1. **EDA**

* train.describe() - базовые статистики
* train.head()
* Анализ числовых признаков
  + дискретные
  + непрерывные
* Анализ категориальных признаков
* Анализ пропусков
* Анализ распределения целевой переменной
* Корреляция между числовыми признаками
* Нелинейная связь между признаками
* Pair Plor
* Scatter Plot
* Box Plot
* Категориальные переменные в разрезе целевой переменной
* Анализ распределения на обучении и тесте (проверка распределений на идентичность)
  + классический - тест Колмогорова-Смирнова.
  + Вычисление PSI и отбор признаков на их основе
  + Adversarial Validation - модный и современный

1. **Валидация** (webinar3\_validation.ipynb Kaggle)

* K-Fold / Stratified K-Fold - для небольшого датасета, где кросс-валидация выполняется не долго
* Hold-Out - для больших датасетов
* Проверить кросс-валидацию на устойчивость
* Проверка значимо ли отличаются скоры разных моделей с помощью t-критерий Стьюдента для связанных выборок
* Выбрасывание или не учитывание выбросов при валидации

1. **Feature engineering** (webinar4\_features\_part1.ipynb Kaggle)

* Масштабирование
  + нормализация
  + стандартизация
* Винсоризация
* Преобразование рангов
* Кодирование категориальных признаков (webinar5\_features\_part3.ipynb)
  + Label Encoding
  + Frequency encoding (частотное кодирование)
  + OneHot Encoding (если уникальных значений категориального признака не много)
  + Target Encoding
  + Mean-Target-Encoding
  + Дополнительная регуляризация
* Кодирование числовых признаков
  + Feature discretization - (quantization or binning), divides a continuous feature into a pre-specified number of categories (bins), and thus makes the data discrete.
  + Feature binarization is the process of tresholding numerical features to get boolean values. Or in other words, assign a boolean value (True or False) to each sample based on a threshold. Note that binarization is an extreme form of two-bin discretization.

1. **Feature selection** (webinar4\_features\_part2.ipynb Kaggle)

* Создание новых признаков:
  + количество заявок
  + аггрегировать признаки
  + GroupBy and Delta
  + Ratios
* Features Interaction - создание новых признаков как взаимодействие нескольких
  + Конкатенация нескольких категориальных признаков (например уровеня образования и семейного положения)
* Оценка полезности новой переменной
  + по коэффициенту корреляции Пирсона между этой переменной и целевой переменной
  + Kernel Density Estimate (KDE)
  + прокси-метрики

1. **Обработка пропусков** (webinar5\_missings.ipynb)

* Обнаружение пропусков
* Понимание что означают пропуски
* Численное определение недостающих значений
* Визуальное определение пропусков
* Поиск причины пропусков
  + msno
  + дендрограмма
* Обработка пропусков
  + Удаление
    - Списочное удаление / удаление строк
    - Удаление полных столбцов
  + Заполнение пропусков
    - Базовые способы замены пропусков
      * замена на константну
      * замена с использованием статистики (среднее, медиана или мода) для каждого столбца, в котором встречаются пропуски (SimpleImputer из sklearn)
    - замена пропусков в задаче с временной зависимостью
      * Линейная интерполяция
    - Методы заполнения пропусков на основе ML
      * Замена пропусков с помощью KNN
      * Многомерная импутация пропусков посредством связанных уравнений (MICE)
      * Алгоритмы, которые умеют в обработку пропусков

1. **Отбор признаков и интерпретация важности признаков** (webinar5\_feature\_selection.ipynb)

* на основе интерпретации поведения модели
* Классические меры важности XGBoost
* Permutation Importance
* sklearn-реализация
* eli5-реализация
* собственная реализация
* SHAP
* Отбор признаков и улучшение модели
* Base BoostARoota (webinar5\_BoostARoota.ipynb)

1. **Тюнинг гиперпарметров** (webinar6\_optimization.ipynb)

* Понять какие параметры сильнее всего влияют на модель и выбрать важные
* понимать как изменение гиперпараметра повлияет на процесс обучения модели
* настроить гиперпараметры вручную или фреймворком оптимизации гиперпараметров: Optuna, HyperOpt, Bayesian-Optimization, Scikit-optimize, Spearmint, GPyOpt, RoBo

1. **Построение ансамблей**

* Создание разных моделей
* Определение корреляции прогнозов
* Создание ансамлей с помощью:
  + Бэггинг
  + Арифмитическое среднее
  + Стэкинг
* Создание модели 2-го уровня: blending & stacking