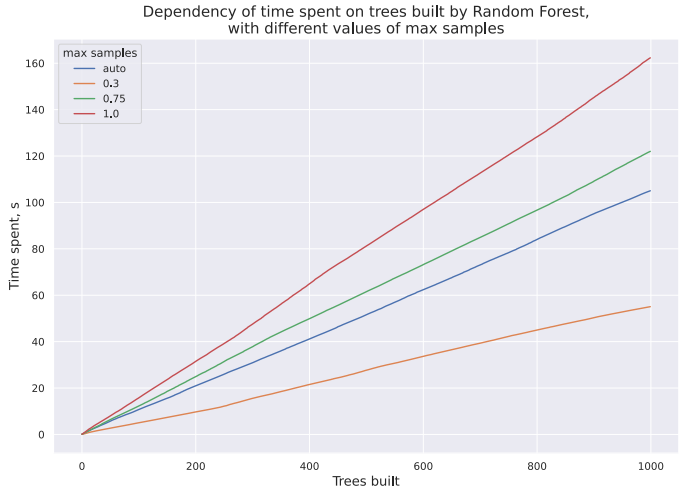


Рис. 4: Зависимость времени работы случайного леса от числа деревьев для различных значений доли объектов, использованных при обучении

Графики, отражающие процесс выбора оптимальной высоты дерева, на рис. 5 и рис. 6, в целом отображают подобную предыдущим пунктам ситуацию: с увеличением сложности модели улучшается качество на обучающей выборке и увеличивается зазор в показателях метрики на обучающей и отложенной выборке. При этом заметен новый тренд: при каждом большем значении рассматриваемого гиперпараметра качество на валидации превышает даже качество на обучающей выборке, показанное моделью, обученной с меньшим значением максимальной глубины деревьев.

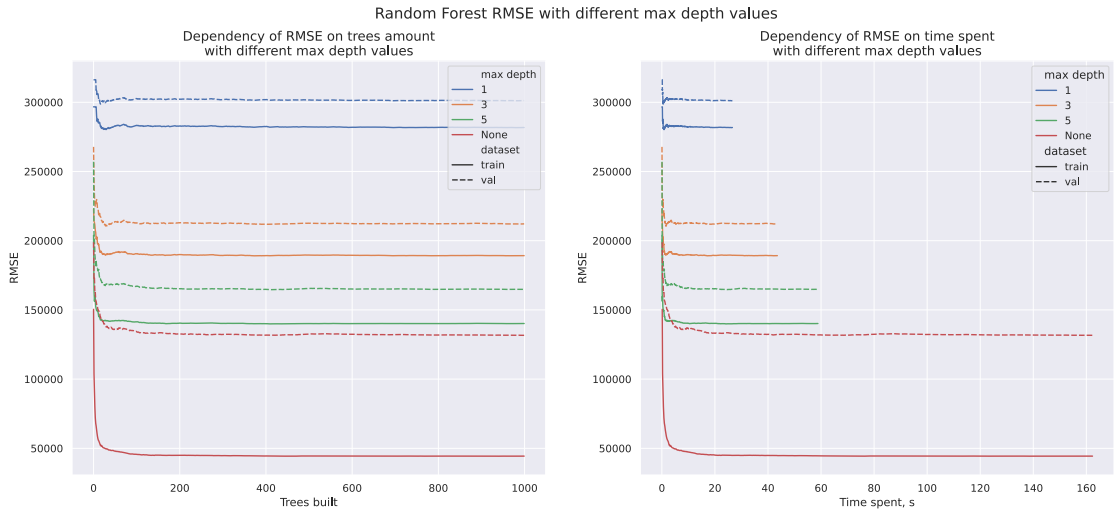


Рис. 5: Зависимость RMSE от числа деревьев и времени работы случайного леса для различных значений максимальной глубины дерева

4 Исследование работы градиентного бустинга

С алгоритмом градиентного бустинга был проведён аналогичный предыдущей секции перебор гиперпараметров, с вышеупомянутым уточнением о жадной реализации алгоритма перебора, а так же учитывая добавление к списку исследуемых параметров величины темпа обучения. В табл. 3 представлены результаты этого перебора.

