Отчёт по заданию

Строньский Патрык Марек — кандидат на вакансию Junior Data Scientist

Введение:
Главное задание это сделать модель для предсказания ли клиент получит кредит или отказ, а если кредит, будет ли он просрочен или нет. Каждый из 24 параметров может оказаться ключевом для классификации ли человек может получить кредит.
Есть 4 части задания:
1. Посмотреть на распределения значении для всех параметров и найти такие, где разница между распределениями для отказа кредита и для выдачи кредита самые большое
2. Обучить модель классификации для выданных кредитов
3. Обучить модель классификации для всех данных.
4. Сравнивать итоги

Часть №1 – анализ данных
Во первых покажу распределение данных в двух случаях — когда клиент:

- не получил кредита вообще (отказ)
- получил или не получил кредита (все варианты)

Целью этого анализа является поиск параметров существенных для предсказания ли клиент получит кредит или нет. Здесь можем увидеть что есть две главные группы — люди какие получили отказ от кредита (bad == NaN) либо получающие кредит (bad != NaN). У первых банк нашёл признаки какие указывали риск что у человека есть серьёзная вероятность что он/она не возвратит деньги. У второй группы зато, этот риск является маленьким и банк дал этим людям возможность получить кредит. Эти люди либо его просрочили, либо возвратили кредит. Я буду смотреть на данные как отличаются люди какие получили отказ на фоне всех клиентов.

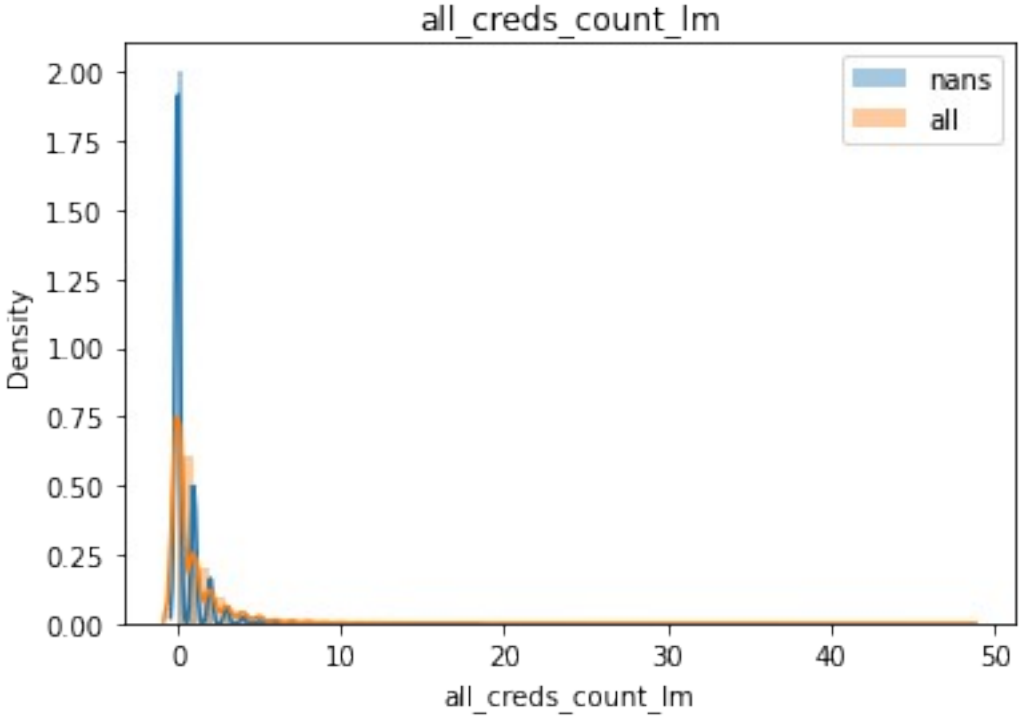
Из-за того что переменных много, я решил чтобы использовать Kernel Density Estimation (KDE) чтобы найти распределение переменных и сравнить их вместе. Для всех переменных я считал разницы между функциями KDE и сделал сумму их квадратов (чтобы избавиться всех минусов).

