

Тема 2. Методы одномерной оптимизации.

Задание 1. Написать (добавить в собственный класс/библиотеку) следующие функции:

1. Поиск экстремума функции одной переменной методом **ЗОЛОТОГО сечения**;

** Возможна самостоятельная декомпозиция задачи на нескольких функций, решающих конкретные подзадачи. В таком случае, для удобства вызова пользователем, необходимо предусмотреть оберточную функцию.*

Формат входных данных:

Обязательные параметры:

а) Функция в аналитическом виде;

б) Границы области оптимизации;

Необязательные параметры:

в) Точность оптимизации по аргументу (по умолчанию: 10^{-5});

г) Максимальное число итераций (по умолчанию: 500);

д) Флаг «вывод промежуточных результатов» (по умолчанию: False) – при установке TRUE выводит полученные значения результатов на каждой итерации);

е) Флаг «запись промежуточных результатов в датасет» (по умолчанию: False) – при установке TRUE записывает номер итерации и полученные значения результатов на каждой итерации в pandas dataset).

** возможно добавление других обязательны или необязательных параметров*

Формат выходных данных:

а) Найденное значение координаты точки экстремума;

б) Значение функции в точке экстремума;

г) Отчет о работе алгоритма (например флаг: 0 – найдено значение с заданной точностью; 1 – достигнуто максимальное количество итераций; 2 – выполнено с ошибкой).

2. Поиск экстремума функции одной переменной методом **парабол**;

** Возможна самостоятельная декомпозиция задачи на нескольких функций, решающих конкретные подзадачи. В таком случае, для удобства вызова пользователем, необходимо предусмотреть оберточную функцию.*

Формат входных данных:

Обязательные параметры:

а) Функция в аналитическом виде;

б) Границы области оптимизации;

Необязательные параметры:

в) Точность оптимизации по аргументу (по умолчанию: 10^{-5});

г) Максимальное число итераций (по умолчанию: 500);

д) Флаг «вывод промежуточных результатов» (по умолчанию: False) – при установке TRUE выводит полученные значения результатов на каждой итерации);

е) Флаг «запись промежуточных результатов в датасет» (по умолчанию: False) – при установке TRUE записывает номер итерации и полученные значения результатов на каждой итерации в pandas dataset).

* возможно добавление других обязательны или необязательных параметров

Формат выходных данных:

а) Найденное значение координаты точки экстремума;

б) Значение функции в точке экстремума;

г) Отчет о работе алгоритма (например флаг: 0 – найдено значение с заданной точностью; 1 – достигнуто максимальное количество итераций; 2 – выполнено с ошибкой).

3. Поиск экстремума функции одной переменной комбинированным методом Брента;

** Возможна самостоятельная декомпозиция задачи на нескольких функций, решающих конкретные подзадачи. В таком случае, для удобства вызова пользователем, необходимо предусмотреть оберточную функцию.*

Формат входных данных:

Обязательные параметры:

а) Функция в аналитическом виде;

б) Границы области оптимизации;

Необязательные параметры:

в) Точность оптимизации по аргументу (по умолчанию: 10^{-5});

г) Максимальное число итераций (по умолчанию: 500);

д) Флаг «вывод промежуточных результатов» (по умолчанию: False) – при установке TRUE выводит полученные значения результатов на каждой итерации);

е) Флаг «запись промежуточных результатов в датасет» (по умолчанию: False) – при установке TRUE записывает номер итерации и полученные значения результатов на каждой итерации в pandas dataset).

* возможно добавление других обязательны или необязательных параметров

Формат выходных данных:

- а) Найденное значение координаты точки экстремума;
- б) Значение функции в точке экстремума;
- г) Отчет о работе алгоритма (например флаг: 0 – найдено значение с заданной точностью; 1 – достигнуто максимальное количество итераций; 2 – выполнено с ошибкой).

4. Алгоритм неточной одномерной минимизации (Алгоритм Бройдена — Флетчера — Гольдфарба — Шанно);

** Возможна самостоятельная декомпозиция задачи на нескольких функций, решающих конкретные подзадачи. В таком случае, для удобства вызова пользователем, необходимо предусмотреть оберточную функцию.*

Формат входных данных:

Обязательные параметры:

- а) Функция в аналитическом виде;
- б) Начальная точка;

Необязательные параметры:

- в) Параметр для первого условия Вольфе (по умолчанию: 10^{-4});
- г) Параметр для второго условия Вольфе (по умолчанию: 0.1);
- д) Максимально возможное значение аргумента функции (по умолчанию 100);
- е) порог выхода по длине интервала поиска (по умолчанию: 10^{-8});
- г) Максимальное число итераций (по умолчанию: 500);
- д) Флаг «вывод промежуточных результатов» (по умолчанию: False) – при установке TRUE выводит полученные значения результатов на каждой итерации);
- е) Флаг «запись промежуточных результатов в датасет» (по умолчанию: False) – при установке TRUE записывает номер итерации и полученные значения результатов на каждой итерации в pandas dataset).

