**Министерство образования Российской Федерации**

**МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**им. Н.Э. БАУМАНА**

Факультет: Информатика и системы управления Кафедра: Информационная безопасность (ИУ8)

**ТЕОРИЯ СИСТЕМ И СИСТЕМНЫЙ АНАЛИЗ**

**Лабораторная работа №1 на тему:**

«Исследование методов прямого поиска экстремума унимодальной функции одного переменного»

Вариант 12

**Преподаватель:**

Коннова Н.С.

**Студент**:

Николаева Е.Д.

**Группа:**

ИУ8-31

Москва 2020

# Цель работы

Исследовать функционирование и провести сравнительный анализ различных алгоритмов прямого поиска экстремума (пассивный поиск, метод дихотомии, золотого сечения, Фибоначчи) на примере унимодальной функции одного переменного.

# Постановка задачи

На интервале [*a*,*b*] задана унимодальная функция одного переменного *f* (*x*). Используя методы последовательного поиска (дихотомии, золотого сечения и Фибоначчи), найти интервал нахождения минимума *f* (*x*) при заданной наибольшей допустимой длине интервала неопределенности *e* = 0,1. Провести сравнение с методом оптимального пассивного поиска. Результат, в зависимости от числа точек разбиения N, представить в виде таблицы.

# Ход работы

Получим таблицу истинности для моделируемой БФ:

(x2 + x4)x1 + x1x3

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| x1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| x2 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| x3 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| x4 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| F | **1** | **1** | **1** | **1** | **1** | **1** | **1** | **1** | **0** | **0** | **0** | **0** | **1** | **0** | **0** | **0** |

На начальном шаге l = 0 (эпоха k = 0) весовые коэффициенты берутся в виде:

w(O) = w(O) = w(O) = w(O) = w(O) = 0

O 1 2 3 4

Норма обучения для всех случаев выбирается 5 = 0.3

1. Обучение НС с использованием всех комбинаций переменных

x1, x2, x3, x4.

* 1. Используя пороговую ФА:

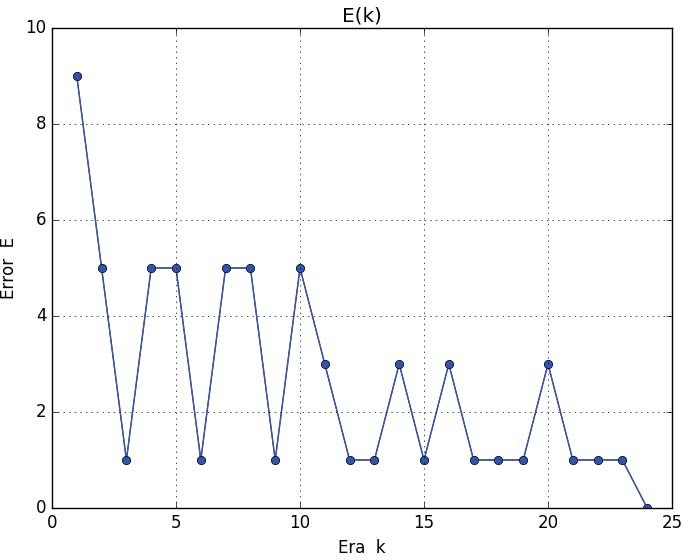
( ) 1, net Σ 0,

f net = {

0, net ≤ 0

*Таблица 1 Параметры НС на последовательных эпохах (пороговая ФА)*

|  |  |
| --- | --- |
| Номер эпохи, k | Вектор весов W, выходной вектор Y, суммарная ошибка E |
| 0 | Y = (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0),  W = (0.0, 0.0, 0.0, 0.0, 0.0), E = 9 |
| 1 | Y = (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0),  W = (0.0, -0.3, 0.0, 0.0, -0.3), E = 9 |
| 2 | Y = (1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0),  W = (0.3, -0.6, 0.0, 0.0, -0.3), E = 5 |
| 3 | Y = (1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0),  W = (0.6, -0.6, 0.0, 0.0, -0.3), E = 1 |
| 4 | Y = (1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0),  W = (0.6, -0.6, 0.0, 0.0, -0.6), E = 5 |
| 5 | Y = (1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0),  W = (0.6, -0.9, 0.0, 0.0, -0.6), E = 5 |
| 6 | Y = (1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0),  W = (0.9, -0.9, 0.0, 0.0, -0.6), E = 1 |
| 7 | Y = (1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0),  W = (0.9, -0.9, 0.0, 0.0, -0.9), E = 5 |
| 8 | Y = (1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0),  W = (0.9, -1.2, 0.0, 0.0, -0.9), E = 5 |
| 9 | Y = (1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0),  W = (1.2, -1.2, 0.0, 0.0, -0.9), E = 1 |
| 10 | Y = (1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0),  W = (1.2, -1.2, 0.0, 0.0, -1.2), E = 5 |
| 11 | Y = (1, 1, 1, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0),  W = (1.2, -1.5, 0.0, -0.3, -0.9), E = 3 |
| 12 | Y = (1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0),  W = (1.5, -1.5, 0.0, 0.0, -0.9), E = 1 |
| 13 | Y = (1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0),  W = (1.5, -1.5, 0.0, 0.0, -1.2), E = 1 |



*Рисунок 1 График суммарной ошибки НС по эпохам обучения (пороговая ФА)*

# Выводы

В ходе проделанной работы было исследовано функционирование и проведен сравнительный анализ алгоритмов прямого поиска экстремума (пассивный поиск, метод Фибоначчи) на примере унимодальной функции одного переменного.

В качестве функции активации бралась две различные функции – пороговая и логистическая. В ходе обучения на полных наборах было выявлено, что с использованием логистической функции активации понадобилось меньше эпох, чем для обучения с использованием пороговой функции активации.

Кроме того, для случаев пороговой и логистической функций активации были найдены минимально возможные наборы векторов, на которых можно обучить НС. В обоих случаях удалось найти наборы, состоящие из четырёх векторов. В случае обучения с использованием пороговой функции активации понадобилось меньшее количество эпох, чем с использованием логистической.

# Приложение А.

*Файл ‘lab-1.py’.*

'''

Лабораторная работа No 1

Исследование однослойных нейронных сетей на примере моделирования булевых выражений.

Цель: Исследовать функционирование простейшей нейронной сети (НС) на базе нейрона с нелинейной функцией активации и ее обучение по правилу Видроу-Хоффа.

Вариант 23. '''

import sys

from AF import \*

from Education import \*

from Tools import boolean\_function, bin\_generation, IntToByte

def initialize\_components(): '''

Функция инциализирует необходимые для расчётов компоненты

:param return: F - значения БФ, W - начальные весовые коэффициенты

'''

W = [0, 0, 0, 0, 0]

n = 4 # число переменных

X = bin\_generation(n) F = get\_F(X)

return F, W

def get\_F(X): '''

Функция возвращает значения БФ на заданных ей наборах переменных

:param X: наборы переменных значения БФ

:param return: значения БФ

'''

F = list() for x in X:

# x0 в расчёт не берётся. Оно необходимо лишь для правила Видроу-Хоффа

F.append(boolean\_function(x[1], x[2], x[3], x[4])) return F

def nnm\_BF(W, F, outputFile): '''

Функция производит расчёт и построения нейросетевой модели БФ, используя пороговую и логистическую ФА

:param W: начальные весовые коэффициенты

:param F: значения БФ

:param outputFile: имя файла вывода

:param return: none '''

# Получим нейросетевую модель БФ, используя пороговую ФА