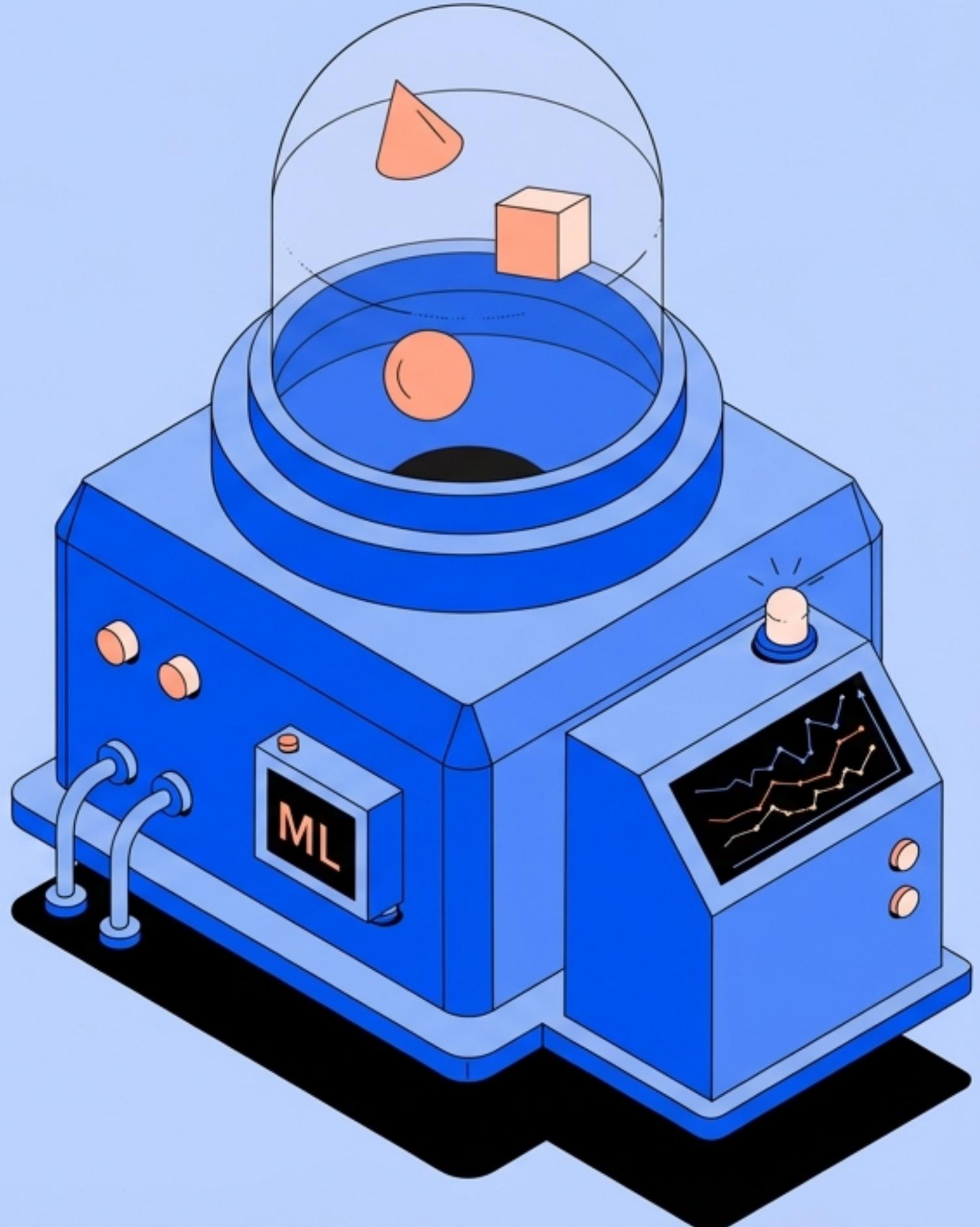


Итоговый проект: Модель кредитного риск-менеджмента

Финальный рубеж курса Machine Learning Junior

Добро пожаловать на финишную прямую. Вы уже освоили регрессию, классификацию, NLP и работу с PySpark. Теперь ваша задача — объединить эти навыки в комплексный продукт. Вы создадите инструмент, который реальные реальные банки используют для защиты миллиардных активов.

Цель: Перейти от обучения к профессиональной практике.