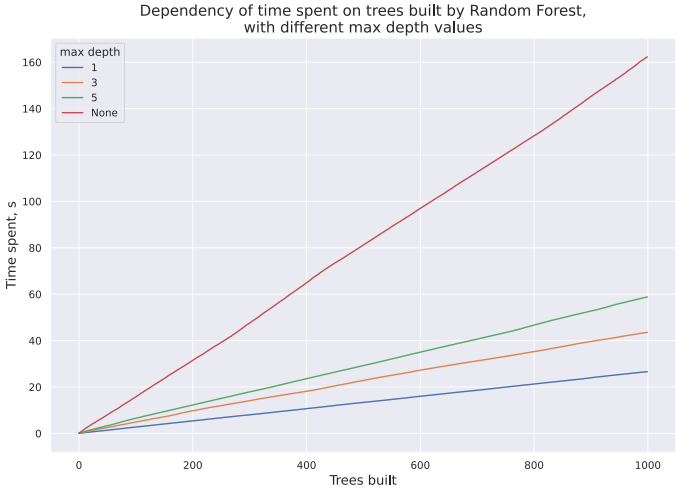


Рис. 6: Зависимость времени работы случайного леса от числа деревьев для различных значений максимальной глубины дерева

Видно отличие от случайного леса в отношении параметра количества обучаемых базовых алгоритмов: раннее прекращение роста качества на контрольной выборке происходит лишь для более сложных моделей, максимально подстраивающихся под предоставляемую зависимость: модель с неограниченной высотой деревьев и модель с шагом алгоритма равным единице. Это различие в значениях с предыдущим рассмотренным методом может быть объяснено тем, что градиентный бустинг это более

		Время, с	Число деревьев	RMSE на обучающей выборке	RMSE на контрольной выборке
Максимальная глубина	1	3.46	999	133 222.18	165 316
	3	7.26	984	84 824.73	128 257
	5	11.12	983	59 583.64	**118 333**
	Не ограничена	107.74	105	0.95	134 854
Размер шага обучения	0.001	11.19	1000	198 328.50	217 776
	0.01	11.47	1000	91 297.07	128 135
	0.1	11.12	983	59 583.64	**118 333**
	1.0	11.61	24	25 226.18	183 867
Доля признаков	**1/3**	11.12	983	59 583.64	**118 333**
	0.1	3.26	997	72 123.74	127 030
	0.75	24.55	994	56 920.99	125 755
	1.0	32.71	956	55 757.48	129 644
Доля объектов	**auto**	11.12	983	59 583.64	**118 333**
	0.3	6.08	997	73 940.49	129 785
	0.75	13.39	995	56 413.79	120 452
	1.0	17.11	990	53 056.34	126 235

Таблица 3: Перебор гиперпараметров градиентного бустинга

сложный алгоритм, и дальнейшее усложнение приводит к быстрому переобучению. При этом благодаря этому для более простых базовых алгоритмов бустинг показывает лучшие по качеству результаты по сравнению со случайным лесом.

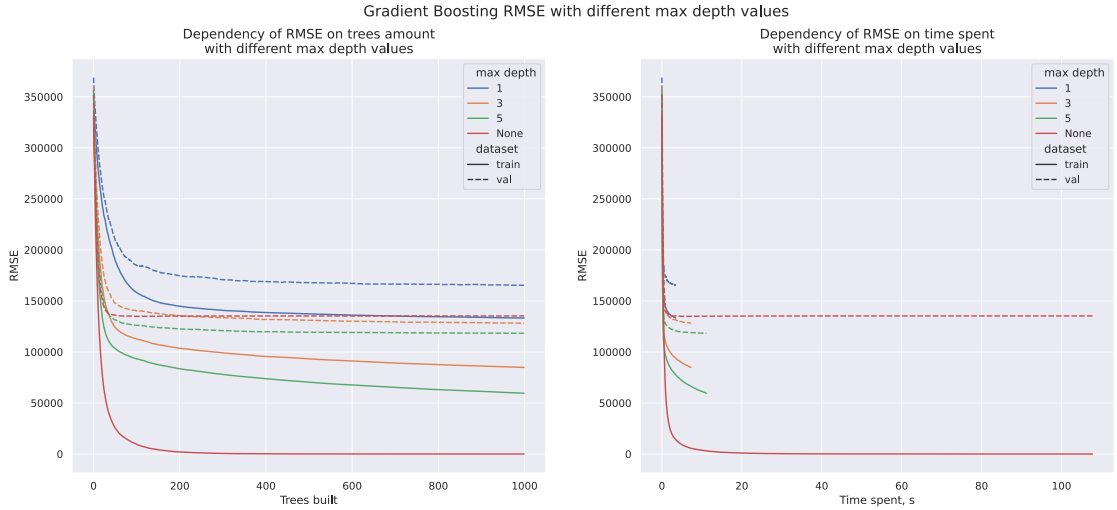


Рис. 7: Зависимость RMSE от числа деревьев и времени работы градиентного бустинга для различных значений максимальной глубины деревьев

