Практикум на ЭВМ 2021/2022
Композиции алгоритмов для решения задачи регрессии

Хисматуллин Владимир, 317 группа

19 декабря 2021 г.

1 Введение

В данной работе разобраны основы методов композиций алгоритмов. Проведены эксперименты регрессии стоимости жилья из датасета о стоимости недвижимости в США.
Регрессия произведена алгоритмами

- Random forest (далее RF).
- Gradient boosting (далее GB).

Рассмотрены гиперпараметры алгоритмов, влияющие на их скорость работы и точность предсказания. Проведён детальный и теоретический анализ.

2 Обозначения

Везде далее введены следующие обозначения и понятия:

- l - размер обучающей выборки.
- N - размер признакового пространства.
- T - количество деревьев в ансамбле.
- d - максимальная глубина каждого из деревьев ансамбля.
- $\xi = \frac{n}{N}$ - отношение размера подпространства признаков, одинаковое для каждого дерева, к размеру всего признакового пространства.
- η - темп обучения алгоритма GB.

3 Эксперименты

3.1 Первичная обработка данных

Для корректной работы переведём все признаки в численные. Разобьём выборку в отношении 7:3 на обучающую и валидационную, после чего будем обучать композиции при разных гиперпараметрах и оценивать их качество на валидационной части.
Время обучения считается корректно и не включает в себя время сбора сторонней информации, такой как качество предсказания (RMSE-отклонение предсказания от таргета).

3.2 Зависимость характеристик RF от гиперпараметров

3.2.1 Теоретические основы

Напомним принцип работы алгоритма Random forest:
На t-ом шаге, $t = \overline{1, T}$: выбирается подпространство признаков F^t размера $n' = N \cdot \xi$
и подвыборка X^t с использованием bagging. На данной выборке и подпространстве обучается алгоритм $b_t = b(X_t, F_t)$, представляющий из себя дерево глубины не более d.
В итоге получается ансамбль алгоритмов $b(x) = \frac{1}{T}\sum_{t=1}^{T} b_t(x)$

3.2.2 Анализ гиперпараметров

3.2.2.1 Параметр T - количество деревьев

Сначала рассмотрим зависимость скорости обучения и RMSE от количества деревьев. Понятно, что время будет зависеть линейно, так как каждое новое дерево обучается аналогично предыдущим, а качество с некоторого момента не будет сильно меняться.

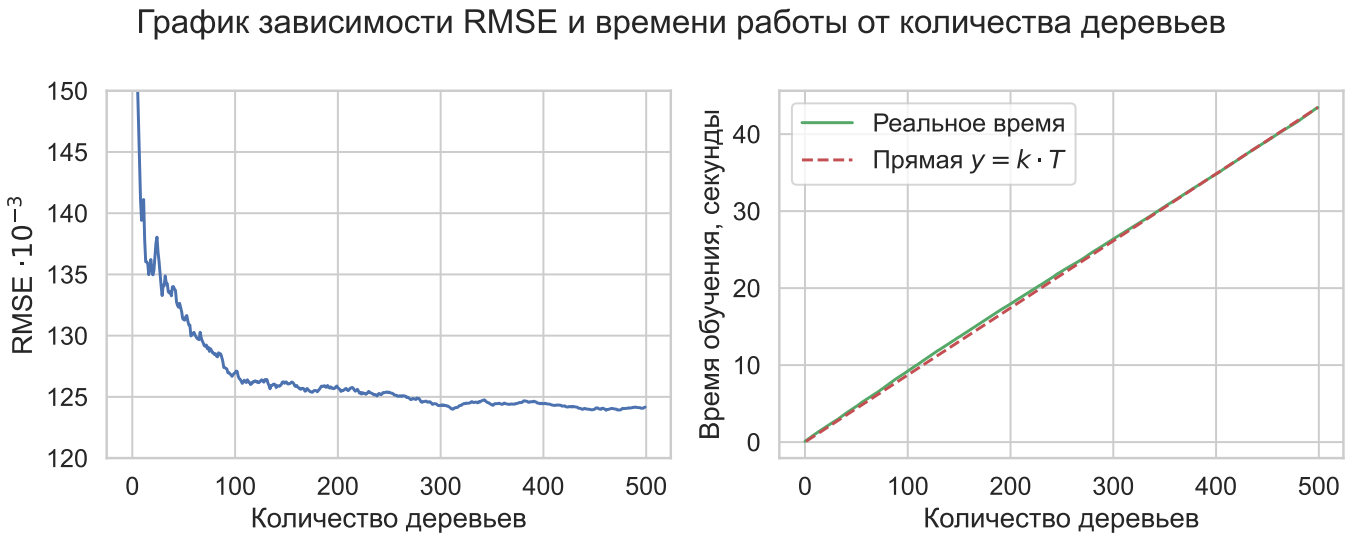

Рис. 1: Характеристики RF от количества деревьев: $d = None$, $\xi = 0.6$

При $T = 200$ график выходит на плато, хотя некоторое время заметны колебания, связанные с тем, что каждый новый алгоритм не зависит от других и до некоторой поры способен заметно влиять на предсказание.