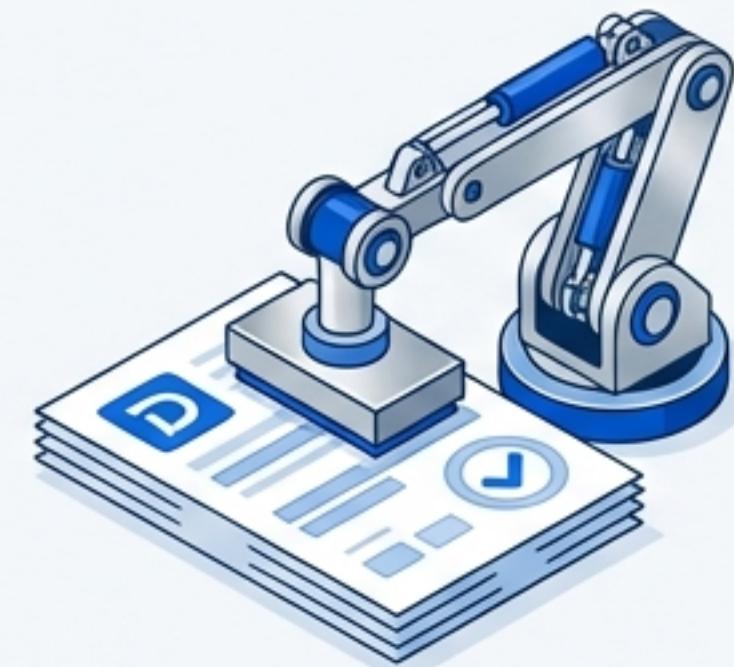
Бизнес-задача: Баланс между доверием и риском

Контекст



Банки живут за счет кредитования, но каждый заем несет риск. Ручная проверка заявок — это медленно и дорого. Бизнесу нужна скорость принятия решений без потери качества.

Решение



Роль ML-модели:

Роль ML-модели: Автоматический скрингинг, который подсказывает менеджеру: «Доверять» или «Отказать».

Главный враг — Дефолт: Ситуация, когда клиент не платит по кредиту более 90 дней.

Ваша миссия: Предсказать вероятность дефолта, чтобы **предотвратить** финансовые потери банка.

Цель проекта: Бинарная классификация



Вход

Атрибуты заемщика, кредитная история, финансовые показатели.

Выход

Вероятность дефолта
(число от 0 до 1).

Результат

Снижение убытков банка за счет раннего отсеивания рискованных клиентов.

Ландшафт данных: Из чего состоит кредитная история

Идентификаторы

- id (заявка), rn (порядковый номер кредита)

Временные метки

- pre_since_opened (давность открытия)
- pre_pterm (плановый срок)
- pre_fterm (фактический срок)

Финансы

- pre_loans_credit_limit (лимит)
- pre_loans_outstanding (остаток долга)
- pre_loans_next_pay_summ (следующий платеж)

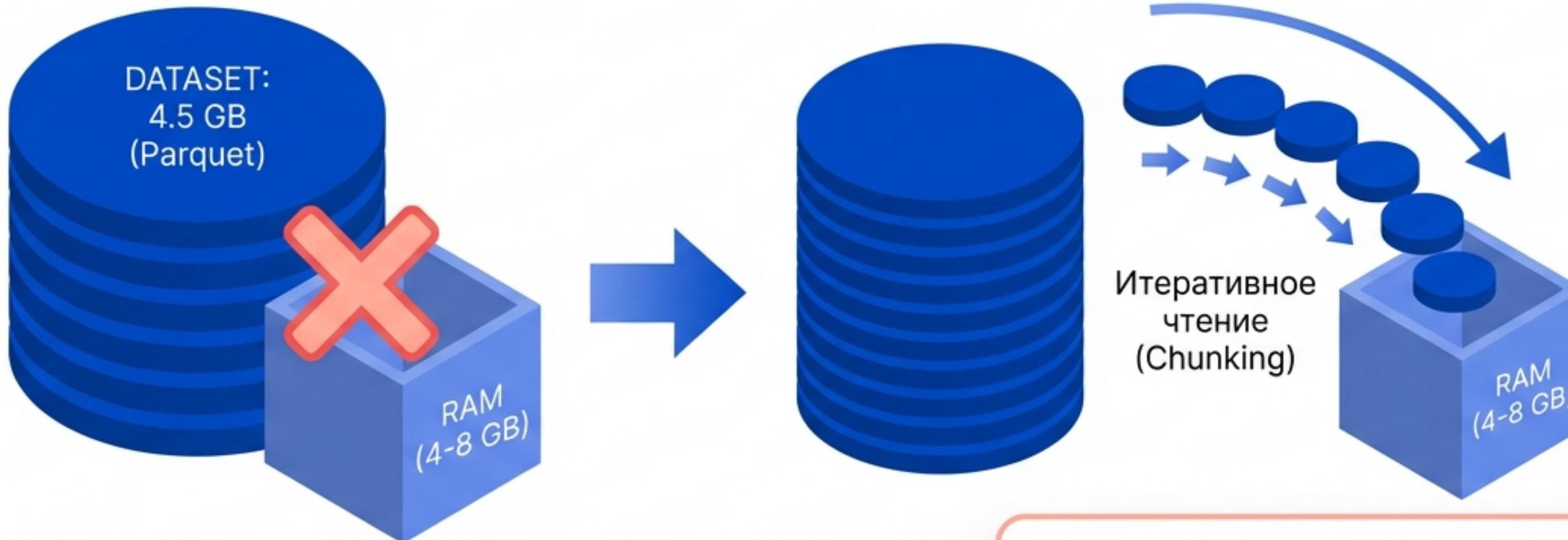
Просрочки (Маркеры риска)

