Бычков Георгий 317

Содержание

1 Введение

В данном отчете изучаются алгоритмы случайный лес и градиентный бустинг. В качестве источника данных, на которых проводится исследование, было выбрано соревнование House Sales in King County, USA по адресу.

2 Обработка данных

Для удобства работы с представленными данными нужно перевести их в числовой формат. Анализ столбцов исходной матрицы данных показал, что почти все столбцы имеют целый или вещественный тип, за исключением столбца с датой. Было решено столбец с датой превратить в 3 столбца с годом, месяцем и номером дня в месяце. Это расширит признаковое пространство и можно будет создавать большее число комбинаций признаков в деревьях . В итоге у нас получается матрица из вещественных чисел, которую можно перевести в массив numpy.

3 Исследование случайного леса

В данной части будет рассмотрен алгоритм случайного леса на приведенном наборе данных. Будет исследованы его результаты при разных значениях входных гиперпараметров.

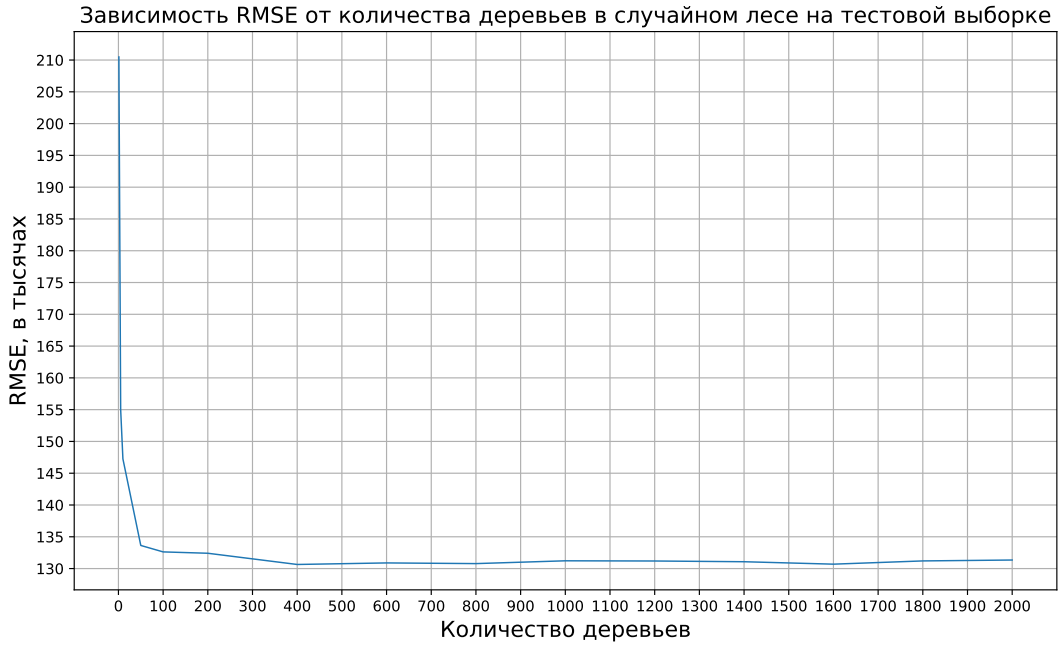

Рис. 1: Зависимость функции потерь (RMSE) от количества деревьев

3.1 Подбор оптимального количества деревьев

Первым параметром подбирается количество деревьев в алгоритме. Остальные параметры задаются по умолчанию. Как видно из графика, с ростом количества деревьев, точность алгоритма сначала увеличивается, но начиная с некоторого значения, увеличение количества деревьев не приводит к изменению функциии ошибок.

Что касается времени выполнения программы, то оно зависит линейно от количества деревьев, как можно видеть на графике. Оптимальным значением можно взять 600 деревьев, так как при этом времени программа будет работать быстрее всего, не теряя точности.

