

Машинное обучение — сессионный проект

1. Описание проекта

Проект реализован в рамках курса «Машинное обучение».

Цель: применить методы линейной и логистической регрессии, провести классификацию, оценить метрики моделей и создать интерактивный интерфейс для демонстрации работы.

Датасет: **Wine Quality (White Wine)**

Источник: [UCI Machine Learning Repository](#)

Описание полей: 11 физических и химических характеристик вина (`fixed acidity`, `volatile acidity`, `citric acid`, ...) + целевая переменная `quality`.

2. Структура проекта

- `notebook.ipynb` — основной ноутбук с кодом и визуализациями.
- `winequality-white.csv` — датасет.
- `requirements.txt` — зависимости.
- `README.md` — инструкция по проекту.

В ноутбуке реализованы задачи:

1. Линейная регрессия (с нуля)

2. Градиентный спуск (batch) для обучения.
3. Вывод MSE, график потерь по эпохам, влияние learning rate.
4. Scatter plot данных с линией регрессии и доверительным интервалом.

5. Логистическая регрессия (с нуля)

6. Реализация сигмоиды, функции лог-потерь.
7. Обучение градиентным спуском (опционально с L2-регуляризацией).

8. Классификация (два подхода)

9. Бинарная классификация: логистическая регрессия + решающее дерево.
10. Метрики: accuracy, precision, recall, F1-score, ROC AUC, confusion matrix.

11. Эксперименты и метрики

12. Изменение learning rate, количества эпох, размера batch.
13. Сравнение качества моделей, анализ влияния признаков и переобучения.

14. Интерактивный интерфейс (widgets)

15. Настройка параметров обучения: `learning rate`, `epochs`, `batch size`.

16. Выбор модели и визуализация графиков потерь и метрик.

3. Инструкция по запуску

1. Скопировать репозиторий на локальный компьютер или открыть в Google Colab:

```
git clone <URL_репозитория>
```

или открыть `.ipynb` напрямую через Colab.

1. Установить зависимости:

```
pip install -r requirements.txt
```

1. Запустить `notebook.ipynb` и выполнить все ячейки последовательно.

4. Зависимости (`requirements.txt`)

```
numpy  
pandas  
matplotlib  
scikit-learn  
ipywidgets
```

Дополнительно можно использовать `seaborn` для визуализации.

5. Примечания

- Все модели линейной и логистической регрессии реализованы **вручную с использованием numpy**.
- Для второго классификатора (Decision Tree) используется `scikit-learn`.
- Интерактивные виджеты реализованы через `ipywidgets` и полностью совместимы с Colab.
- Датасет публичный, указан источник и описание полей.