Там, где суммы самые большие — там разницы между распределениями тоже большие.

Эти признаки оказались самые важные:

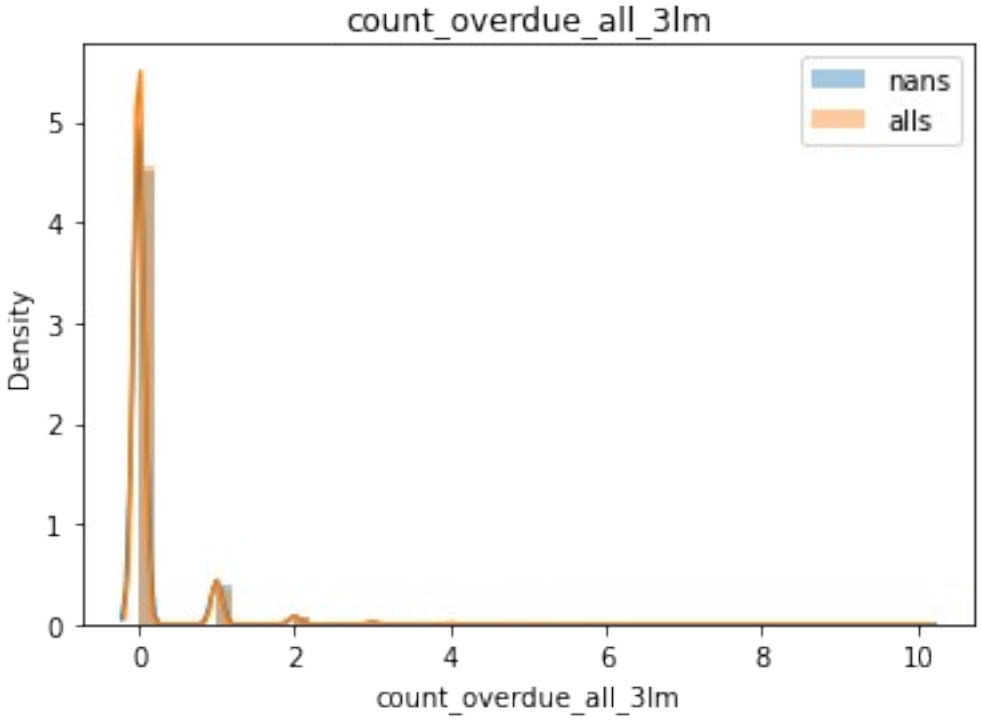
- all_creds_count_lm
- count_overdue_all_3lm
- work_code
- mfo_inqs_count_month
- mfo_closed_count_ly

all_creds_count_lm - Количество кредитов, взятых за последний месяц



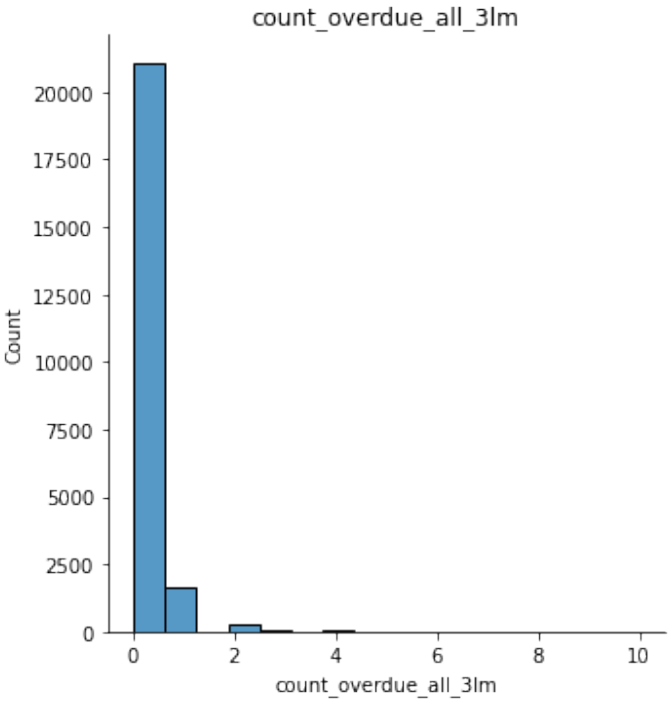
Как можно здесь увидеть, больше людей получают кредиты когда у них есть уже кредитная история. Кроме того больше людей получает кредит когда они уже взяли кредит в последнем месяце. Это может быть связано со созданием кредитной истории. Кроме того, другие призраки тоже могут иметь значение — как например то, что у человека есть банк какой даёт ему кредит и этот человек берёт новый, либо человек возвратил кредит какой взял месяц назад или хорошо его оплачивает (без просрочки). Тоже может быть у клиента есть кредитная карта.

