

Добрый день, Анастасия!

Практически по всем пунктам критериев Вы отлично справились с заданием, замечательная и красивая работа! :)

Требования к оформлению:

- + на github: отличный подробный readme-файл, содержащий цели и задачи, описание данных и этапов работы над проектом;
- + структура оформления на GitHub: отличная идея разделить проект на крупные блоки и сделать для каждого отдельный файл;
- + структура оформления ноутбука: всё последовательно разделено на логические части, описаны цели и задачи, каждый этап работы, подробные сопровождающие пояснения и содержательные выводы о проделанной работе;
- + общие правила оформления: читаемый понятный код с грамотными названиями функций и переменных, графики построены по всем правилам визуализации, соблюдение стандартов PEP-8, отформатированные выводы в отдельных ячейках типа Markdown.

Анализ и обработка данных:

Отлично оформили этап первичного исследования данных: здорово, что посмотрели на наличие пропусков, дубликатов и выбросов; воспользовались `info` и `describe`!

Здорово, что построили столько оригинальных графиков для исследования признаков! Можете попробовать ещё функции `pairplot` - при помощи неё можно посмотреть сразу на распределения и взаимные зависимости всех признаков. Плюс за карту корреляций, это очень полезный и информативный график, который может дать возможность вовремя детектировать мультиколлинеарность признаков и избежать переобучения.

Отдельный плюс за тестирование статических гипотез и применение методов отбора признаков!

Единственное - сделать поменьше выводы на экран (убрать "процесс" работы алгоритмов - можно удалить или поставить параметр, например, у `CatBoost` это `silent=True`)

В остальном качественно сделали очистку данных и непосредственную подготовку их на вход модели, здесь нечего добавить.

Применение ML и DL:

Здорово, что последовательно "по всем правилам" протестировали все известные методы от простых к сложным. Зачастую случается так, что классические алгоритмы для некоторых задач показывают гораздо лучшие метрики.

`CatBoost` - сейчас самый популярный алгоритм, дающий, как правило, самые лучшие метрики, про него, кстати, часто спрашивают на собеседованиях - очень хорошо, что разобрались с ним.

Здорово, что пытались подобрать гиперпараметры при помощи `GridSearchCV` - чтобы "дождаться" результатов можете или уменьшить число и диапазон подбираемых гиперпараметров, или воспользоваться библиотекой `optuna` - она работает эффективнее за счёт того, что не обрабатывает те значения из предложенного диапазона, которые заведомо являются неоптимальным направлением.

Отдельно хочется отметить:

Хорошо, что пояснили весь код комментариями, в том числе в ячейках Markdown.

Отличная реализация варианта решения в продакшене!

Серьёзно подошли к выполнению проекта, это очень здорово! Отличная работа
работа 👍

Из советов и пожеланий:

Можете также посмотреть про модуль tqdm - с его помощью легко следить за прогрессом выполнения операций в цикле, обучения моделей и применения apply.

Спасибо за выполненное задание! Если возникнут вопросы, можете обратиться ко мне в канал с окончанием 08_final в пачке, постараюсь ответить на все вопросы и разобраться с моментами, которые вызывают трудности. Удачи в обучении!