Подход 1

1. Выбранные столбцы с пропусками: first\_blood\_team, radiant\_courier\_time. В первом, вероятно, что существую матчи, где ни одна из команд не сделала первое убийство в течение 5-ти минут от начала матча. Во втором - пропуски означают, что сторона света не купила курьера в течение первых 5-ти минут от начала матча.
2. Столбец, содержащий целевую переменную - «radiant\_win».
3. Время: 0:01:05.52; Число деревьев 30; Качество 0.6884550884891
4. Определённо, смысл использовать больше 30 деревьев в градиентном бустанге ЕСТЬ, но время на обучение тоже будет расти. Большее число деревьев позволяет покрывать большее число остатков. Eсли выборка репрезентативна, тогда такой подход будет эффективным. Для увеличения скорости обучения при увеличении количества деревьев можно:

* Использовать подвыборку (теряем качество),
* Задействовать больше вычислительных ресурсов,
* Поиграть с параметрами (lineiar\_rate, max\_depth и другие).

Подход 2

1. Лучшее значение параметра C и качество при - {'C': 0.01} 0.7162964186682237. Качество логистической регрессии выше. Разница в качестве объясняется тем, что квадратическая функция потерь в градиентном бустинге накладывает слишком низкие штрафы. Логистическая регрессия работает быстрее.
2. При удалении категориальных признаков качество увеличивается совсем немного, причиной тому является стандартизация признаков, мы привели категории в сравнимый вид, что некорректно. Добавляя их в выборку, мы путаем классификатор, что отображается на качестве его работы. (Новые метрики качества - #{'C': 0.01} 0.7164823873552106)
3. 112
4. Наблюдается повышение качества, т.к. в классификации теперь учитываются различные наборы пиков для команд, что весомо влияет на исход игры. Метрики - {'C': 0.01} 0.7517947196144443.
5. Max = 0.9964051844033891, Min = 0.0035948155966109008.