

# Лабораторная работа 4. Решающие деревья, ансамбли

## Задание 1: [Решающее дерево](#)

1. Реализуйте алгоритм построения решающего дерева классификации:
  - Выберите и обоснуйте критерий разбиения.
  - Критерий остановки разбиения вершины на поддеревья.
  - Класс должен иметь стандартный интерфейс: `fit`, `predict`
2. Обучите дерево на [датасете](#)
3. Оцените качество классификации на тестовой выборке. Объясните выбор метрик качества

Обратите внимание, что присутствует признак с пропусками. Модель должна каким-то образом это обрабатывать и не ломаться.

Задание со звёздочкой: расчёт важности признаков для одного дерева

## Задание 2: [Случайный лес](#)

1. На основе реализованного дерева постройте случайный лес:
  - Для каждого дерева выбирайте случайно признаки и объекты для обучения (подробнее об этом можно прочитать по [ссылке](#)).
  - Реализуйте механизм голосования.
  - Добавьте метод `predict_proba`
2. Проведите обучение и тестирование случайного леса.
3. Сравните качество классификации с одиночным деревом.
4. Постройте график зависимости точности от количества деревьев в ансамбле.
5. Визуализируйте уверенность модели в ответах (отдельно на неверно классифицированных объектах). Используйте `predict_proba`

Задание на исследовать мир: постройте случайный лес из решающих пней, сравните с лесом из глубоких деревьев

## Задание 4: [AdaBoost](#)

1. То же самое: реализовать, оценить качество, сравнить с другими моделями, построить график зависимости точности от количества деревьев, объяснить график

## Задание 5: Сравнение с библиотечными реализациями

1. Используйте `scikit-learn` для обучения `DecisionTreeClassifier`, `RandomForestClassifier` и `AdaBoostClassifier`
2. Сравните результаты с вашей реализацией:
  - Точность
  - Скорость обучения
  - Интерпретируемость
  - Поддерживаемые фичи