

<p><b>Сфера / Область применения</b></p> <p>Пользователи: банки и финтех</p> <p>Данные:  <a href="https://www.kaggle.com/datasets/sgpjesus/bank-account-fraud-dataset-neurips-2022">https://www.kaggle.com/datasets/sgpjesus/bank-account-fraud-dataset-neurips-2022</a></p> <p>Пререквизиты: скрытые данные банка</p>	<p><b>Цели</b></p> <p>Тип: рекомендательный</p> <p>Идея: создание решения для вычисления банками мошеннических аккаунтов на основании данных о пользователе</p> <p>Сценарии: банк предоставляет данные о пользователе по нескольким обязательным критериям, получает предположение о том, мошеннический аккаунт или нет, альтернативный сценарий - при наличии дополнительных данных о пользователе могут быть указаны для более точного результата</p>	<p><b>Функциональность</b></p> <p>Хранение: не предусмотрено</p> <p>Обновление: нет</p> <p>Пользовательская настройка: можно регулировать количество параметров, заполняя данные аккаунта</p> <p>Персонализация: да (см. сценарии)</p> <p>Ограничения: работа с конфиденциальными данными</p>
<p><b>Методы</b></p> <p>ML: логистическая регрессия, бустинг</p> <p>не ML: агрегация, отбор по условию, стат. тесты</p> <p>Инструменты: R</p>	<p style="text-align: center;"><b>(Группа №22)</b></p> <p style="text-align: center;"><b>Определение мошеннических банковских аккаунтов</b></p> <p><b>Оценивание</b></p> <p>Метрики: точность предсказания, качество модели</p> <p>Экспертное: самооценивание</p> <p>Пользовательское: поскольку целевая аудитория - банки, протестировать приложение реальному пользователю-банку представляется не очень возможным, так что не подразумевается</p>	<p><b>Интерфейс</b></p> <p>Способ ввода: id клиента банка из загруженного датасета / данные об аккаунте по представленным параметрам</p> <p>Способ вывода: сообщение о том, является ли пользователь мошенником с вероятностью этого (текст)</p> <p>Сообщения об ошибках: progress bar</p> <p>Зависимость от контекста: при выборе заполнения расширенного набора параметров - возможность их ввести</p> <p>Объяснения: выводится пояснение работы модели - на какие показатели мы опирались и почему</p>

## Примечания:

- Сфера / Область применения
  - Пользователи: целевая аудитория, для кого приложение / кому интересны результаты исследования, есть ли разные группы пользователей с разными задачами
  - Данные: откуда данные, ссылка на данные / как планируется их собрать
  - Пререквизиты: нужно ли пользователям обладать определенными свойствами (иметь профиль в ВК, аккаунт на Steam, знать, что такое машинное обучение, смотреть аниме и т.д.)
- Цели
  - Тип: исследовательский / рекомендательный
  - Идея: о чем проект
  - Сценарии: ожидаемое поведение (пользователь вводит то-то, получает ответ о том-то), будут ли разные сценарии в зависимости от групп пользователей (например, новички идут по такому-то сценарию приложения (а-ля “Нажмите одну кнопку”), опытные в данной сфере получают вариант с возможностью настраивания (а-ля отбор по параметрам))
- Функциональность
  - Хранение: будет ли сохранение пользовательского ввода для дальнейшего использования, если да, то куда и как, как организовано хранение данных, необходимых для работы
  - Обновление: будет ли обновляться датасет с течением времени, если да, то как
  - Пользовательская настройка: что в поведении системы может контролировать пользователь (выбор алгоритма, выбор формы представления результата, показать или не показать объяснение и т.д.) -- вариант *ничего* тоже возможен
  - Персонализация: будут ли разные сценарии в зависимости от пользовательского ввода (например, новички / с опытом в области, с личным кабинетом / без личного кабинета, есть аккаунт ВК / нет аккаунта)
  - Ограничения: есть ли какие-то ограничения (на использование данных, на хранение данных, необходимость дополнительных технических ресурсов)
- Методы
  - ML: какие методы машинного обучения планируется использовать и для чего (кластеризация, предсказательные модели, тематические модели и т.д.)
  - не ML: какие еще методы анализа будут использованы (не вошедшие в предыдущий пункт) -- стат. тесты, сетевой анализ, базовый текстовый анализ (без ML который, просто с подсчетом частот), агрегация, отбор по условию, парсинг данных
  - Инструменты: язык R / Python / оба, приложения для сбора данных, другие внешние приложения (Google Docs, например)
- Оценивание
  - Метрики: как будет оценено приложение (качество модели / точность предсказания / скорость работы / адекватность рекомендации с точки зрения пользователя / общая удовлетворенность пользователя от работы с системой / удалось ли решить задачу)
  - Экспертное: как организована оценка адекватности экспертами, кто эти эксперты (например, сами участники команды), предполагается ли вообще такое оценивание, например, самооценивание
  - Пользовательское: как организовано оценивание на пользователях, где и как будете набирать участников
- Интерфейс

- Способ ввода: что пользователь должен ввести (ответить на вопросы анкеты, о чем вопросы, оценить что-то из предложенных вариантов, ввести id для какого-то внешнего сервиса и т.д.)
- Способ вывода: что является результатом, как он представлен (текст, картинка, таблица, график, карта, ...)
- Сообщения об ошибках: что будет, если пользователь ввел недопустимые значения, как дать ему понять, что исправить, если рекомендация считается долго, как дать пользователю знать, что идут расчеты, а не все зависло
- Зависимость от контекста: меняется ли интерфейс в зависимости от ввода пользователя (например, дополнительные вопросы при положительном ответе на один из предыдущих)
- Объяснения: сопровождаются ли рекомендации объяснениями, почему ответ именно такой, или советами, что нужно дополнить или изменить, что еще будет интересно