3.2.2.2 Параметр d - максимальная глубина дерева

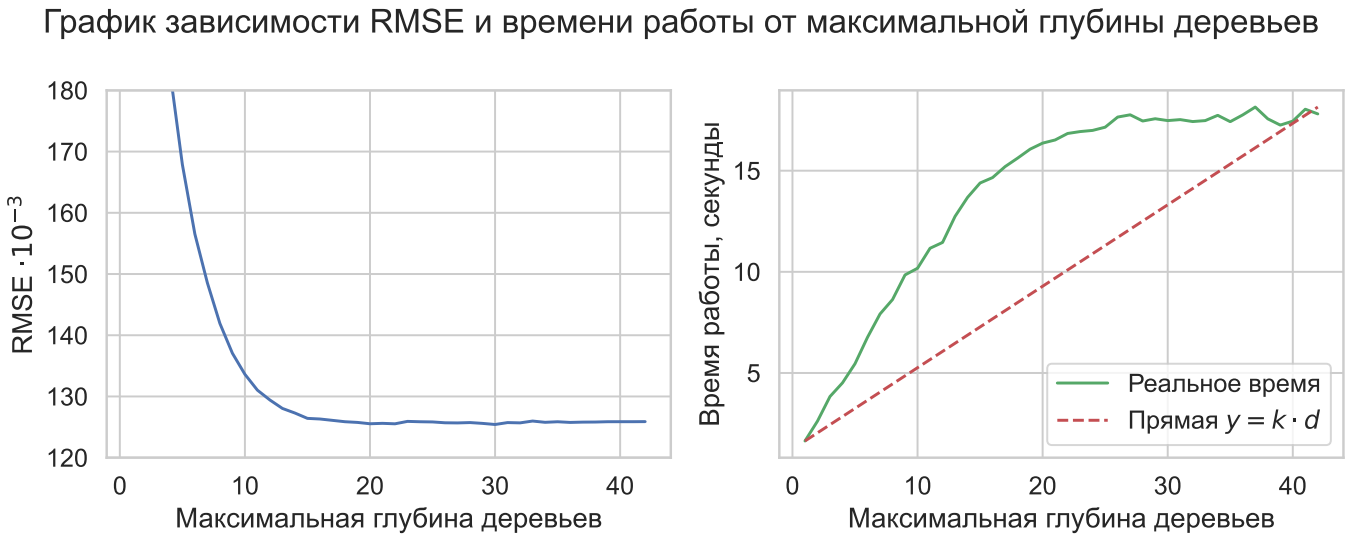

Рис. 2: Характеристики RF от максимальной глубины: $T = 200$, $\xi = 0.6$

Рассмотрим параметр максимальной глубины дерева. Отметим, что после определённой глубины увеличивать дерево становится бессмысленно.

1. В случае RMSE это означает выход графика на плато после некоторого значения максимальной глубины дерева.

2. В случае времени - это выражается в зависимости близкой к линейной на участке до $d = 20$ и флуктуациях после. В общем случае асимптотику оценить сложно, так как мы фиксируем не саму глубину дерева, а её максимум.

3.2.2.3 Параметр ξ - относительный размер подпространства

Рассмотрим зависимость работы от отношения размера подпространства каждого дерева к размеру всего пространства. Скорость растёт линейно, а RMSE быстро убывает до некоторого момента.