- pre_loans_total_overdue (сумма просрочки)
- pre_loans5...pre_loans90 (количество просрочек по дням)

Кодированные признаки

- enc_paym_{0..N} (статус платежей)
- enc_loans_credit_type (тип кредита)

Технический вызов: Big Data и ограничения памяти



В реальных проектах данные редко помещаются в RAM. При распаковке объем увеличивается в разы.



Pro Tip

Решение: Мы предоставляем скрипт для пакетной обработки. Это навык уровня Middle — уметь работать с данными, превышающими объем оперативной памяти.

Этап 1: Сбор данных и Feature Engineering

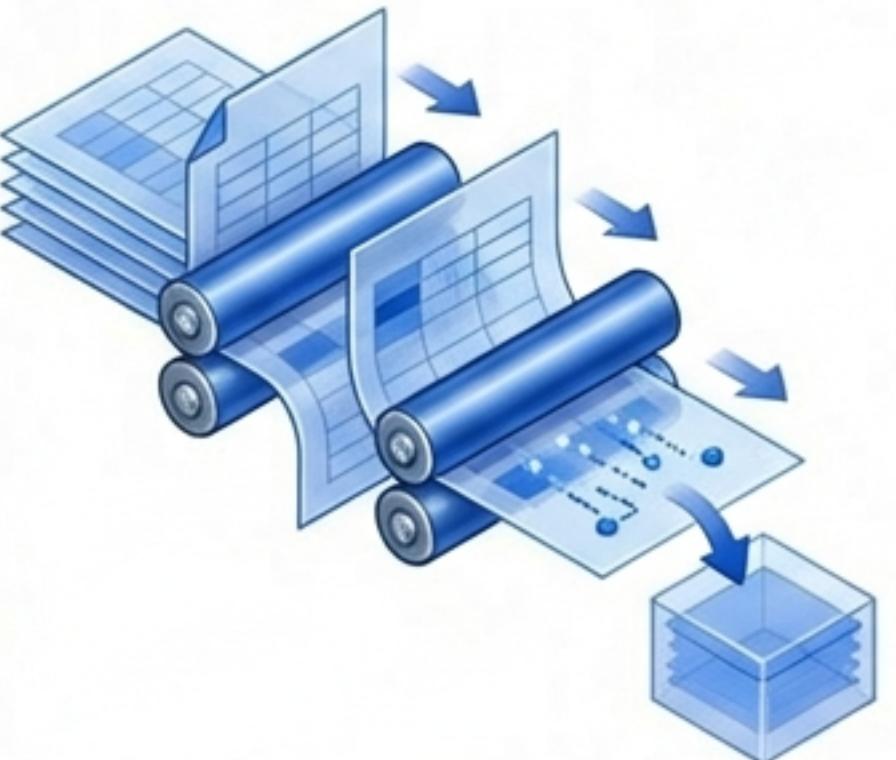
Эффективная обработка и обогащение данных для моделирования кредитного риска.

1. Распаковка



Работа с Parquet файлами.
Эффективное хранение и чтение.

2. Итеративное чтение



Считывайте файлы по очереди
(chunks), извлекая только
необходимые признаки, чтобы не
перегрузить память.

