ТЗ pet-проекта Dota2

## Описание:

В этом прона основе всех характеристик игры, доступных на определённый момент времени. екте ваша задача — предсказать исход игры Dota 2

В Dota 2 участвуют две команды: Radiant и Dire. Вам нужно оценить вероятность победы команды Radiant. Данные игры представлены в виде числовых и категориальных признаков, журналов событий, временных рядов и т.д. Работа с такими данными и умение выявлять закономерности может оказаться очень полезным для дальнейшего изучения Data Science.

В обычной игре Dota 2 каждая из двух команд — Radiant и Dire — состоит из 5 игроков. Каждый игрок выбирает героя, который играет определённую роль. Dota 2 — командная игра, поэтому состав команды имеет большое значение. Карта игры содержит базы команд (фонтан), 3 линии для каждой стороны, магазины, логово Рошана и другие элементы.

В течение игры игроки улучшают своих героев, покупают предметы, разрушают башни, убивают героев противника, фармят крипов врага и "отрекаются" от своих крипов (не дают их убивать врагу). Цель игры — разрушить фонтан противника, и ничья невозможна.

## Оценка

Для оценки используется метрика ROC-AUC. Результат этой задачи — бинарный: для каждой игры нужно предсказать победу команды Radiant (1) или поражение (0). Поскольку мы оцениваем вероятность исхода, результат будет находиться в интервале [0,1]. Затем это значение сравнивается с определённым порогом для получения бинарного ответа (так, например, работает Логистическая регрессия, Дерево решений, Случайный лес и другие алгоритмы).

ROC-кривая строится на основе значений FPR (доля ложных положительных) и TPR (доля истинных положительных) на разных значениях порога. Базовым ориентиром для ROC-AUC будет случайное угадывание, где площадь под кривой равна 0,5. Чем больше площадь под ROC-кривой, тем лучше предсказание.

## Логика решения задачи

1. **Исследовательский анализ данных (EDA)**:
   * Изучение данных: анализ категориальных и числовых признаков, их распределений и корреляций.
   * Обработка пропущенных значений и проверка на выбросы.
   * Визуализация распределений и корреляций между признаками.
2. **Предобработка данных**:
   * Кодирование категориальных признаков (например, с помощью One-Hot Encoding).
   * Масштабирование числовых признаков.
   * Создание новых признаков, которые могут улучшить модель (например, общая стоимость предметов у команды).
3. **Построение и обучение моделей**:
   * Выбор нескольких моделей классификации: Логистическая регрессия, Случайный лес, Градиентный бустинг.
   * Обучение моделей на тренировочных данных.
   * Настройка гиперпараметров моделей (например, с использованием GridSearch или RandomizedSearch).
4. **Оценка качества моделей**:
   * Использование ROC-AUC для оценки качества моделей на валидационном наборе данных.
   * Анализ важности признаков для понимания, какие факторы влияют на предсказания.
5. **Интерпретация и выводы**:
   * Объяснение полученных результатов студентам.
   * Разбор ошибки модели и возможных путей улучшения.

## Технологии и стек

* **Python**: для обработки данных и построения моделей.
* **Pandas и NumPy**: для анализа данных и их предобработки.
* **Scikit-learn**: для обучения моделей и оценки их качества.
* **Matplotlib и Seaborn**: для визуализации данных и результатов.
* **Jupyter Notebook**: для удобной работы над проектом и презентации результатов студентам.

## Описание файлов

1. train\_features.csv: основной тренировочный набор данных, подготовленный организаторами.
2. test\_features.csv: основной тестовый набор данных, подготовленный организаторами.
3. train\_targets.csv: ответы для игр из тренировочного набора данных (информация о том, выиграла ли команда Radiant).
4. train\_matches.jsonl: необработанные данные для тренировочного набора.
5. test\_matches.jsonl: необработанные данные для тестового набора.

Каждый файл содержит разные типы данных, необходимые для обучения и тестирования моделей. Например, train\_features.csv содержит основные признаки игр для тренировок, а train\_targets.csv содержит целевые метки (результаты игр), которые помогут модели научиться предсказывать исходы.

## Описание данных

Данные представлены в формате JSON и содержат информацию о каждом игроке в игре. Важно понимать, что в Dota 2 команда Radiant обозначается префиксом r, а команда Dire — d. Номер игрока в команде указывается после префикса (например, r1 - первый игрок команды Radiant, d2 - второй игрок команды Dire).

**Статистика игрока**

**Каждый игрок имеет следующие поля, которые описывают его состояние и действия во время игры:**

* hero\_id: идентификатор героя, выбранного игроком.
* K/D/A: количество убийств (Kills), смертей (Deaths) и ассистов (Assists), которые получил игрок.
* lh (last hits): количество крипов противника, убитых игроком.
* denies: количество крипов своей команды, убитых игроком, чтобы противник не получил золото.
* gold: количество золота, имеющееся у игрока.
* xp (experience): количество опыта игрока.
* level: текущий уровень игрока.
* health и max\_health: текущее и максимальное количество здоровья игрока.
* max\_mana: максимальное количество маны.
* x, y: текущие координаты игрока на карте.
* stuns: время оглушений, наложенных игроком.
* creeps\_stacked: количество "стакаемых" крипов, созданных игроком для союзников.
* camps\_stacked: количество "лагерей" крипов, улучшенных для фарма.
* rune\_pickups: количество собранных рун.
* firstblood\_claimed: указывает, участвовал ли игрок в получении "первой крови".
* teamfight\_participation: участие игрока в командных боях.
* towers\_killed: количество уничтоженных башен.
* roshans\_killed: количество убитых Рошанов.
* obs\_placed, sen\_placed: количество установленных наблюдательных и сторожевых вардов.

**Кроме перечисленных полей, также существуют поля с суффиксом \_log, которые содержат журналы различных событий. Например:**

* buyback\_log: информация о том, когда игрок воспользовался "выкупом" (быстрым возвращением в игру после смерти).
* ward\_log: информация о времени и месте установки варда.

**Временные ряды**

Кроме обычных данных, JSON-файлы также содержат временные ряды, которые могут быть полезны для анализа динамики игры:

* gold\_t: значения золота в разные моменты времени.
* xp\_t: значения опыта в разные моменты времени.
* lh\_t: количество убийств крипов в разные моменты времени.
* dn\_t (denied creeps): количество "отреченных" крипов в разные моменты времени.