4.1 Максимальная глубина дерева

На рис. 7 отображены зависимости метрики RMSE от числа деревьев и от времени обучения. За исключением вышеописанного и разобранного случая использования базовых алгоритмов неограниченной высоты, ситуация похожа на аналогичную для алгоритма случайного леса: с увеличением сложности алгоритма улучшается качество и на обучающей и на отложенной выборках. При этом эти улучшения не такие «резкие», как в случае сравниваемого алгоритма, и хотя рост качества с увеличением числа деревьев замедляется, он тем не менее происходит, в отличие от ситуации со случайным лесом. На рис. 8 видна предсказуемая и аналогичная уже рассмотренным выше случаям зависимость времени, потраченного на обучение, от числа базовых алгоритмов, использованных моделью.

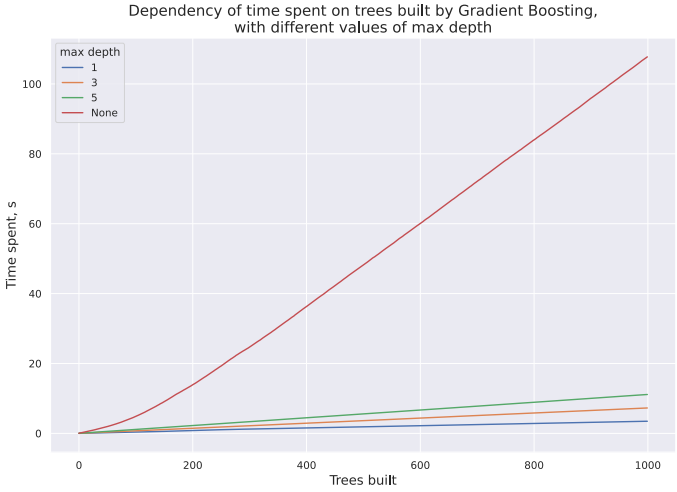


Рис. 8: Зависимость времени работы градиентного бустинга от числа деревьев для различных значений максимальной глубины деревьев

4.2 Величина шага алгоритма

На графиках на рис. 9 показаны характеристики работы исследуемого алгоритма при изменении параметра величины шага алгоритма. Видно, что в отличие от других рассмотренных гиперпараметров, данный не влияет на время обучения при фиксированном количестве обучаемых базовых алгоритмов (в пределах допустимой погрешности). Это объяснимо тем, что данный гиперпараметр меняет поведение лишь самой модели: деревья строятся одинаково для каждого значения, меняется лишь множитель перед весом, с которым они добавляются в ансамбль. Нетрудно предположить, что чем больше значение этого гиперпараметра, тем меньше требуется деревьев для достижения некоторой величины метрики на обучающей выборке. При этом происходит большее «подстраивание» под зависимости, представленные именно в обучении, то есть усиляется эффект переобучения. Это подтверждается графиком: при величине шага алгоритма равной единице, то есть в ситуации, когда каждое последующее дерево добавляется в модель с оптимальным с точки зрения потерь на обучающей выборке весом, модель для каждого значения числа использованных деревьев показывает лучшие значения метрики на обучающей выборке, при этом на отложенной выборке показывает одну из худших величин качества. В противоположность этому, при наименьшем по величине из значений рассматриваемого гиперпараметра, модель не «доучилась» даже при 1000 деревьев, при этом показывая стабильный по числу базовых алгоритмов разрыв между значениями метрики на обучении и на валидации.