count_overdue_all_3lm - Количество кредитов на просрочке, взятых за последние 3 месяца

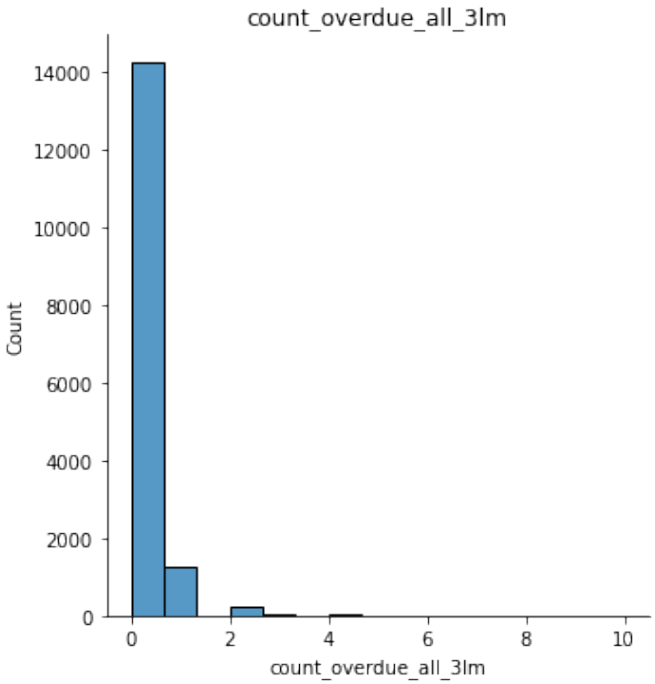


В том же случае можно увидеть, что разница просто в количестве случаев. Распределение не отличается никак друг от друга.