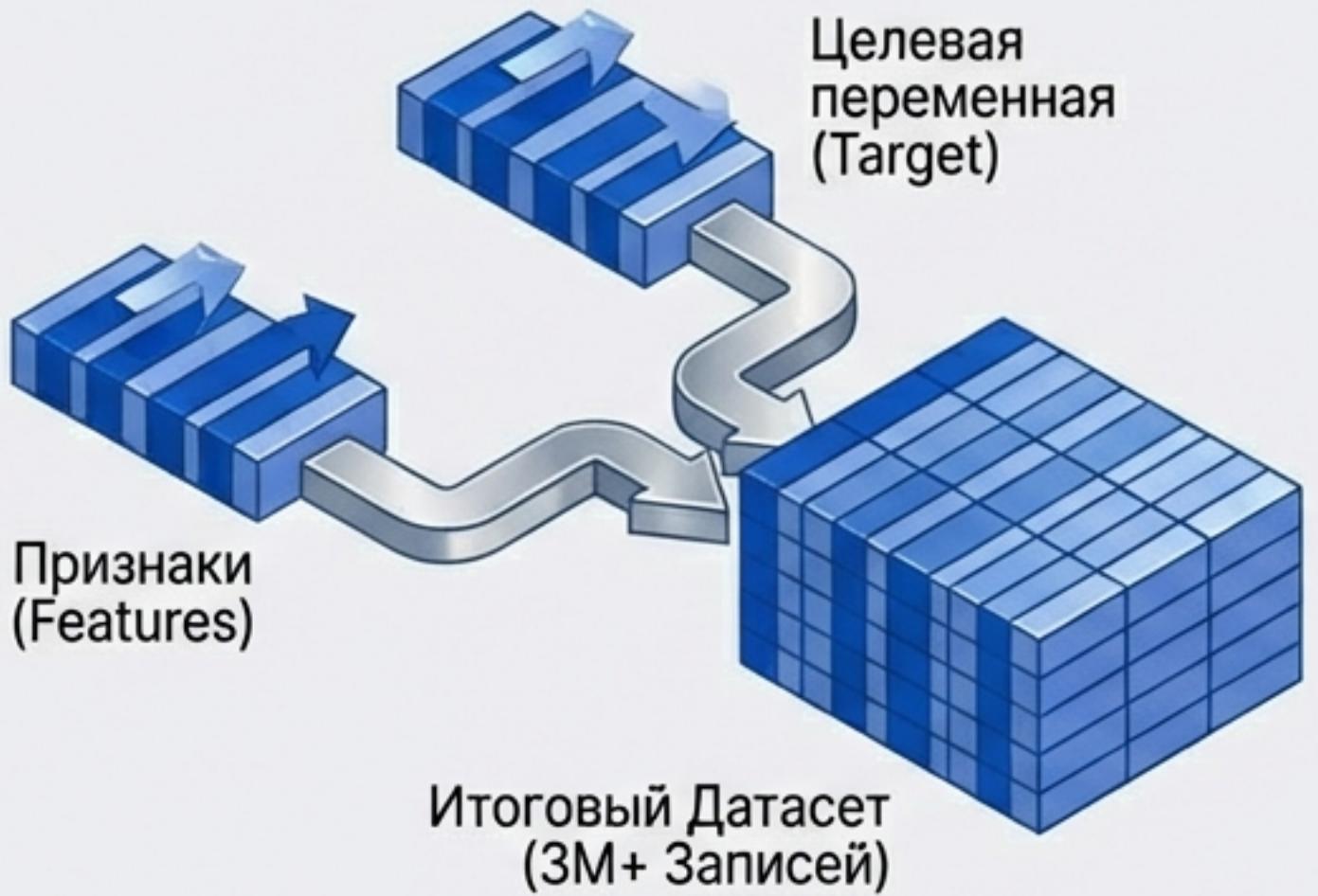
3. Генерация фичей



Не просто загружайте — создавайте
смысли. Используйте биннинг и
кодирование. Каждая фича должна
иметь обоснование влияния на риск.

Этап 2: Слияние и подготовка датасета

Объединение



Смержите подготовленные признаки с целевой переменной (Target). Результирующий датафрейм должен содержать около 3 миллионов записей.



Этап 3: Моделирование и валидация

Критически важные шаги для построения надежной и эффективной модели.



Разбиение:

Train/Test в пропорции 70/30 или 80/20.



Эксперименты:

Пробуйте разные алгоритмы.
Не останавливайтесь на первом.



