

Министерство образования Республики Беларусь
Учреждение образования
«Брестский государственный технический университет»
Кафедра ИИТ

Лабораторная работа №2

За шестой семестр

По дисциплине: «Модели решения задач в ИС»

Тема: «Адаптивный шаг обучения в однослоином персептроне (метод наискорейшего спуска)»

Выполнил:
Студент 3 курса
Группы ИИ-26(1)
Пилипук М.А.
Проверил:
Андренко К.В.

Брест 2026

Цель: изучить алгоритм оптимизации градиентного спуска с использованием адаптивного шага обучения. Реализовать модифицированный персепtron, в котором параметр скорости обучения t вычисляется на основе минимизации квадратичной формы ошибки для каждой итерации. Сравнить скорость сходимости с классическим алгоритмом из Лабораторной работы №1. Вариант сохраняется.

Задачи работы:

1. Модифицировать алгоритм последовательного обучения (из Лаб №1) таким образом, чтобы на каждой итерации t значение α вычислялось автоматически на основе текущего входного вектора x .
2. Применить вычисленный $\alpha(t)$ для обновления весов i_j и порогов T_j согласно дельта-правилу.
3. Используя данные своего варианта, провести два эксперимента:
 - Обучение с фиксированным шагом (например, $\alpha=0.1$ или $\alpha=1p$).
 - Обучение с адаптивным шагом по Теореме 2.1 (формула 2.36).
4. Следующие задания — такие же, как ЛР №1.

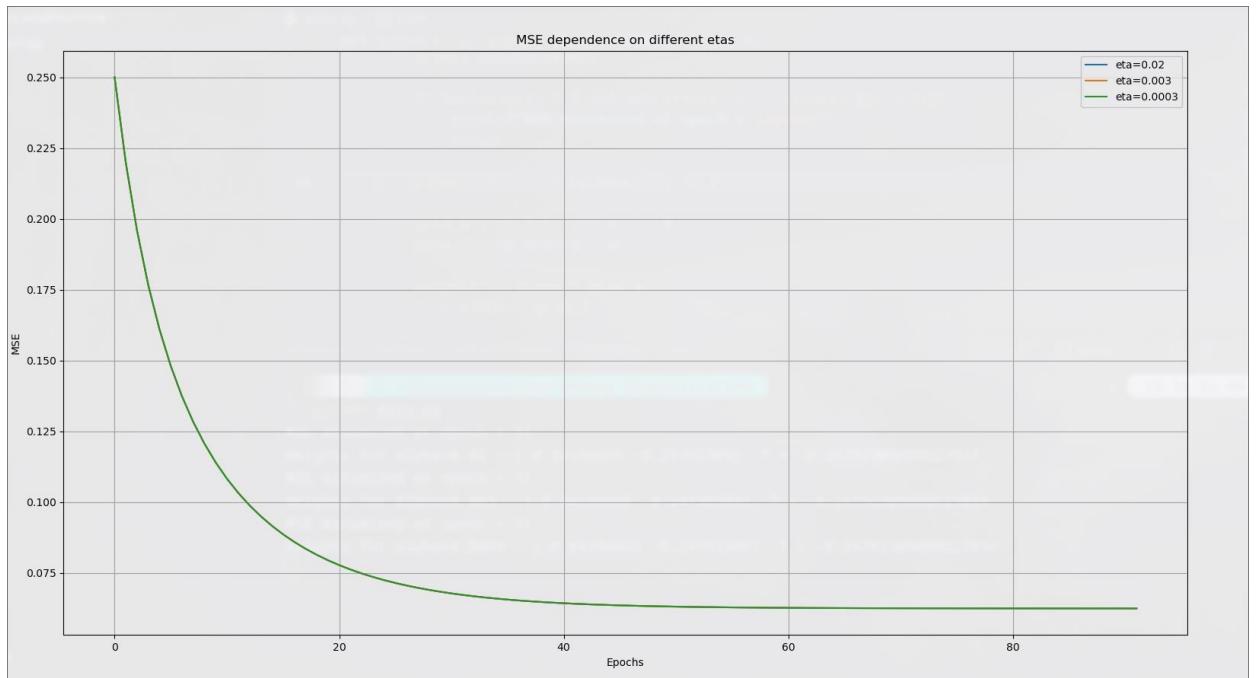
Код программы все также в файле main.py.

Результаты работы:

Адаптивный шаг на каждой итерации обучения вычислялся по следующей формуле:

$$\alpha(t) = 1 / (1 + \text{sum}(x_i^2)).$$

В результате работы функции обучения с данным изменением получаем следующий график:



Как видно по графику, все три шага обучения пришли к одному единственному значению, что графически вызвало наложения кривых друг на друга.