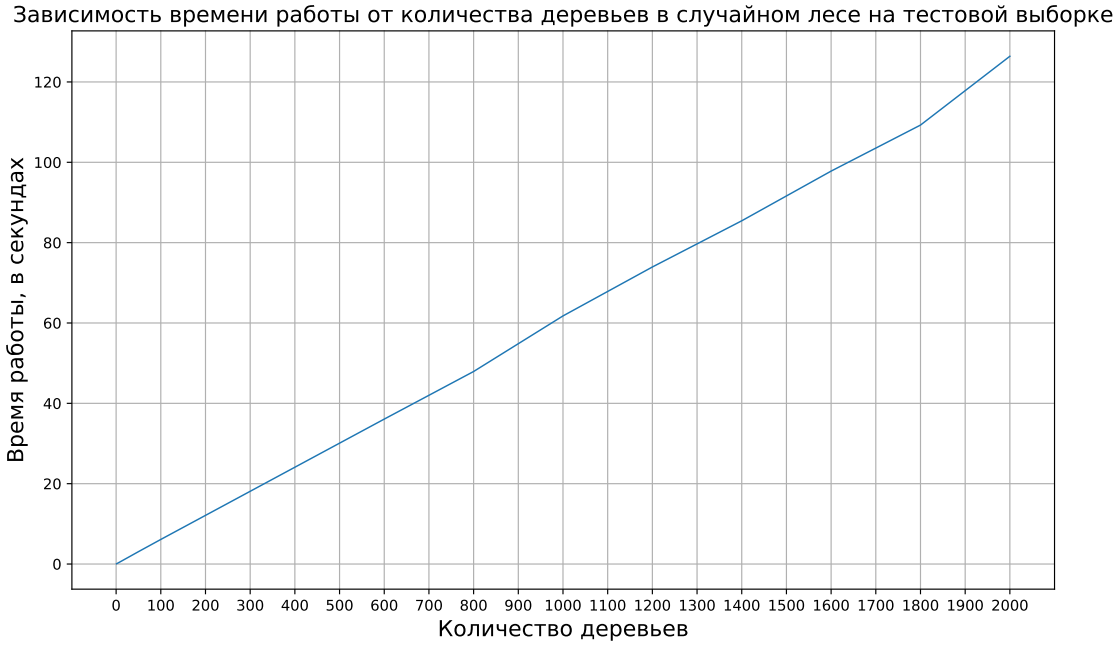

Рис. 2: Зависимость времени выполнения программы от количества деревьев

Рис. 3: Зависимость функции потерь (RMSE) от количества признаков

3.2 Подбор оптимального количества признаков

Затем подбирается оптимальное количество признаков. Количество деревьев равно 600, остальные параметры берутся по умолчанию. Как видно из графика, с ростом количества признаков значение RMSE убывает и начиная с некоторого значения точность остается почти неизменной.

С ростом числа признаков, время работы алгоритма растет линейно. Поэтому можно считать оптимальным количество признаков равным 17, так как это наименьшее значение, при котором качество остается на почти минимальном уровне, то есть алгоритм будет работать наиболее быстро с той же точностью.