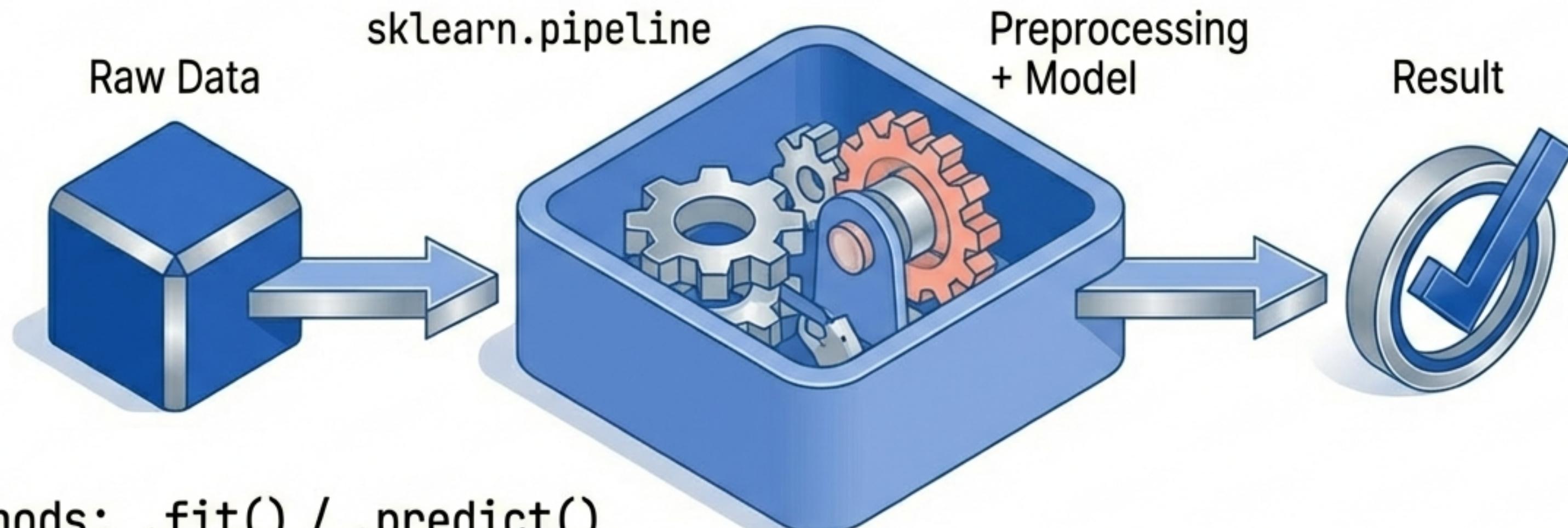
Тюнинг:

Подбор гиперпараметров обязателен.



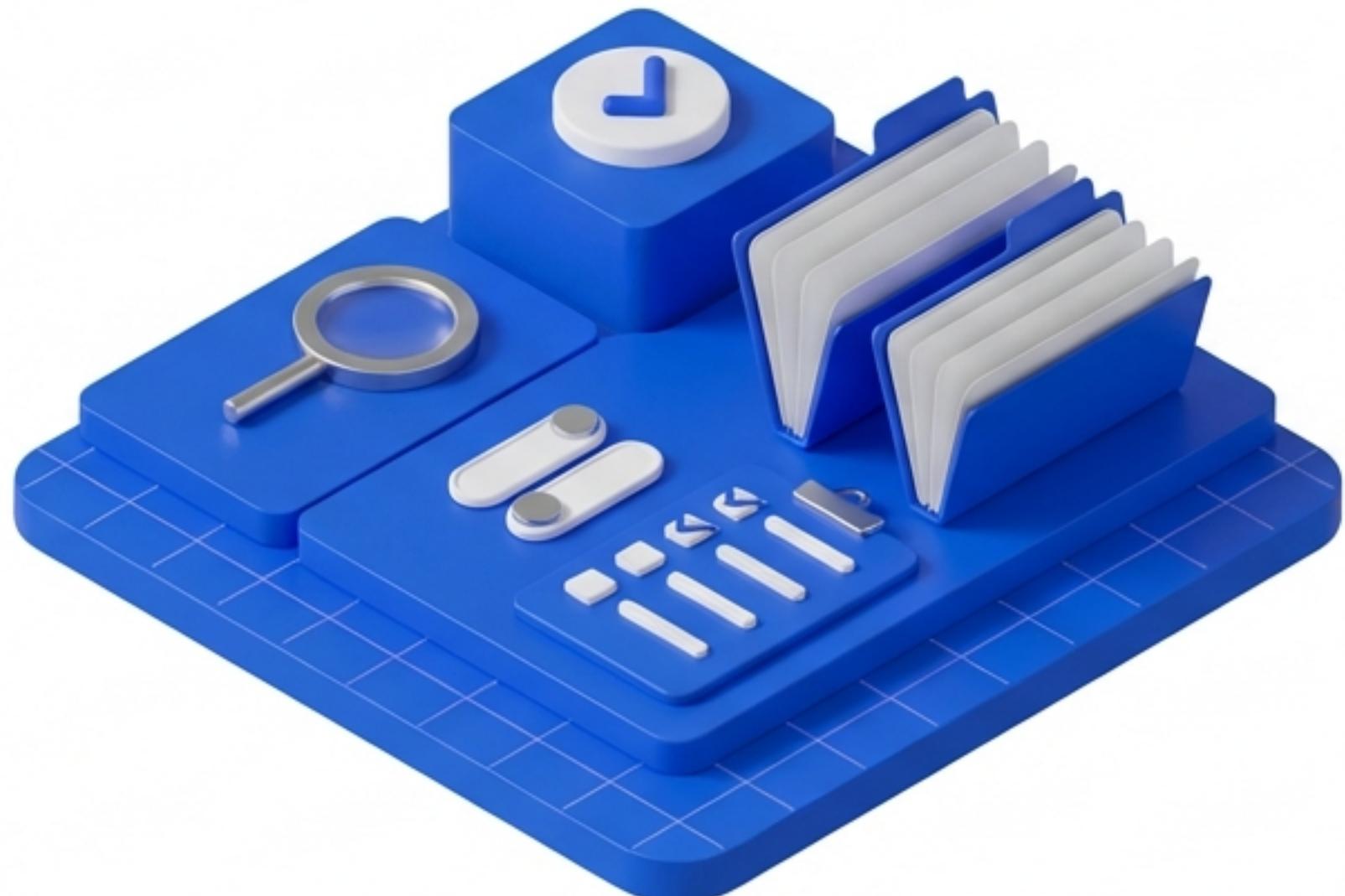
Это минимальный бейслайн. Результат ниже этой планки означает, что модель не готова к эксплуатации. Следите за переобучением!