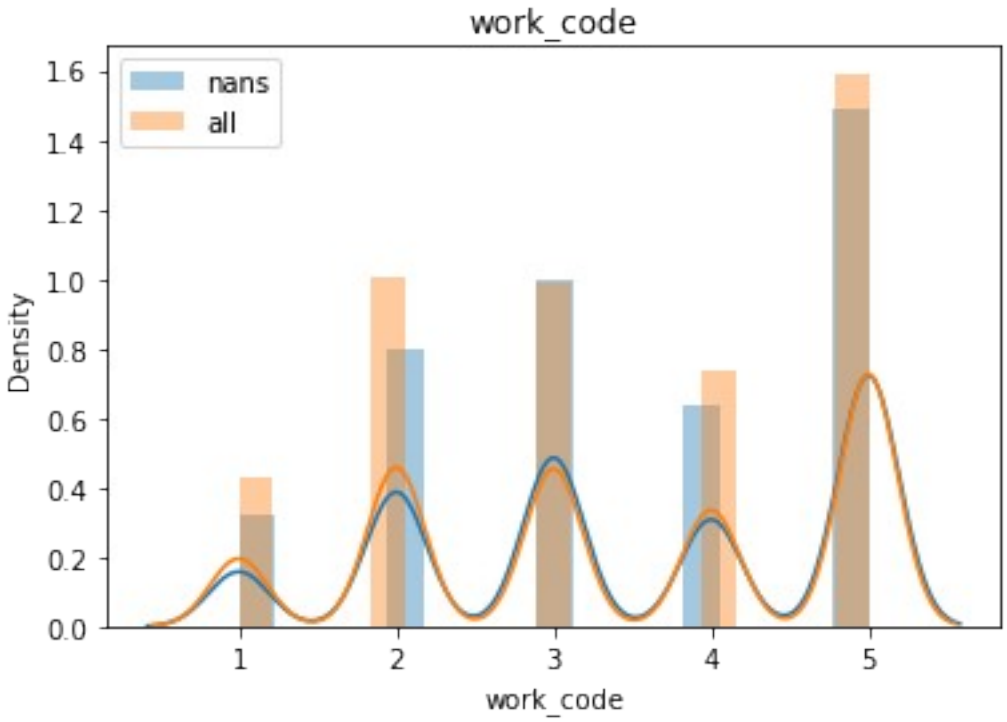
Count_overdue_all_3lm для всех случаев



Count_overdue_all_3lm кргда клиент получил отказ

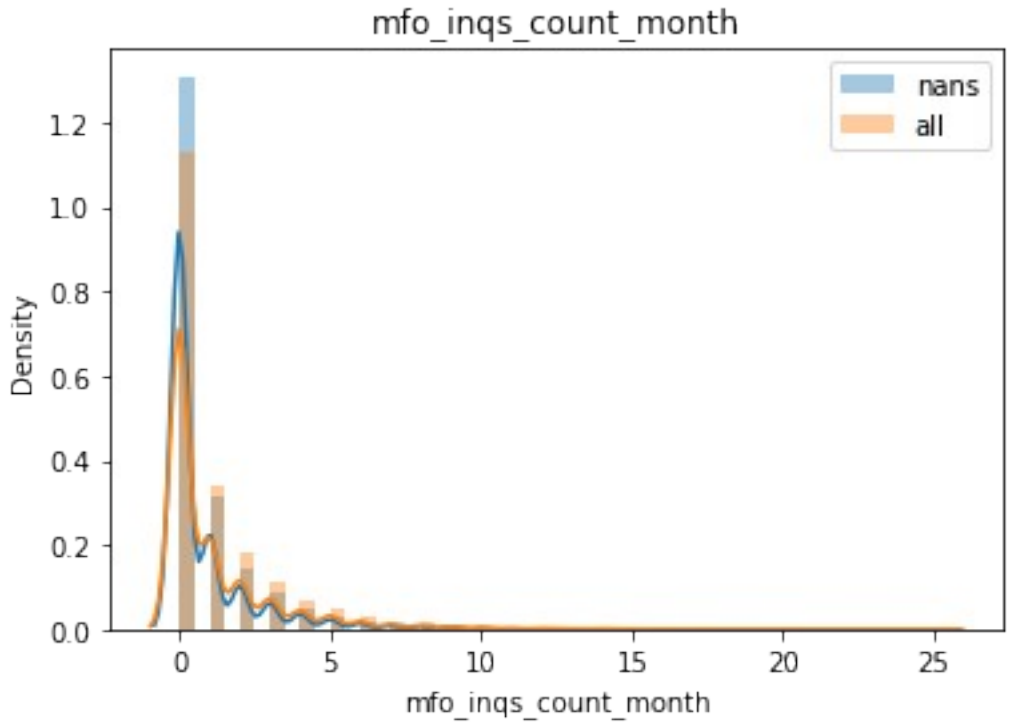


work_code - Профессия. 5 - рабочие профессии (слесарь, токарь). 3 - офисный работник (бухгалтер, программист). 1 - госслужащий (полицейский, медсестра)