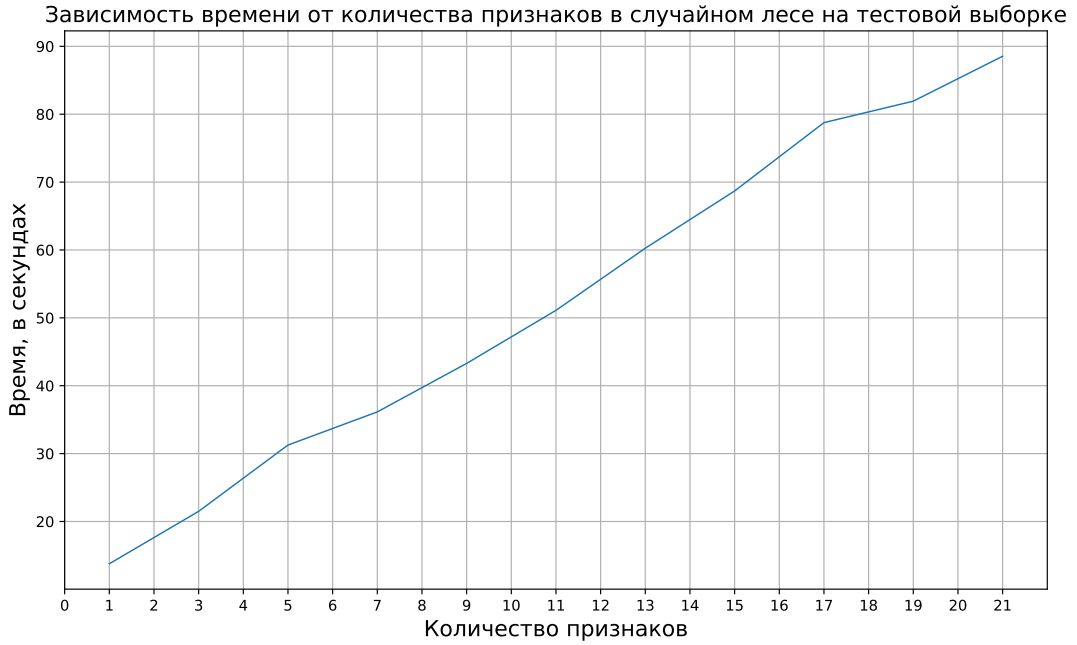

Рис. 4: Зависимость времени выполнения программы от количества признаков

Рис. 5: Зависимость функции потерь (RMSE) от максимальной глубины деревьев

3.3 Подбор оптимальной глубины деревьев

Следующим подбирается оптимальное значение глубины деревьев. Были рассмотрены значения максимальной глубины до 50, а также отдельно случай с неограниченным деревом. Как видно из графика, более глубокие (переобученные) деревья будут давать лучшую точность на тестовой выборке.

Время обучения в этот раз зависит линейно от параметра только при глубине меньшей 15, затем она начинает выходить на практически постоянное значение. Таким образом, лучшую точность будут давать глубокие деревья.

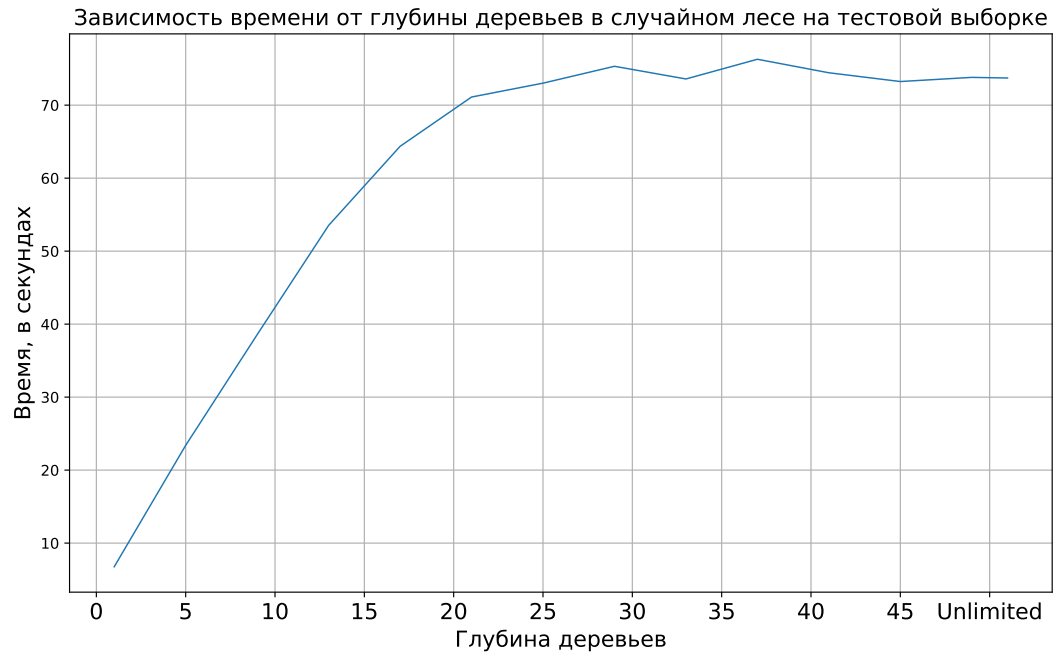

Рис. 6: Зависимость времени выполнения программы от максимальной глубины деревьев

4 Исследование градиентного бустинга

В данной части будет исследован градиентный бустинг над деревьями и его значения в зависимости от следующих гиперпараметров: количество деревьев, максимальное количество признаков, максимальная глубина, скорость обучения.

Рис. 7: Зависимость функции потерь (RMSE) от количества деревьев

4.1 Подбор оптимального количества деревьев

Первым параметром подбирается количество деревьев в алгоритме. Остальные параметры задаются по умолчанию, темп обучения по умолчанию принят равным 0.1. Как видно из графика, с ростом количества деревьев растет его точность и, как и в случае с случайным лесом, начиная с некоторого количества деревьев, точность не будет сильно меняться.

От количества деревьев время работы программы зависит так же линейно. Можно принять оптимальным значение в 1000 деревьев по таким же соображениям: наименее затратное по времени, с высокой точностью.