Этап 4: Автоматизированный Pipeline



В продакшене код не запускают ячейками ноутбука. Нужен автоматизированный конвейер. Сохраните обученный пайплайн в бинарный формат `.pickle`. Это ваш финальный продукт.

Критерии успеха: Три столпа оценки



Качество (Quality)

ROC-AUC ≥ 0.75 на тестовой выборке.
Отсутствие переобучения.

Чистота кода (Code Hygiene)

Стандарт PEP 8. Осмысленные имена переменных. Читаемая структура.

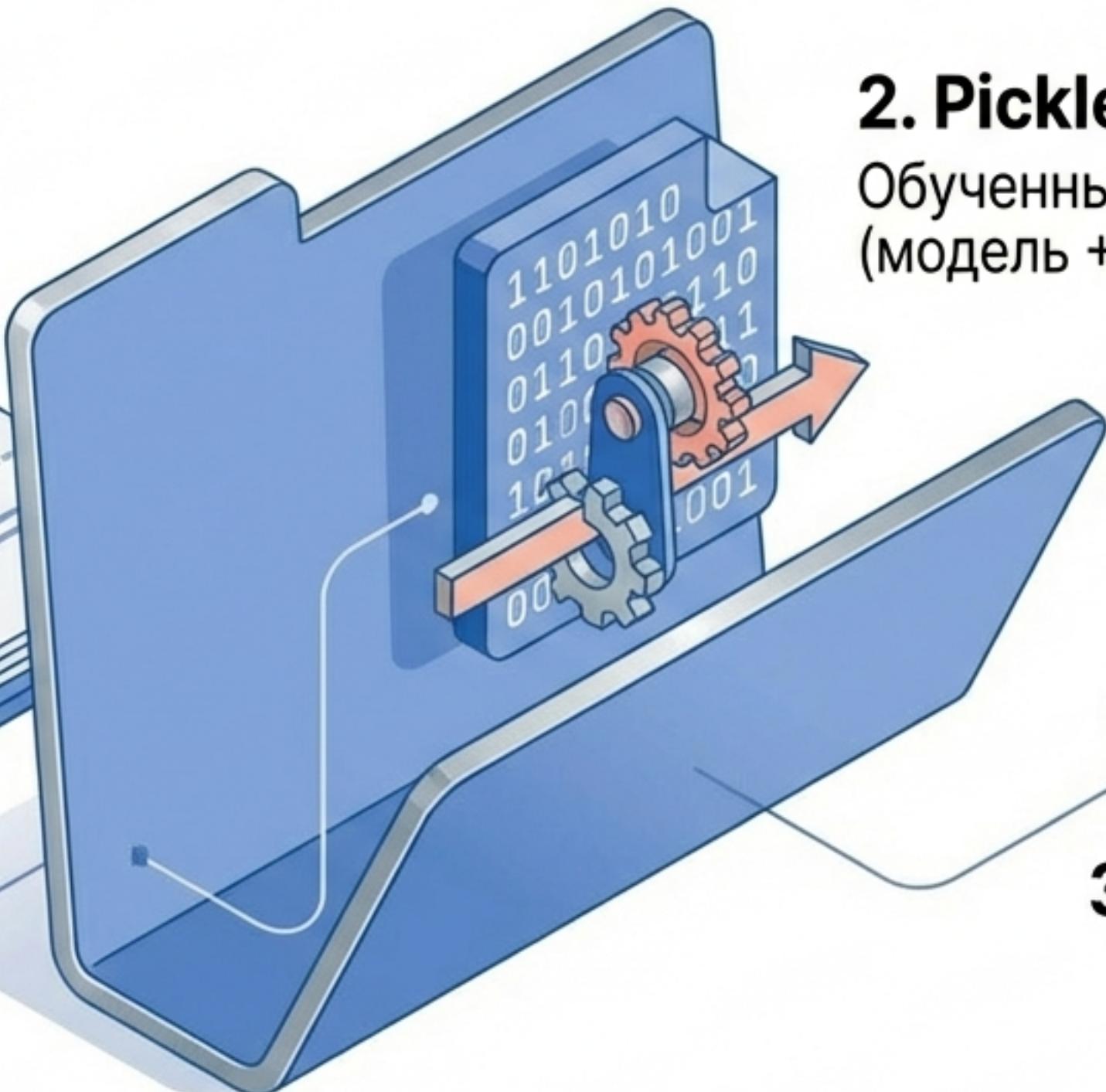
Научный подход (Science)