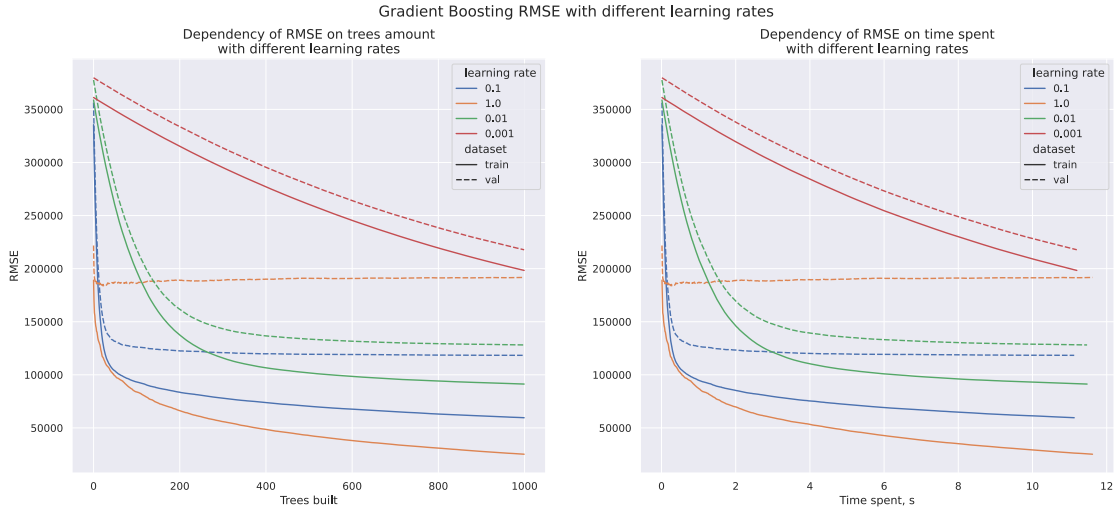


Рис. 9: Зависимость RMSE от числа деревьев и времени работы градиентного бустинга для различных значений величины шага обучения

4.3 Параметры объёма обучающей выборки для базовых алгоритмов

В этой подсекции рассмотрено влияние изменения гиперпараметров доли объектов и доли признаков, использованных для обучения отдельно взятых деревьев. На рис. 10 представлены исследуемые зависимости при изменении параметра размерности признакового пространства.

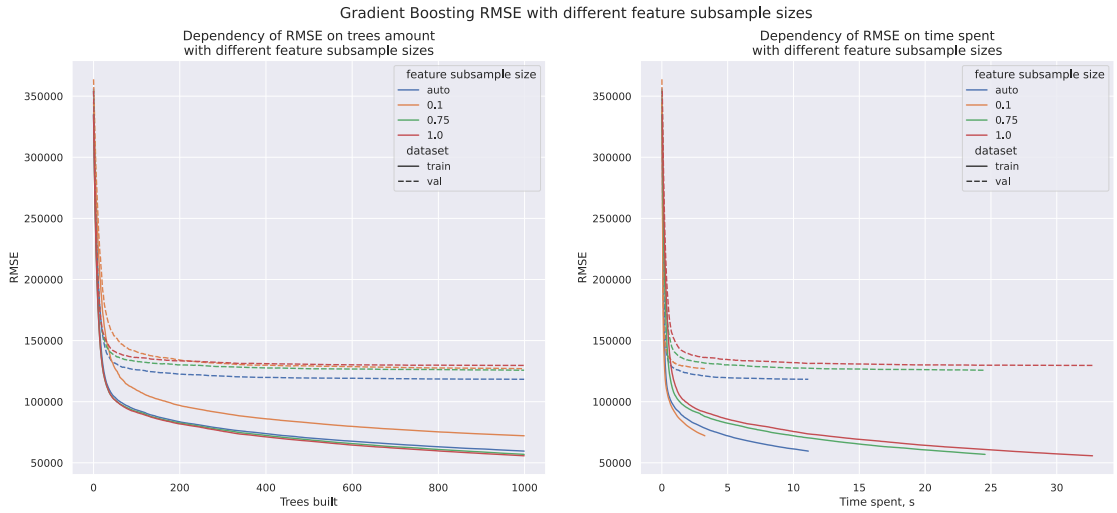


Рис. 10: Зависимость RMSE от числа деревьев и времени работы градиентного бустинга для различных значений доли используемых признаков