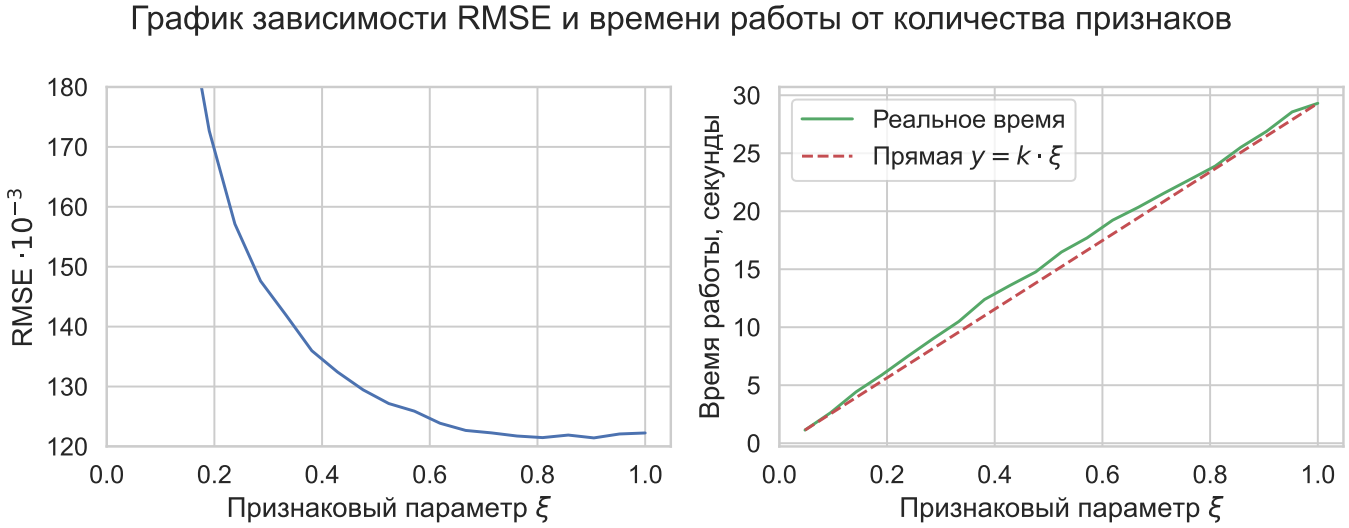

Рис. 3: Характеристики RF от размера подпространства: $T = 200$, $d = None$

Объясним природу такой зависимости:
Понятно, что, обучая 200 деревьев на малых подпространствах, мы получим 200 плохих алгоритмов, голосование по которым в общем случае даст плохое качество. С другой стороны, обучая их на практически одинаковых пространствах ($\xi \approx 1$), мы не улучшим качество, а лишь зря потратим время. С этой точки зрения, оптимальным параметром является $\xi \in [0.6, 0.8]$.

3.2.3 Выводы

Для получения наилучшего качества алгоритм Random forest требует построения большого количества глубоких деревьев по большим подпространствам признаков.
Характерные параметры: $T = 200$, $d = 20$, $\xi = 0.8$ Это объясняется тем, что все алгоритмы независимы и строятся одинаковым образом.
Достоинство данной модели - слабая зависимость от случайности в выборе подпространств и выборок, так как в пределе при $T \to \infty$ происходит их полный перебор.

3.3 Зависимость характеристик GB от гиперпараметров

3.3.1 Теоретические основы

Напомним принцип работы алгоритма Gradient boosting:
Будем представлять ансамбль в виде $b(x) = \sum_{t=1}^{T} \alpha_t b_t(x)$
Положим $f_0 = 0$ - начальный вектор приближений таргета, далее $f_t^i = \sum_{k=1}^{t} \alpha_t b_k(x_i)$.
Тогда на t-ом шаге, $t = \overline{1, T}$ выбирается подпространство признаков F^t размера $n' = N \cdot \xi$.
Затем оптимизируется $Q(b_t, \alpha_t, X) = \sum_{i=1}^{N} \mathcal{L}(f_t^i, y_i)$ по параметрам b_t, α_t.
Для этого $b_t(X^t)$ приближает антиградиент $-g_t = -\mathcal{L}'(f_{t-1}, y)$. В случае $\mathcal{L}(f_{t-1}, y) = (f_{t-1} - y)^2$ имеем $\mathcal{L}'(f_{t-1}, y) = 2(f_{t-1} - y)$. После чего решается задача подбора оптимального значения α_t.
Для уменьшения эффекта переобучения вместо оптимального α_t используется $\alpha_t \cdot \eta$, $\eta \in [0, 1]$ - темп обучения.

3.3.1.1 Параметр T - количество деревьев

Графики для количества деревьев мало отличаются от 1. Отметим, однако, что в данном случае выход на плато происходит значительно ранее, при $T = 50$, и флуктуации не наблюдаются. Построение 100 деревьев RF занимало $\tilde{8}$ секунд, для GB это время в 1.4 раза больше.

