МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский университет ИТМО»
(Университет ИТМО)

Лабораторная работа 1
по дисциплине
«Математический анализ»

Выполнили:
Ежов Дмитрий Александрович
гр. J3113
ИСУ 471242,
Трифонов Василий Максимович
гр. J3113
ИСУ 467758,
Соловьев Матвей Михайлович
гр. J3113
ИСУ 467551,

Отчет сдан: 15.01.2025

Санкт-Петербург 2023

Оглавление

1. Задание 1

1.1. Теорема 1

Пусть функция $f(x)$ кусочно-непрерывна на отрезке $[a, b]$ и унимодальна на каждом отрезке, где она непрерывна. Тогда функция достигает минимума в точке $x_0 in [a, b]$. Пусть точки $c < d$ принадлежат $[a, b]$. Если $f(c) l = f(d)$, то $x_0 in [a, d]$, иначе $x_0 in [c, b]$.

1.1.1. Доказательство

1. **Разбор условий задачи**
   - Унимодальность $f(x)$ означает, что существует точка или интервал $[\alpha, \beta]$, такие что:
     1. $f(x)$ строго убывает на $[a, \alpha]$ (если $a < \alpha$);
     2. $f(x)$ строго возрастает на $[\beta, b]$ (если $\beta < b$);
     3. $f(x)$ достигает минимального значения на интервале $[\alpha, \beta]$.
   - Для строго унимодальной функции $\alpha = \beta$, и это точка глобального минимума x_0.
   - Кусочная непрерывность $f(x)$ на $[a, b]$ означает, что:
     1. $f(x)$ непрерывна на всех своих подотрезках;
     2. Минимум на любом подотрезке достигается (по теореме Вейерштрасса).
2. **Сужение интервала**
   Рассмотрим точки $c, d \in [a, b]$, такие что $c < d$. Покажем, как определить интервал, содержащий точку минимума x_0, в зависимости от значений $f(c)$ и $f(d)$.
   - **Случай 1:** $f(c) \leq f(d)$.
     1. Минимум x_0 не может находиться правее d, так как $f(x)$ возрастает на $[d, b]$ из унимодальности.
     2. Следовательно, $x_0 in [a, d]$.
   - **Случай 2:** $f(c) > f(d)$.
     1. Минимум x_0 не может находиться левее c, так как $f(x)$ убывает на $[a, c]$ из унимодальности.
     2. Следовательно, $x_0 \in [c, b]$.

   Таким образом, исходный интервал $[a, b]$ можно сократить в зависимости от соотношения $f(c)$ и $f(d)$.
3. **Достижимость минимума**
   1. Кусочная непрерывность $f(x)$ гарантирует, что минимум на каждом непрерывном подотрезке достигается (теорема Вейерштрасса).
   2. Унимодальность обеспечивает, что глобальный минимум на $[a, b]$ совпадает с минимумом на интервале $[\alpha, \beta]$, где $f(x)$ убывает слева от α и возрастает справа от β.
4. **Заключение**
   - При сравнении $f(c)$ и $f(d)$:
     1. Если $f(c) \leq f(d)$, то $x_0 \in [a, d]$.
     2. Если $f(c) > f(d)$, то $x_0 \in [c, b]$.
   - Повторное применение этого правила позволяет сужать интервал неопределённости до любой заданной точности.

Таким образом, теорема доказана.

1.2. Аналитический этап

1. **Выбор функции** Выберем функцию средней сложности, например: $f(x) = x^3 - 3x^2 + 4$.
2. **Определение унимодальности** Установим интервал, на котором функция унимодальна. Например, на $[0, 3]$ функция строго убывает, а затем возрастает.
3. **Нахождение локальных экстремумов** Найдем производную $f'(x)$ и решим уравнение $f'(x) = 0$: $f'(x) = 3x^2 - 6x$. Корни: $x = 0$ и $x = 2$.
   Определим тип экстремумов:
   - $x = 0$: локальный максимум (проверяется по второму производному или изменению знака).
   - $x = 2$: локальный минимум.
4. **Определение экстремальных значений** Вычислим значения функции в точках экстремумов и на концах интервала:
   - $f(0) = 4$,
   - $f(2) = 2$,
   - $f(3) = 4$.