Видно, что при значении этого параметра, выбранном по эмпирическому правилу, то есть как треть размерности общего признакового пространства, полученная модель показывает качество на обучающей выборке чуть хуже более «знающих» моделей, при этом демонстрируя лучшее значение метрики на контрольном наборе данных. Это свидетельствует о меньшем темпе переобучения по сравнению с другими вариантами обучения алгоритма. Кроме того, по понятным и рассмотренным выше в анало-

гичной ситуации для сравниваемого алгоритма причинам, такой способ занимает меньше времени на обучение по сравнению с моделями, обученными с большими значениями указанного гиперпараметра и показывающими сравнимые по качеству результаты. При этом модель, деревья которой обучаются лишь на десятой части признаков, демонстрирует показатели качества на валидации лучше, чем модель, базовые алгоритмы которой построены на всём признаковом пространстве. Это демонстрирует важность рассматриваемого гиперпараметра в задаче уменьшения эффекта переобучения модели.

Зависимости между различными по величине значениями гиперпараметра числа сэмплируемых для обучения дерева объектов похожи на рассмотренные соотношения между значениями размерности признакового пространства, и выводы из графиков аналогичны. Графики упомянутых зависимостей размещены в аппендиксе А, так же как и графики аналогичных рассмотренным выше зависимостей времени обучения от числа деревьев для указанных в настоящем пункте гиперпараметров.

5 Сравнение работы алгоритмов для представленной задачи

После выбора оптимальных наборов гиперпараметров для обоих методов, было проведено сравнение их поведения при использовании зафиксированных ранее значений настроек алгоритмов. Результаты этого сравнения отображены на графике на рис. 11. В данном случае деревья, составляющие основу алгоритма случайного леса, обучались последовательно, так как описанный выше способ измерения времени для параллельной реализации позволяет отобразить относительные зависимости по времени между разными моделями случайного леса, но не абсолютные значения времени работы. Можно отметить, что алгоритм случайного леса «выучивает» обучающую выборку при малом значении числа деревьев, и затем качество на отложенной выборке только ухудшается. В противоположность этому, градиентный бустинг на протяжении всего процесса добавления деревьев постепенно улучшает качество как на обучающей выборке, так и на валидационной, хотя и со снижающейся скоростью.

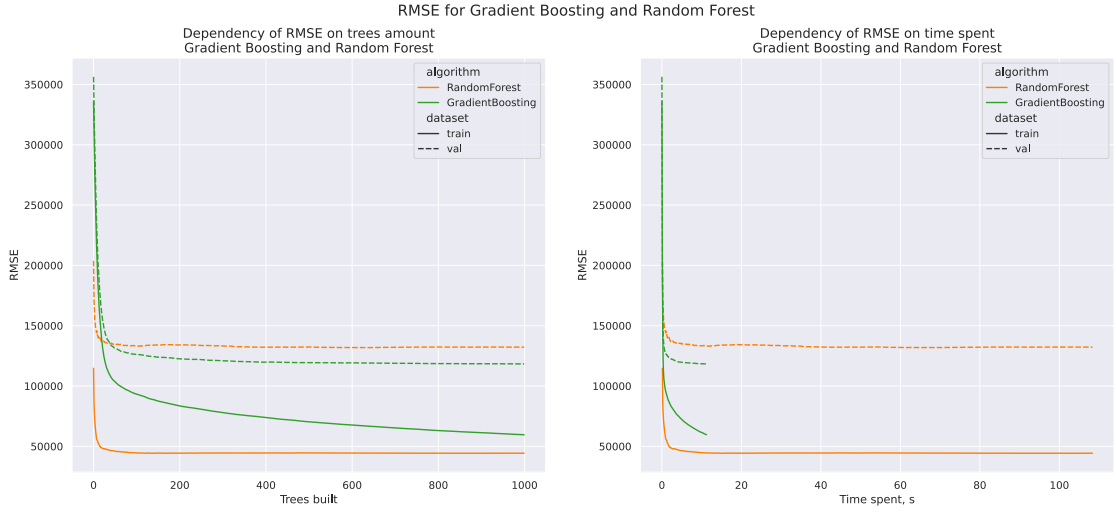


Рис. 11: Зависимость RMSE от числа деревьев и времени работы градиентного бустинга и случайного леса

Кроме того, заметно, что при последовательном подходе к построению деревьев, время, затрачиваемое на обучение алгоритма случайного леса более чем в 6 раз превышает значение той же величины для градиентного бустинга, что подтверждается графиком на рис. 12. Это объяснимо значениями

подобранных оптимальных гиперпараметров: для случайного леса максимальная глебина не ограничена, а параметры объёма сэмплируемых для дерева данных не меньше по величине соответствующих гиперпараметров для градиентного бустинга, что, как было замечено выше, кардинальным образом повышает время обучения алгоритма. При этом, несмотря на указанные препятствия, параллельно обучаемая модель случайного леса тратит на этот процесс ненамного больше времени, чем градиентный бустинг: на используемой для проведения экспериментов машине значения времени обучения составили 15.67 секунды для случайного леса и 15.14 секунды для градиентного бустинга.

