**Лабораторная работа № 9**

**E12.1** Мы хотим обучить сеть, показанную на рисунке E12.1, на тренировочном наборе

{(p1 = [-2]), (t1 = [0.8])}, {(p2 = [2]), (t2 = [1])},

где каждая пара одинаково вероятна.

Напишите M-файл MATLAB, чтобы создать контурный график для среднеквадратичного показателя эффективности ошибок.

(Смотри рисунок в книге!)

Рисунок E12.1 Сеть для упражнений E12.1

**E12.2.** Продемонстрировать эффект дозирования путем вычисления направления начального шага для SDBP с и без пакетной обработки для задачи, описанной в Упражнении E12.1, начиная с первоначального приближения

w(0) = 0, b(0) = 0.5.

**E12.3** Вспомните квадратичную функцию, используемую в Задаче P9.1:

F(x) = ½ xT[10 -6-6 10]x + [4 4]x.

Мы хотим использовать алгоритм наименьшего спуска с импульсом для минимизации этой функции.

1. Предположим, что скорость обучения α = 0.2. Найдите значение для коэффициента импульса, для которого алгоритм будет устойчивым. Используйте идеи, представленные в Задаче P12.2.
2. Предположим, что скорость обучения α = 20. Найдите значение для коэффициента импульса γ, для которого алгоритм будет устойчивым.
3. Напишите программу MATLAB для построения траекторий алгоритма для значений и частей (i) и части (ii) на графике контура, начиная с первоначального приближения

x0 = [-1 -2.5]T.

**E12.4**. Рассмотрим следующую квадратичную функцию.

F(x) = ½ xT[3 -1-1 3]x + [4 -4]x.

Мы хотим использовать алгоритм наименьшего спуска с импульсом для минимизации этой функции.

1. Выполните две итерации (нахождение x1 и x2) самого крутого спуска с импульсом, начиная с начального условия x0 = [0 0]T. Использовать скорость обучения α = 1 и коэффициент импульса γ = 0.75.
2. Является ли алгоритм устойчивым с этой скоростью обучения и этим импульсом? Используйте идеи, представленные в Задаче P12.2.
3. Будет ли алгоритм устойчивым с этой скоростью обучения, если импульс равен нулю?

**E12.5.** Рассмотрим следующую квадратичную функцию.

F(x) = ½ xT[3 11 3]x + [1 2]x + 2.

Мы хотим использовать алгоритм наименьшего спуска с импульсом для минимизации этой функции.

1. Предположим, что скорость обучения α = 1. Является ли алгоритм устойчивым, если коэффициент импульса γ равен 0? Используйте идеи, представленные в Задаче P12.2.
2. Предположим, что скорость обучения α = 1. Является ли алгоритм устойчивым, если коэффициент импульса γ равен 0.6?

**E12.6**. Рассмотрим следующую квадратичную функцию.

F(x) = ½ xT[2 11 2]x + [1 2]x + 2.

Мы хотим использовать алгоритм наименьшего спуска с импульсом для минимизации этой функции. Предположим, что скорость обучения α = 1. Найдите значение для коэффициента импульса, чтобы алгоритм был устойчивым. Используйте идеи, представленные в Задаче. P12.2.

**E12.7** Для функции Exercise E12.3 выполните три итерации алгоритма переменной скорости обучения, с первоначальным приближением

x0 = [-1 -2.5]T.

Постройте траекторию алгоритма на контурном графике F(x). Использовать параметры алгоритма

α = 0.4, γ = 0.1, η = 1.5, ρ = 0.5, ζ = 5%.

**E12.8**. Рассмотрим следующую квадратичную функцию:

F(x) = x12 + 2x22.

Выполните три итерации алгоритма с переменной скоростью обучения, с первоначальным приближением

x0 = [0 -1]T.

Используйте параметры алгоритма

α = 1, γ = 0.2, η = 1.5, ρ = 0.5, ζ = 5%.

(Считайте итерацию каждый раз, когда функция оценивается после первоначального приближения).

