Более подробное описание файлов.

**Файл Step1.ipynb**

это файл где медиана в пустых ячейках заполнена общая, а это не верно

* данные не нормализованные, и нет чистки данных.

Смотря на этот ноутбук было логично приступить к следующим шагам по оптимизации модели.

**ФАЙЛ KernelStep2.ipynb**

В этой модели учитывал что Field это просто id поля которое ни на что не влияет и было удалено. (Этот момент проговаривали на собеседовании )

Это Линейная регрессия в которой я уже обработал данные и нормализовал, выбросы были удалены, и медиана заполнена в зависимости от кластера.

**Проверка данных на дублирование**, так как максимальный id поля меньше чем количество ячеек в столбце соответственно возникло предположение того что происходит дублирование ячеек, для проверки моей гипотезы я выбрал такие парламенты «Field","Year","Cluster","Area","Yield" так как у нас площадь поля уникальная ( с точностью до 3 знака после запятой ) то я с уверенностью могу утверждать о дублировании.

Для анализа выбросов сочетал [«Criteria»]=[«Area»]/["Yield", соответственно.

Кросс валидация - МЕТРИКА(R^2)

**array([0.43401269, 0.70508608, 0.31456666, 0.61526245, 0.17491436])**- значения кросс валидации деление на 5 частей

**RMSE = 17**

**array([0.63615911, 0.59821573, 0.63476868, 0.66694886, 0.72222351,0.42750253, 0.11192827])** - на 7 частей

**RMSE: 16,5**

Потому решил перейти к использованию других регрессивных моделей что бы уменьшить метрик по оценке модели.

**Файл CatBLR.ipynb (catboost regression)**

Для выбора колонок и можно было использовать gridsearchcv, randomizedsearchcv, но я в тестовом задании смотрел на значимость параметров и в зависимости от таблицы я удалял в ручную их

Точность этой модели

RMSE(Среднеквадратическое отклонение): 14.2569

R2: 0.8334

**Файл Kernel.ipynb**

Это xgboost regression эта модель лучше работает для роботы с данными у которых только числовые значения.

Кросс валидация 3 повторения по 5 раз.

[0.82055749 0.80213989 0.78699346 0.81907174 0.80376643]- это просто модель XGBoost без параметров оптимизации. Сразу можем заметить то на сколько данная модель работает стабильнее предидущих. Следующий шаг это оптимизация параметров.

**[0.84235656 0.82609395 0.81835304 0.84513712 0.82953099] - после оптимизации**

Файл TestKernel это уже модель для прогноза результата с лучшей метрикой.

**Все Элементы чистки данных так же применялись и на тестовом наборе . Что так же способствует улучшению модели.**

**Графики закомментировал есть в финальном файле!**