Цифровые значения следующие:

MSE minimized at epoch = 91

Weights for alpha=0.02 : [-0.04166668 -0.24792309], T = -0.24792309008627034

MSE minimized at epoch = 91

Weights for alpha=0.003 : [-0.04166668 -0.24792309], T = -0.24792309008627034

MSE minimized at epoch = 91

Weights for alpha=0.0003 : [-0.04166668 -0.24792309], T = -0.24792309008627034

Если сравнивать с прошлыми результатами:

MSE minimized at epoch = 212

Weights for alpha=0.02 : [-0.04166667 -0.24654961], T = -0.24654961104058393

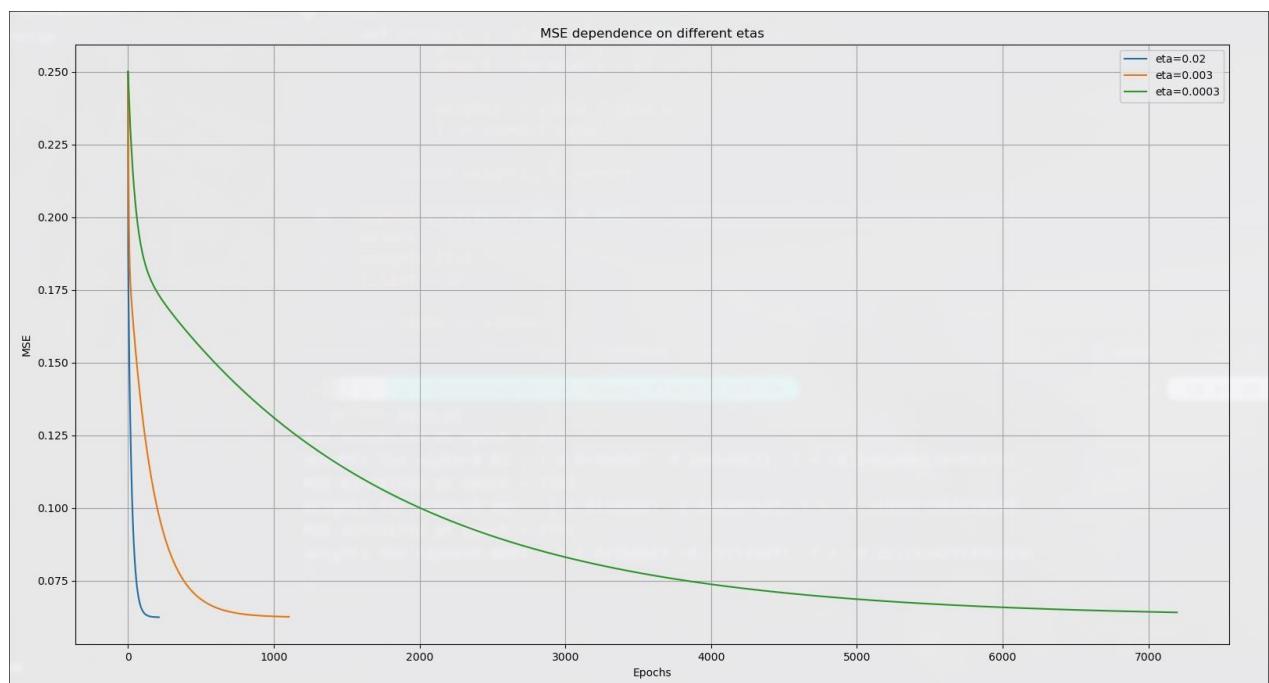
MSE minimized at epoch = 1103

Weights for alpha=0.003 : [-0.04166667 -0.24090716], T = -0.2409071583829589

MSE minimized at epoch = 7196

Weights for alpha=0.0003 : [-0.04166667 -0.22114345], T = -0.22114344914863696

И прошлым графиком:



Невооруженным взглядом можно увидеть колоссальный результат! Уменьшение эпох с 7196 до 91 действительно круто. В результате процесса адаптации, шаг обучения подбирал оптимальное значение вне зависимости от его исходного, что позволило в несколько раз увеличить скорость работы и получить достаточно точные результаты.

Вывод: в ходе данной работы я изучил принцип работы адаптивного шага обучения, что может быть очень полезно при написании обучения для нейронных сетей, так как сильно оптимизирует их работы и ускоряет процесс.

ss