**E12.9.** Для функции Exercise E12.3 выполните одну итерацию алгоритма сопряженного градиента с первоначального приближения

x0 **=** [-1 -2.5]T.

Для определения интервалов использования линейной минимизации путем оценки функции и сокращения интервалов путем поиска Золотого сечения. Выделите путь поиска на контурном участке функции F(x).

**E12.10**. Рассмотрим следующую квадратичную функцию.

F(x) = ½ xT[4 00 2]x + [-2 -1]x.

Мы хотим минимизировать эту функцию вдоль линии

x = [0 0]T + α[1 1]T.

1. Нарисуйте эту строку в плоскости (x1, x2).
2. Скорость обучения α должна опускаться где-то между 0 и 3. Выполните одну итерацию поиска золотого сечения. Вы должны найти a2, b2, c2 и d2, и указать эти точки вдоль линии, на которую вы нарисовали часть i.

**E12.11**. Рассмотрим следующую квадратичную функцию.

F(x) = ½ xT[1 11 1]x + [1 1]x.

Мы хотим минимизировать эту функцию вдоль линии

x = [0 0]T + α[-1 0]T.

1. Используйте метод, описанный на стр. 12-16, чтобы определить начальный интервал, содержащий минимум. Используйте ε = 0.5.
2. Возьмите одну итерацию поиска золотого сечения, чтобы уменьшить интервал, полученный в части i.

**E12.12**. Рассмотрим следующую квадратичную функцию.

F(x) = ½ xT[1 00 2]x.

Мы хотим минимизировать эту функцию вдоль линии

x = [1 1]T + α[1 -1]T.

Выполните две итерации поиска Золотого Сечения (k = 1, 2), чтобы найти интервал [a1, b1]. Предположим, что начальный интервал определен с a1 = 0 и b1 = 1. Сделайте приблизительный эскиз контурного графика F(x), нарисуйте линию поиска на том же рисунке и укажите точки поиска (точки, где вы оценили F(x)) на указанной линии.

**E12.13**. Рассмотрим следующую квадратичную функцию.

F(x) = ½ xT[2 00 1]x.

Мы хотим минимизировать эту функцию вдоль линии

x = [0 1]T + α[1 -1]T.

Выполните две итерации поиска Золотого Сечения (k = 1, 2), чтобы найти интервал [a3, b3]. Предположим, что начальный интервал определен с a1 = 0 и b1 = 1. Сделайте приблизительный эскиз контурного графика F(x), нарисуйте линию поиска на том же рисунке и укажите точки поиска (точки, где вы оценили F(x)) на указанной линии.

**E12.14** Мы хотим использовать сеть, показанную на рисунке E12.2, для приближения функции

g(p) = 1 + sin(π/4p), -2 <= p <=2.

Первоначальные параметры сети выбираются так, чтобы

w1(0) = [-0.27 -0.41]T, b1(0) = [-0.48 -0.13]T, w2(0) = [0.09 -0.17], b2(0) = [0.48].

Чтобы создать обучающий набор, мы проецируем функцию g(p) в точках p = 1 и p = 0. Найти матрицу Якоби для первого шага алгоритма LMBP. (Некоторая информация, которая вам понадобится, была рассчитана в примере, начиная со страницы 11-14.)

(Смотри рисунок в книге!)

Рисунок E12.2 Сеть для упражнений E12.14

**E12.15**. Покажите, что для линейной сети алгоритм LMBP сходится к оптимальному решению на одной итерации, если μ = 0.

**E12.16.** В упражнении E11.25 вы написали программу MATLAB для реализации алгоритма SDBP для сети 1 – S1 - 1 и обучили сеть аппроксимировать функцию

g(p) = 1 + sin(π/4p), -2 <= p <=2.

Повторите это упражнение, изменив свою программу, чтобы использовать обучающие процедуры, обсуждаемые в этой главе: пакетный режим SDBP, MOBP, VLBP, CGBP и LMBP. Сравните результаты конвергенции различных методов.