Минимум 3 полноценных эксперимента.
Обоснование выбора гиперпараметров.

Пакет материалов для сдачи

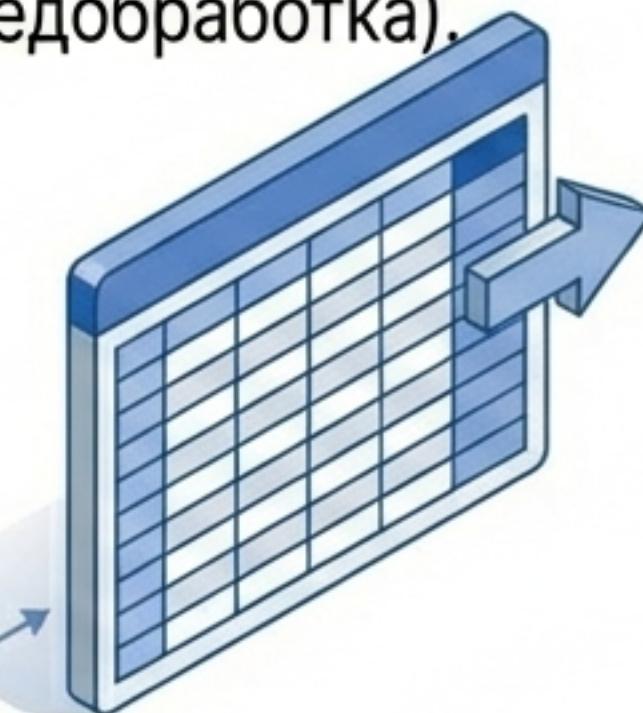
1. Jupyter Notebook

Полный код со всеми этапами решения.



2. Pickle файл

Обученный пайплайн
(модель + предобработка).