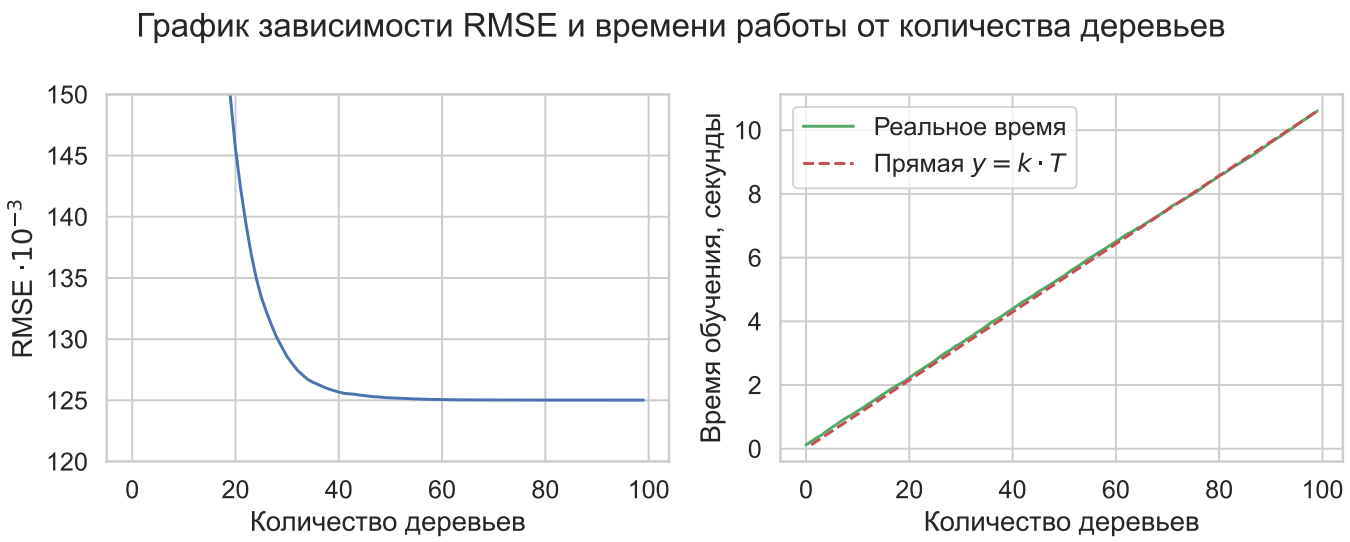

Рис. 4: Характеристики GB от размера подпространства: $d = None$, $\xi = 0.6$, $\eta = 0.1$

По понятным причинам время всё ещё зависит линейно, однако задача подбора оптимального α_t значительно увеличила время работы. Флуктуации отсутствуют, так как на каждой новой итерации строится уже не новое, независимое дерево, а дерево, немного улучшающее работу композиции, построенной итерацией ранее.

3.3.1.2 Параметр d - максимальная глубина дерева

График для максимальной глубины 5 отлично отражает важное отличие GB от RF:
Если RF не способен достигнуть необходимого качества при малой максимальной высоте (2), то GB за счёт зависимости алгоритмов может. Действительно, если раньше при построении T средних деревьев ($d \approx 6$) мы получали T средних алгоритмов, которые давали не самое