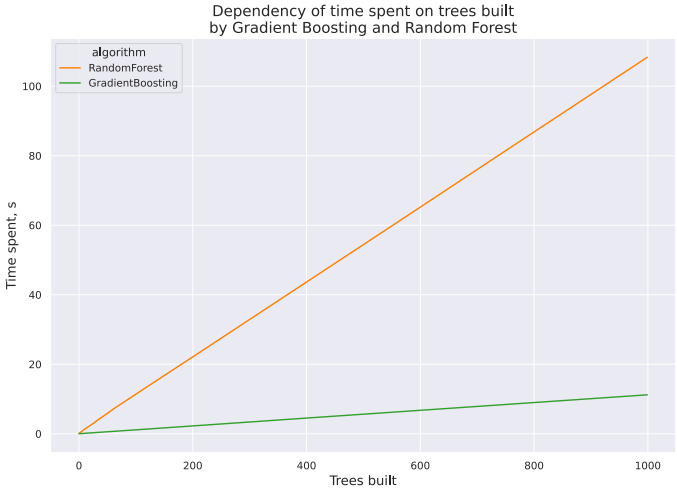


Рис. 12: Зависимость времени работы градиентного бустинга и случайного леса от числа деревьев

6 Заключение

В результате исследования было замечено, что ввиду большой сложности алгоритмов модели, построенные по ним, чувствительны к выбору гиперпараметров, и неточный их подбор может привести к совершенно неудовлетворительным результатам на проверке, при почти идеальном значении метрики качества на обучающей выборке. При этом несмотря на это, благодаря ансамблевой структуре алгоритмов, качество моделей на отложенных данных не начинает несоразмерно ухудшаться по мере усложнения алгоритма путём добавления новых деревьев в каждом из рассмотренных случаев, что было бы невозможно в ситуации одиночного алгоритма. Также были отмечены различия в поведении алгоритмов случайного леса и градиентного спуска, а именно, что при усложнении базового алгоритма дерева случайный лес всё равно может улучшать показатели на отложенной выборке, хотя и ненамного, а градиентный бустинг при самых сложных параметрах способен добиться идеального качества на обучении, продемонстрировав худшие значения метрики на валидации. Были замечены и различия в изменении поведения при вариации некоторых значений параметров, таких как доля объектов и доля признаков, используемых при обучении отдельных деревьев в составе алгоритма.

Список литературы

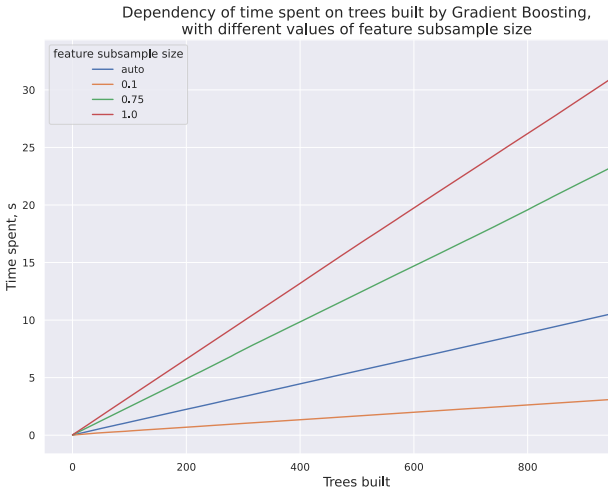
[1] Array programming with NumPy / Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt et al. // *Nature*. — 2020. — . — Vol. 585, no. 7825. — Pp. 357–362. https://doi.org/10.1038/s41586-020-2649-2.

[2] *Joblib Development Team*. Joblib: running python functions as pipeline jobs. — 2020. https://joblib.readthedocs.io/.