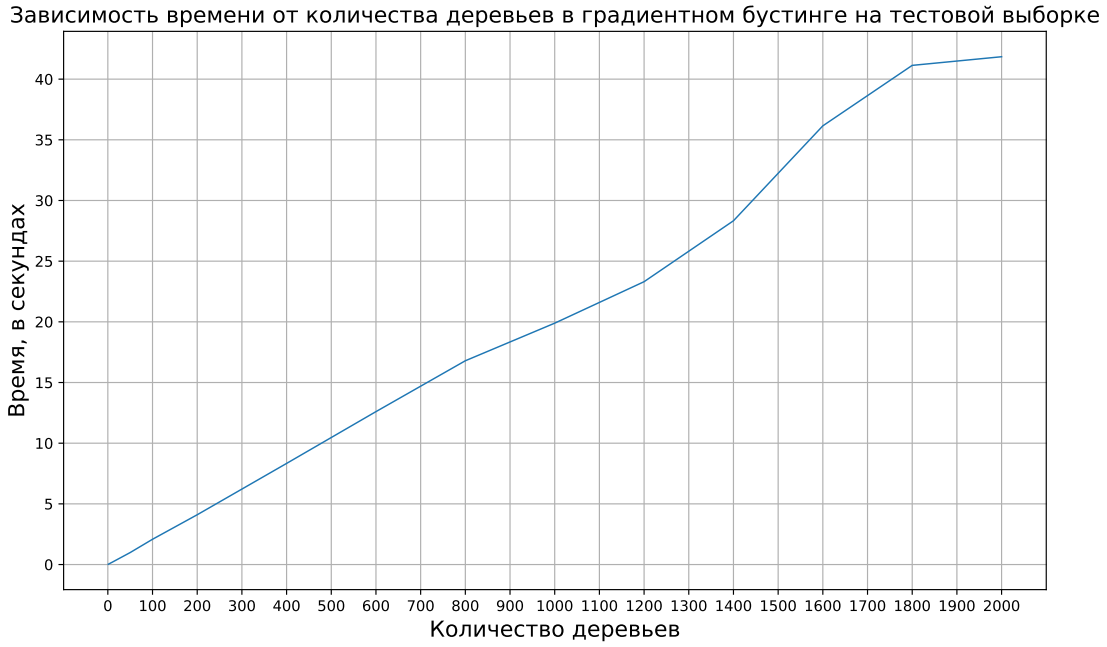

Рис. 8: Зависимость времени выполнения программы от количества деревьев

Рис. 9: Зависимость функции потерь (RMSE) от количества признаков

4.2 Подбор оптимального количества признаков

Затем подбирается оптимальное количество признаков. Количество деревьев берется из предыдущего пункта, остальные параметры по умолчанию. Как видно из графика, оптимальным является среднее количество признаков, что отличает бустинг от случайного леса. При большом или слишком маленьком количестве признаков значение RMSE больше, чем у средних. 11 признаков можно принять оптимальным.

По результатам экспериментов можно увидеть, что зависимость времени выполнения от количества признаков линейна.

Рис. 10: Зависимость времени выполнения программы от количества признаков

Рис. 11: Зависимость функции потерь (RMSE) от максимальной глубины деревьев

4.3 Подбор оптимальной глубины деревьев

Подбирается оптимальное значение глубины деревьев. Количество деревьев и количество признаков берутся из предыдущих экспериментов, остальное по умолчанию. Тут алгоритм вновь отличается от случайного леса: вместо ансамблей с глубокими деревьями оптимальными стали ансамбли с малой максимальной глубиной. То есть переобученные деревья будут давать худший результат.

Время работы алгоритма линейно при небольших значениях глубины, а затем становится почти постоянным. Оптимальным берется значение глубины равное 5.

Рис. 12: Зависимость времени выполнения программы от максимальной глубины деревьев

Рис. 13: Зависимость функции потерь (RMSE) от скорости обучения

4.4 Подбор оптимального значения скорости обучения

В данном эксперименте подбирается оптимальное значение скорости обучения. Как видно, при малых значениях параметра модель недообучается, а при значениях близких к 1 переобучается. Поэтому можно считать оптимальным значение в 0.1.

Время работы алгоритма не зависит от данного параметра, что можно заметить по графику. Результаты отличаются незначительно.

Рис. 14: Зависимость времени выполнения программы от скорости обучения

5 Выводы

Было проведено исследование случайного леса и градиентного бустинга. Были выяснены оптимальные параметры для каждого из алгоритмов. На представленных данных случайный лес оказался лучше градиентного бустинга (у случайного леса RMSE < 120, а у градиентного бустинга >120).