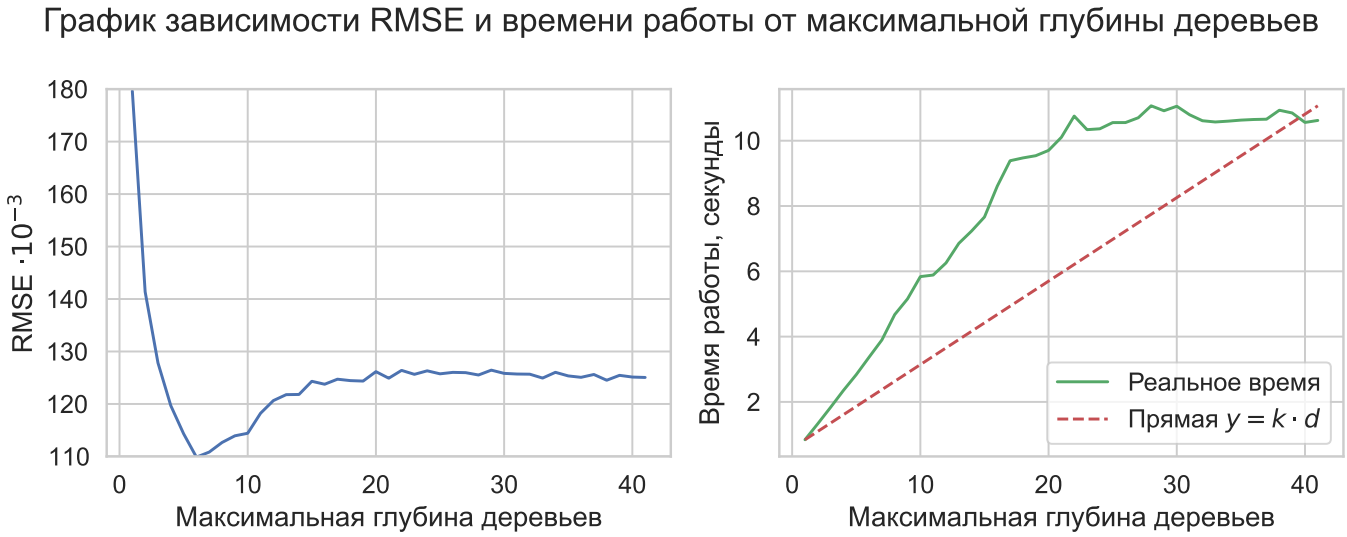

Рис. 5: Характеристики GB от максимальной глубины: $T = 100$, $\xi = 0.6$, $\eta = 0.1$

удачное качество, то теперь мы получаем T алгоритмов, которые оптимально улучшают друг друга, суммарно работая лучше, чем при большой глубине.
Как и ранее, после $d \approx 20$ изменение данного гиперпараметра не влияет ни на качество, ни на время работы, так как базовый алгоритм не может улучшить результат.

3.3.1.3 Параметр ξ - относительный размер подпространства

Зависимость от размера признаковых подпространств тоже меняется качественно: Теперь при

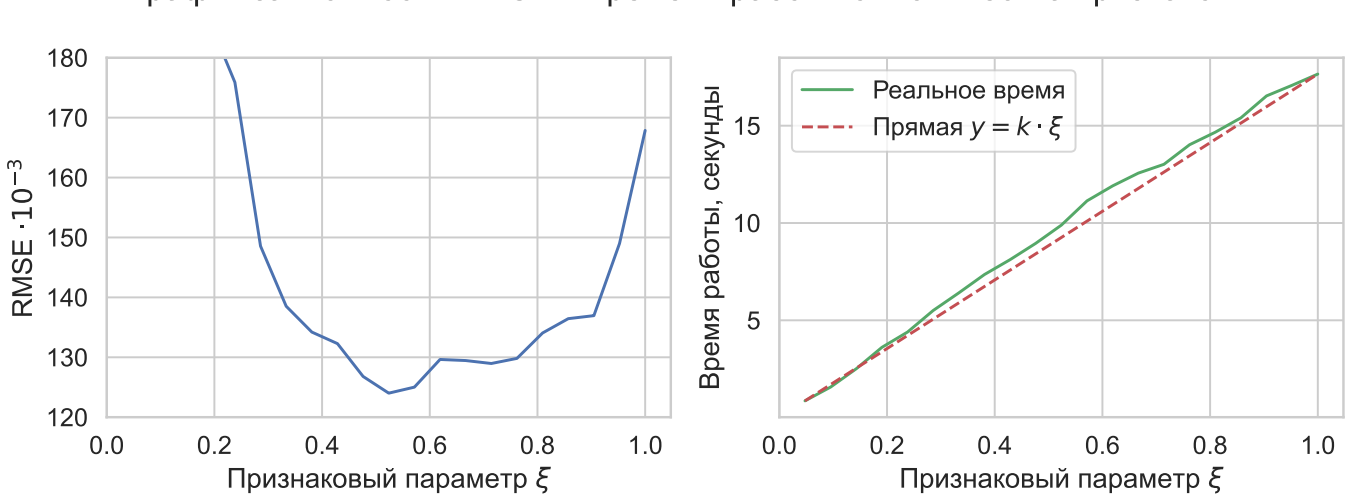

Рис. 6: Характеристики GB от размера подпространства: $T = 100$, $d = None$, $\eta = 0.1$

больших ξ композиция быстро переобучается и выдаёт плохое качество. Связано это с тем, что при чередовании признаков переобучение значительно уменьшается. А при их повторении алгоритм способен быстро подогнаться под обучающую выборку. График зависимости времени работы немного отличается от линейного, так как с ростом ξ увеличивается время обучения

деревьев, но не оптимизации α_t.