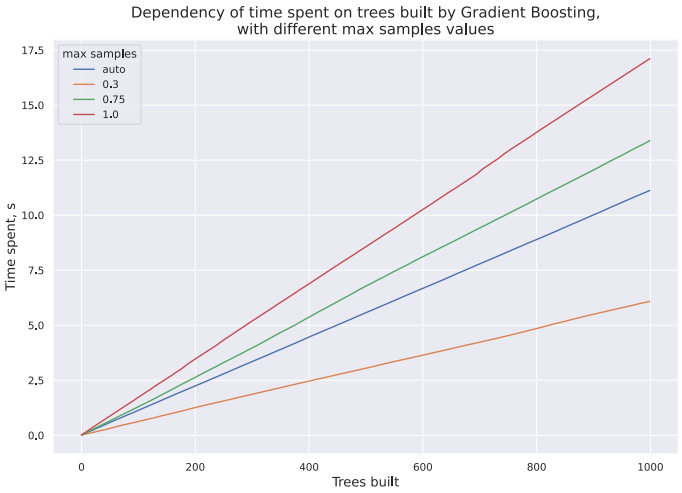
[3] Scikit-learn: Machine learning in Python / F. Pedregosa, G. Varoquaux, A. Gramfort et al. // *Journal of Machine Learning Research*. — 2011. — Vol. 12. — Pp. 2825–2830.

[4] SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python / Pauli Virtanen, Ralf Gommers, Travis E. Oliphant et al. // *Nature Methods*. — 2020. — Vol. 17. — Pp. 261–272.

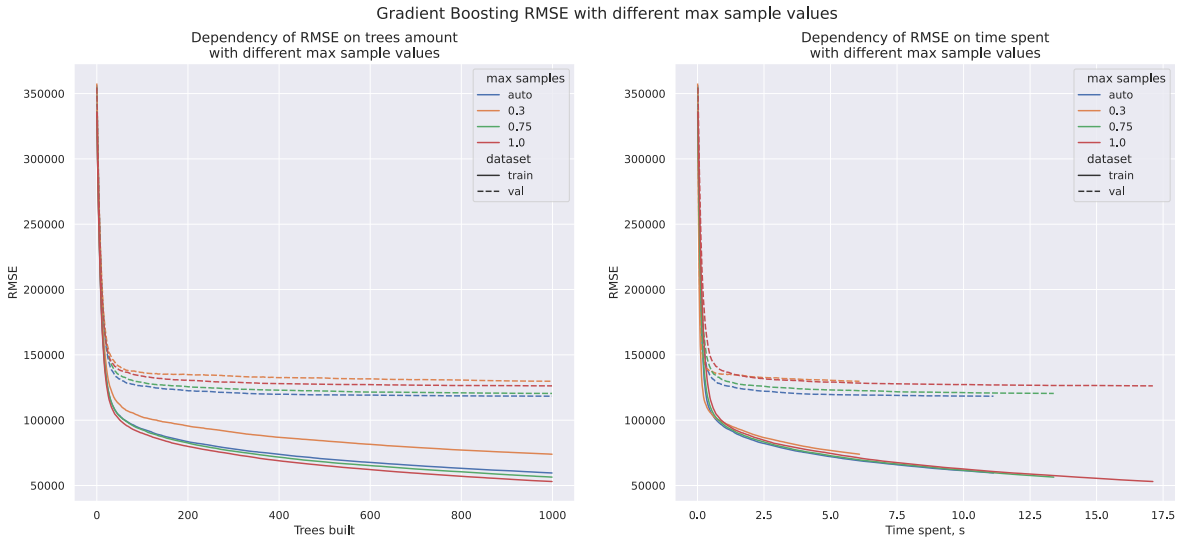
A Гиперпараметры объёма данных обучения для градиентного бустинга



(a) Зависимость времени работы градиентного бустинга от числа деревьев для различных значений доли используемых признаков



(b) Зависимость времени работы градиентного бустинга от числа деревьев для различных значений доли используемых объектов



(c) Зависимость RMSE от числа деревьев и времени работы градиентного бустинга для различных значений доли используемых объектов

Рис. 13: Зависимости RMSE от числа деревьев и времени обучения, и времени обучения от числа деревьев для градиентного бустинга при различных наборах гиперпараметров, определяющих объём обучающей выборки.