Так уж случилось, что некоторым из нас приходится часто работать с данными. Моя программа предназначена как раз для таких людей и может им выполнять рутинную работу быстрее.

Мой проект называется QtAnalyzer и он представляет из себя инструмент для проведения автоматического анализа колонок в таблице, нахождения зависимости между колонками, конвертации баз данных из одного формата в другой.

Стоит сказать, что для использования программы пользователь должен знать хотя бы немного из анализа данных, например, понимать, чем категориальные признаки отличаются от числовых, или, к примеру, что такое ковариация и корреляция.

Начнём работу. В начале убедимся, что все библиотеки, перечисленные в requirements.txt, установлены. Далее с помощью Python запускаем скрипт loader.py:

Python loader.py

Открывается такое окно:

Изображение выглядит как текст

Автоматически созданное описание

Вводим путь к файлу и нажимаем на кнопку подтвердить. При работе с SQLITE базами данных придётся также указать и название таблицы.

Изображение выглядит как стол

Автоматически созданное описание

Здесь мы видим перечисление всех колонок в таблице. С помощью кнопок в верхнем правом углу можно сохранить таблицу в формате, отличном от того, который был загружен.

Чтобы провести сравнение двух колонок, нужно пометить первую, как целевую, и проанализировать вторую колонку. Найдём зависимость цены автомобиля от года его выпуска.

Выделяем колонку Price и нажимаем справа кнопку «Сменить статус». Это сделает колонку с ценой целевой переменной. Далее выделим колонку Prod. Year и нажимаем «Провести анализ»:

Изображение выглядит как стол

Автоматически созданное описание

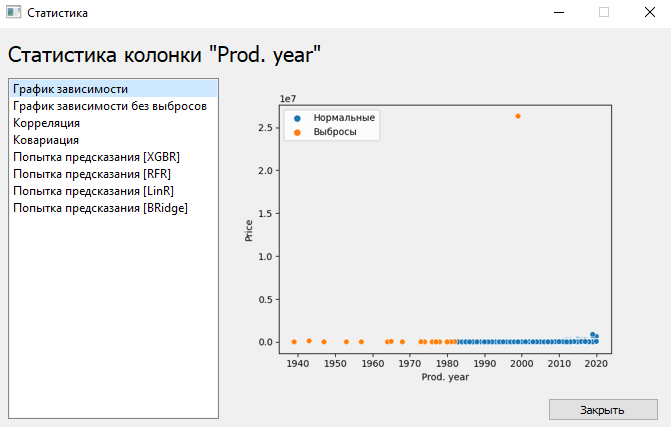
Анализ больших данных – дело не мгновенное. Возможно, придётся подождать около 10 секунд.

Изображение выглядит как текст

Автоматически созданное описание

Видим такой интерфейс. Нажав на нижнюю кнопку «Сохранить всё», можно выгрузить все файлы, полученные в ходе анализа, например, в отдельную папку.

Выделяем «Взаимосвязь колонок …» и нажимаем «Открыть»



Перед нами открывается форма просмотра проведённого анализа. Например, перед нами график зависимости двух переменных. Кстати, может показаться, что программа неправильно выделила выбросы. На самом деле, тем самым она сильно подняла корреляцию данных, что хорошо для предсказания.