3.3.1.4 Параметр η - темп обучения

Темп обучения - ключевой параметр алгоритма GB. При больших η алгоритм очень быстро переобучается, что очень хорошо видно на левом графике. Отметим, также, что, так как при переобучении $f - y \approx 0$, то временные затраты на поиск оптимального α_t, ровно как построения дерева по $-g_t \approx 0$ уменьшаются.

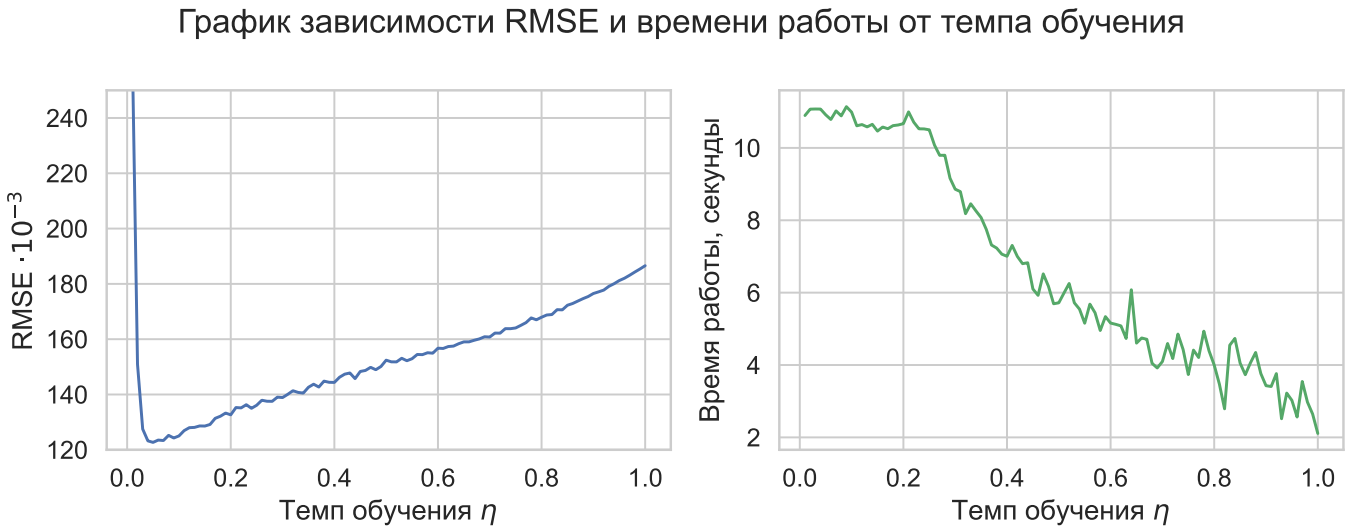

Рис. 7: Характеристики GB от темпа обучения: $T = 100$, $d = None$, $\xi = 0.6$

3.3.2 Выводы

Метод Gradient boosting оптимально и быстро решает задачу регрессии. Наиболее удачно он работает как композиция небольших алгоритмов (с глубиной ≈ 6) с небольшой компетенцией ($\xi \approx 0.5$). Причём качество сильно зависит от темпа обучения, так как алгоритм способен быстро переобучиться.
Его недостатком относительно Random forest является существенная зависимость от первого приближения: При неудачном случайном выборе признаков первого дерева, приближение f_1 может иметь сильное смещение для некоторых объектов из обучающей выборки из-за свойств некоторого признака. Это сразу приведёт к качественному изменению работы всего алгоритма.
Наилучшими оказались значения параметров $T = 100$, $d = 7$, $\eta = 0.1$, $\xi = 0.5$.
Соответствующее RMSE $\approx 110 \cdot 10^3$.

4 Git-репозиторий

Специально для отчёта создан Git-репозиторий. В нём содержится 4 ветки: master, dev, experiments, report. dev - файл с кодом, о назначении остальных можно догадаться по названию.

5 Заключение

В ходе экспериментов были подобраны оптимальные параметры алгоритмов RF и GB, максимизирующие точность регрессии.
Подчёркнуты и объяснены закономерности, связывающие точность и скорость работы методов композиции с их гиперпараметрами.
Задание с веб-сервером, как и творческая часть не выполнялась, литература не использовалась, решения экспериментов не обсуждались.