   Минимальное значение на интервале: $f(2) = 2$.
5. **Выбор метода оптимизации** Выбираем метод золотого сечения, так как он эффективно сокращает интервал неопределенности.
6. **Доказательство сходимости метода** Метод золотого сечения сходится, так как на каждой итерации длина интервала уменьшается пропорционально числу $\varphi = \dfrac{3-\sqrt{5}}{2}$. Условие завершения достигается, когда длина интервала становится меньше ε.
7. **Оценка числа итераций** Для интервала $[0, 3]$ и заданной точности $\varepsilon = 0.01$, количество итераций определяется формулой: $n = \left\lceil \log_{\varphi}\left(\dfrac{\varepsilon}{b_0 - a_0}\right)\right\rceil$, где $\varphi \approx 0.618$.
8. **Исследование на других интервалах** Рассмотрим работу метода на интервалах, где функция $f(x)$ не является унимодальной. Это важно для проверки устойчивости алгоритма и его поведения в случае, если входные данные нарушают условия задачи.
   Пример 1: Интервал с двумя локальными экстремумами Возьмем функцию $f(x) = x^4 - 6x^2 + 5$ на интервале $[-3, 3]$. На этом интервале функция имеет два локальных минимума ($x \approx -1.73$ и $x \approx 1.73$) и один локальный максимум ($x = 0$).
   1. Применим метод золотого сечения. Метод будет сужать интервал, приближаясь к одному из локальных минимумов, в зависимости от начальных условий. Это связано с тем, что метод оптимизации учитывает только значения $f(y)$ и $f(z)$ на текущем интервале, не имея информации о глобальной структуре функции.
   2. Вывод: Алгоритм будет корректно сходиться к минимуму, если начальный интервал охватывает один из локальных минимумов. Если интервал включает оба минимума, то результат зависит от расположения точек анализа y и z.

   Пример 2: Интервал с отсутствием унимодальности Рассмотрим функцию $f(x) = \sin(x)$ на интервале $[0, 4\pi]$. На этом интервале функция имеет множество локальных экстремумов (минимумы в точках $x = \pi, 3\pi$ и максимумы в точках $x = 2\pi, 4\pi$).

1. При применении метода: Алгоритм будет "зацикливаться" между локальными минимумами и максимумами, так как длина интервала уменьшается, но не гарантирует нахождение глобального минимума.
2. Вывод: Для функций с несколькими экстремумами метод золотого сечения может давать ошибочные результаты или не сходиться к глобальному минимуму. Это подчеркивает важность выбора интервалов, где функция унимодальна.

9. **Дополнительные задания для алгоритма золотого сечения**

- **Доказательство выбор точек образует золотое сечение**

Метод золотого сечения делит интервал неопределённости в пропорции золотого сечения. Золотое сечение определяется как отношение, при котором длина большей части к меньшей равна длине всего отрезка к большей:

$$\varphi = \frac{b-a}{x-a} = \frac{x-a}{b-x}$$

где

$$\varphi = \frac{\sqrt{5}-1}{2} \approx 0.618$$

.

Доказательство:

1. **Определение точек y_k и z_k:**

   Точки на текущем интервале вычисляются как: $y_k = a_k + (1-\varphi) \cdot (b_k - a_k)$, $z_k = a_k + \varphi \cdot (b_k - a_k)$, где a_k и b_k — границы интервала на k-й итерации.

2. **Свойства деления:**

   После вычисления значений $f(y_k)$ и $f(z_k)$ выбирается новый интервал:

   - Если $f(y_k) \leq f(z_k)$, то $\left[a_{\{k+1\}}, b_{\{k+1\}}\right] = [a_k, z_k]$.
   - Если $f(y_k) > f(z_k)$, то $\left[a_{\{k+1\}}, b_{\{k+1\}}\right] = [y_k, b_k]$.

3. **Сохранение пропорции:**

   На каждой итерации новый интервал сокращается в пропорции φ. Формулы для y_k и z_k гарантируют, что интервал остаётся в пропорции золотого сечения.

Алгоритм сохраняет пропорцию золотого сечения, так как точки y_k и z_k вычисляются с использованием числа φ.

- **Погрешность метода золотого сечения**

  - Формула погрешности

  На каждой итерации длина интервала неопределённости уменьшается в φ раз:

  $L_{\{k+1\}} = \varphi \cdot L_k$, где $L_k = b_k - a_k$.
  После n итераций длина интервала: $L_n = \varphi^n \cdot L_0$, где $L_0 = b_0 - a_0$ — начальная длина интервала.
  Погрешность метода определяется как половина длины интервала: $\varepsilon_n = \frac{L_n}{2} = \frac{\varphi^n \cdot L_0}{2}$.