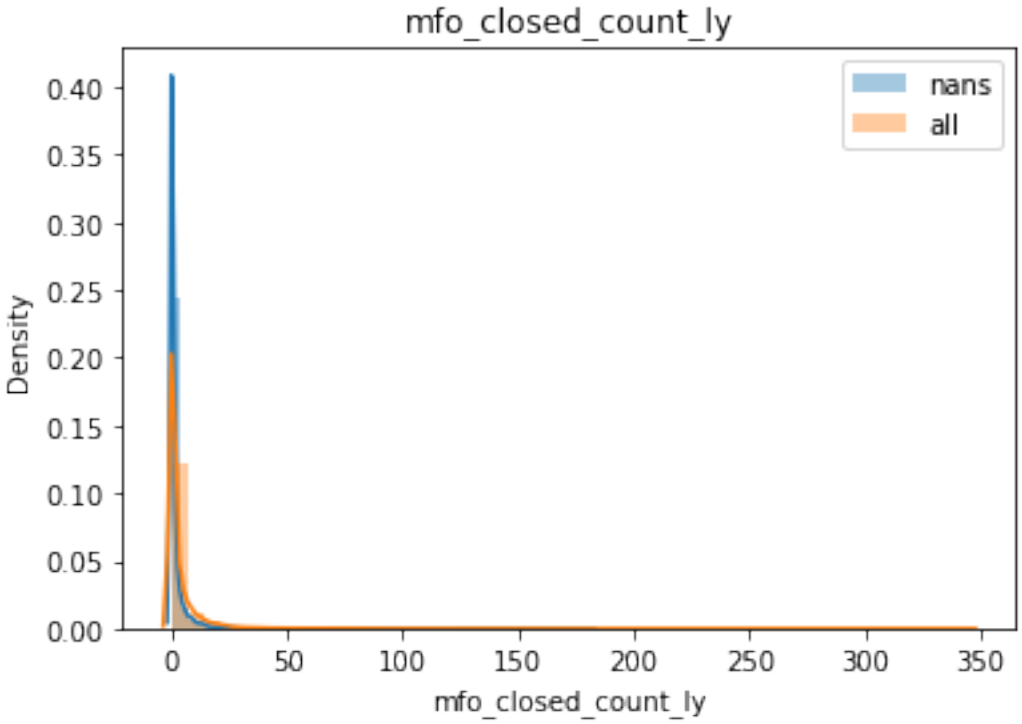
Тут можем заметить что профессия имеет значение при получении кредита. Если например кто-то является программистом (№2) у этого человека вероятность получения кредита большая чем если человек является бухгалтером (№3) например.

mfo_inqs_count_month - количество запросов на кредиты в другие МФО



Тут очевидно что если клиент больше спрашивает про кредит вероятность получения его растит.

mfo_closed_count_ly - Количество закрытых МФО кредитов, взятых за последний год



Тот график показывает как кредитная история связана с возможностью получения кредита. Люди получают кредит когда у них есть много закрытых кредитов.

Часть №2 — обучить модель только на выданных кредитах.
Модель я обучил на основах анализа данных используя KDE. Так как и в первом случае, только поделил датасет на две выборки — там где кредит просрочен и там где возвращён. Дальше я построил модель используя Random Forest и KNN алгоритмы.

Часть №3 — обучить модель только на всех данных.
В том случае я поступил похоже как в первом. На самом деле, я взял переменные какие я анализировал в первой части и на их основе сделал первые модели. Позже добавил ещё несколько переменных и получил результат. Алгоритмы те же самые как в части №2.

Часть №4 — сравнить итоги
Я получил разные результаты. Для первой части самый большой оказался KNN для переменных 'count_overdue_all_3lm', 'work_code', 'mfo_inqs_count_month', 'all_creds_count_lm', 'delay_more_sum_all'. С параметром n_neighbors=30 точность это 0.73.

Для второй для 5 переменных 0.766 это самая высокая точность для KNN. Переменные для создания модели это: 'all_creds_count_lm', 'count_overdue_all_3lm', 'work_code', 'mfo_inqs_count_month', 'mfo_closed_count_ly'. Немножко лучшую точность получил когда взял больше параметров - 'count_overdue_all_3lm', 'work_code', 'all_creds_count_lm', 'mfo_inqs_count_month', 'bank_inqs_count_quarter', 'mfo_closed_count_ly', 'all_creds_count_all', 'delay_more_sum_all', 'region', 'cred_day_overdue_all_sum_all'. К сожалению точность только чуть-чуть высшая — 0.769.

Заключение
После анализа данных мне удалось добиться точности более 75%. Чтобы исправить итоги ещё больше можно подумать над лучшим подбором параметров, рассмотреть больше случаев и попробовать больше алгоритмов.

Мой Github с проектом https://github.com/PatrykStronski/Devim_test