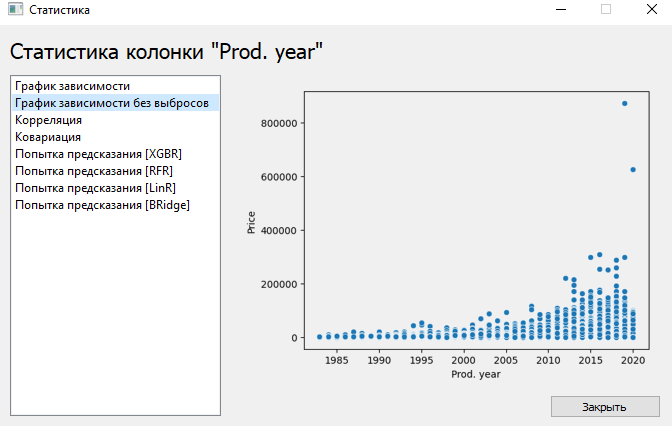
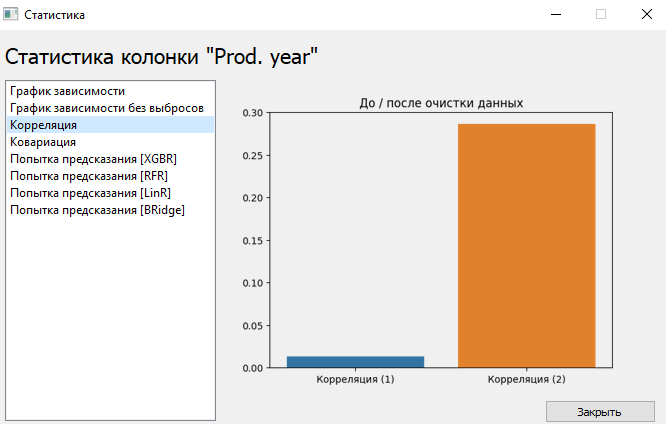


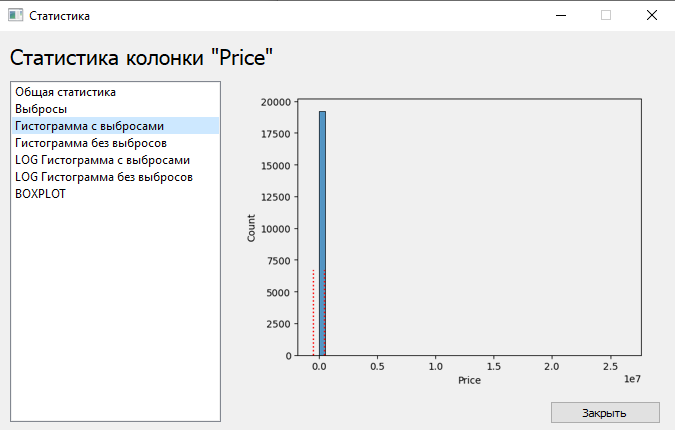
График без выбросов



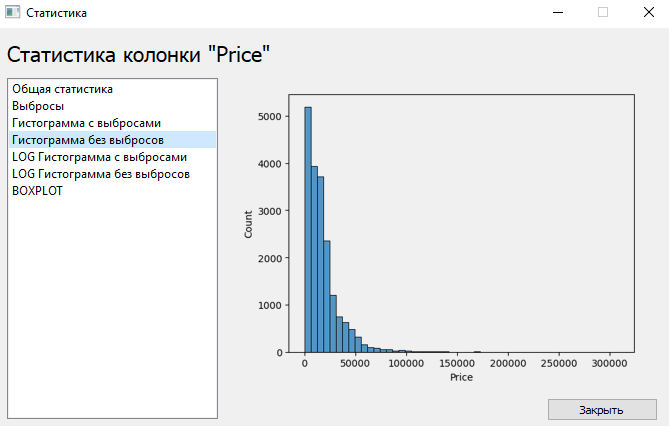
Повышение корреляции

Ну хорошо, теперь проведём анализ самой колонки «Price». Выходим в главное меню, выделяем по нужную колонку и нажимаем «Провести анализ».

Тут тоже есть множество данных о колонке, например, взглянем на удаление выбросов. Гистограмма до удаления:



А теперь после удаления:



Хороший результат! Это была всё обучение пользованию моей программой.

Стоит сказать, что проект может быть легко расширен, например, можно добавить вывод другой информации при особом анализе. Вся простота объясняется хорошей ООП структурой анализатора.

Если бы я продолжил заниматься данным проектом, то сделал бы возможность анализа датасета целиком и экспорта моделей для предсказания целевых переменных.

Проект был сделан полностью на Python, применялся QtDesigner для разработки интерфейсов. Использовались следующие библиотеки: numpy, pandas, sklearn, scipy, PyQt5, sqlite3, xgboost, matplotlib, seaborn.