  - Число итераций для заданной точности

Если требуется точность ε, то n определяется из условия: $L_n = 2 \cdot \varepsilon$.
Подставляя $L_n = \varphi^n \cdot L_0$: $\varphi^n \cdot L_0 = 2 \cdot \varepsilon$.
Решая относительно n:

$$n = \frac{\ln\left(\frac{2 \cdot \varepsilon}{L_0}\right)}{\ln(\varphi)}$$

.

Вывод

1. Алгоритм сохраняет пропорцию золотого сечения на каждой итерации благодаря точному выбору точек y_k и z_k.
2. Погрешность уменьшается экспоненциально, и минимальное количество итераций для точности ε можно вычислить как:

$$n = \frac{\ln\left(\frac{2 \cdot \varepsilon}{L_0}\right)}{\ln(\varphi)}.$$

Общий вывод по анализу:

- Алгоритм успешно работает для строго унимодальных функций, где на интервале присутствует только один экстремум.
- Для функций с несколькими экстремумами или отсутствием унимодальности метод может сходиться к локальному, а не глобальному минимуму.
- Чтобы избежать ошибок, необходимо либо ограничить начальный интервал унимодальными участками, либо использовать дополнительные проверки глобальной структуры функции.

1.3. Практический этап

Написан в `solution1.py`

2. Задача 2

2.1. Аналитический этап

1. **Выбор функций** Возьмём две функции, у которых можно доказать существование корней:
   - $f_1(x) = x^2 - 4$.
   - $f_2(x) = \sin(x) - 0.5$.
2. **Нахождение корней**
   - Для первой функции $f_1(x)$:

     Решим уравнение $x^2 - 4 = 0$. Корни: $x = -2$ и $x = 2$.

   - Для второй функции $f_2(x)$:

     Рассмотрим отрезок $[0, 2\pi]$. Уравнение $\sin(x) - 0.5 = 0$ имеет корни:

     $x_1 = \arcsin(0.5) = \dfrac{\pi}{6}$ и $x_2 = \pi - \dfrac{\pi}{6} = \dfrac{5\pi}{6}$.
3. **Доказательство существования корней**
   - Для $f_1(x)$: На отрезке $[-3, 3]$ функция $f_1(x)$ принимает значения разного знака: $f_1(-3) = 5$, $f_1(0) = -4$. По теореме Больцано корень существует.
   - Для $f_2(x)$: На отрезке $[0, 2\pi]$ значения функции $\sin(x) - 0.5$ изменяются от -0.5 до 1. Применяя теорему о промежуточных значениях, доказываем существование корней.

2.2. Практический этап

См подробнее в `solution2.py`

1. **Реализация алгоритма нахождения корня** Используем метод половинного деления:
   - Зададим начальный интервал $[a, b]$ и точность $\varepsilon > 0$.
   - Повторяем до тех пор, пока длина интервала $b - a > \varepsilon$:
     - Находим середину интервала $c = \dfrac{a+b}{2}$.
     - Если $f(c) = 0$ или длина интервала меньше ε, возвращаем c.
     - Иначе обновляем интервал:
       - Если $f(a) \cdot f(c) < 0$, то $b = c$.
       - Иначе $a = c$.
2. **Критерий остановки** Алгоритм завершается, когда длина интервала становится меньше ε.
3. **Изучение числа итераций** Для функции $f_1(x)$ на интервале $[-3, 3]$ при $\varepsilon = 0.01$ потребуется:

$$n = \left[\frac{\log_2(b-a)}{\varepsilon}\right] = \left[\log_2\left(\frac{6}{0.01}\right)\right] \approx 10 \text{ итераций.}$$

   Для функции $f_2(x)$ на интервале $[0, 2pi]$: аналогично вычисляем.

4. **Среднеквадратичное отклонение (СКО)** Реализуем вычисление СКО для результатов, используя формулу:

$$\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - x^2)}$$

.

5. **Модификация программы** Если корень отсутствует, выводим сообщение об ошибке и проверяем границы интервала. Например:

   Если $f(a) \cdot f(b) > 0$, то корень отсутствует на заданном интервале.