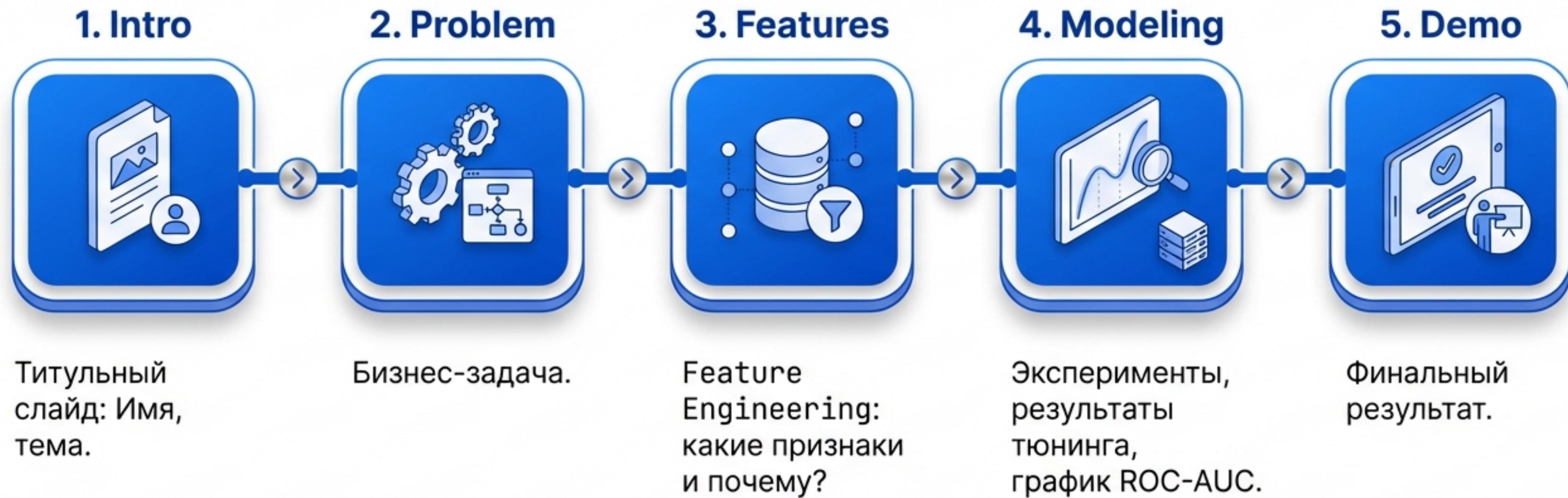
3. Файл с предиктами

Результаты на тестовой выборке.

Проект будет возвращен, если код не запускается или метрика ниже 0.75.

Структура защиты проекта

10 минут презентация + 10 минут вопросы



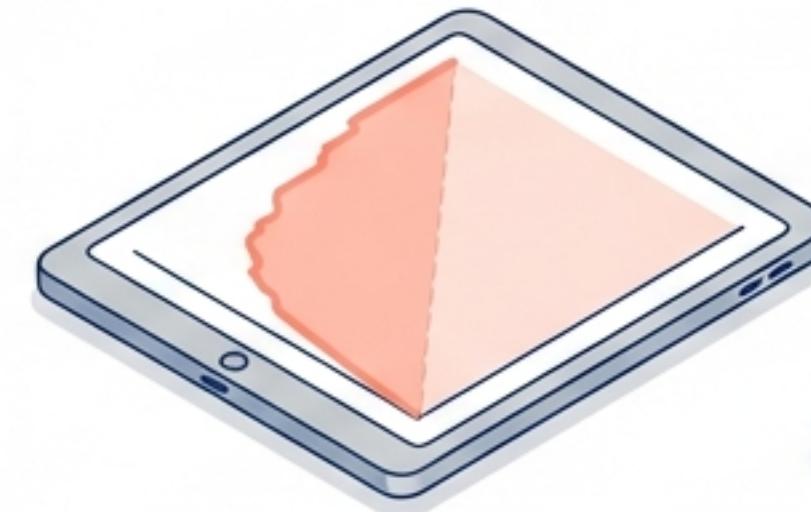
Советы для успешной защиты

Рассказывайте историю



Не просто перечисляйте функции. Объясните, почему вы выбрали именно эти фичи.

Рассказывайте историю



Визуализируйте успех

Покажите график ROC-AUC. Визуализация метрик убедительнее сухих цифр.

Покажите эволюцию



До



После

Сравните результаты «до» и «после» настройки. Это покажет вашу работу над качеством.



Будьте готовы к вопросам

- Почему эта **модель**?
- Как боролись с **дисбалансом**?
- Знайте свои данные.

Вы готовы к старту карьеры



Этот проект сложен и комплексен. Это симуляция реальной рабочей задачи Data Scientist'a.

Выполняя его, вы подтверждаете навыки: от проектирования нейросетей и РСА до обработки больших данных и построения пайплайнов.

Это мощный кейс для вашего резюме. У вас все получится!

Полезные ресурсы



PEP 8

Руководство по написанию чистого кода на Python.



ODS.ai

Материалы соревнования по карточным транзакциям.



Scikit-learn Documentation

Документация по Pipelines и метрикам.

**Желаем успехов в работе
над итоговым проектом!**