** возможно добавление других обязательны или необязательных параметров*

Формат выходных данных:

- а) Найденное значение координаты точки экстремума;
- б) Значение функции в точке экстремума;
- г) Отчет о работе алгоритма (например флаг: 0 – точка удовлетворяющая условию Вольфе найдена; 1 – точка удовлетворяющая условию Вольфе найдена с заданной точностью; 2 – достигнуто максимальное количество итераций; 3 – Достигнуто ограничение на максимально возможное значение аргумента; 4 – выполнено с ошибкой).

5. Возможно добавление других функций (опционально)

Задание 2. Для тестирования написанных функций а так же прототипирования различных методов проводится следующее приемо-сдаточное тестирование:

** Данные задания оформляются в отдельном пайплайне (в случае подключения собственной библиотеки), или ячейках, располагающихся ниже. Каждый тест пишется в своей ячейке и решает свой класс задач.*

1. Решение задачи на нахождение экстремума одномерной функции.

С клавиатуры вводится функция в аналитическом виде и начальные условия. Результатом работы программы является список точек экстремумов и значения исследуемой функции в данной точке, число итераций и другие параметры, описанные в выходных параметрах функций из п.1. Задание выполняется для каждого из разработанных алгоритмов (1-3).

Протестировать реализованные алгоритмы на следующем наборе задач оптимизации:

- $f(x) = -5x^5 + 4x^4 - 12x^3 + 11x^2 - 2x + 1$ на интервале $[-0.5, 0.5]$;
- $f(x) = \log^2(x - 2) + \log^2(10 - x) - x^{0.2}$ на интервале $[6, 9.9]$;
- $f(x) = -3x \sin 0.75x + \exp(-2x)$ на интервале $[0, 2\pi]$;
- $f(x) = \exp(3x) + 5 \exp(-2x)$ на интервале $[0, 1]$;
- $f(x) = 0.2x \log x + (x - 2.3)^2$ на интервале $[0.5, 2.5]$;

2. Решение задачи на одномерную неточную оптимизацию.

С клавиатуры вводится функция в аналитическом виде и начальные условия. Результатом работы программы является список точек экстремумов и значения исследуемой функции в данной точке, число итераций и другие параметры, описанные в выходных параметрах функций из п.1.

Протестировать реализованный алгоритм на следующем наборе задач оптимизации:

- $\phi(\alpha) = -\frac{\alpha}{\alpha^2 + \beta}$ для $\beta = 2$;
- $\phi(\alpha) = (\alpha + \beta)^5 - 2(\alpha + \beta)^4$ для $\beta = 0.004$;
- $\phi(\alpha) = \phi_0(\alpha) + \frac{2(1-\beta)}{l\pi} \sin\left(\frac{l\pi}{2}\alpha\right)$, где

$$\phi_0(\alpha) = \begin{cases} 1 - \alpha, & \alpha \leq 1 - \beta, \\ \alpha - 1, & \alpha \geq 1 + \beta, \\ \frac{1}{2\beta}(\alpha - 1)^2 + \frac{\beta}{2}, & \alpha \in [1 - \beta, 1 + \beta]. \end{cases}$$

Здесь $\beta = 0.01$, $l = 39$;

- $\phi(\alpha) = \gamma(\beta_1)\sqrt{(1 - \alpha)^2 + \beta_2^2} + \gamma(\beta_2)\sqrt{\alpha^2 + \beta_1^2}$, где $\gamma(\beta) = \sqrt{1 + \beta^2} - \beta$, для следующих трёх пар (β_1, β_2) : $(0.001, 0.001)$, $(0.01, 0.001)$, $(0.001, 0.01)$;

Дополнительные задания (опционально)

3. Визуализировать работу одного из разработанных алгоритмов. Предлагается построить график исходной функции и на нем отобразить способ построения приближенной аппроксимации для быстро сходящейся функции.

Пример визуализации работы метода парабол:

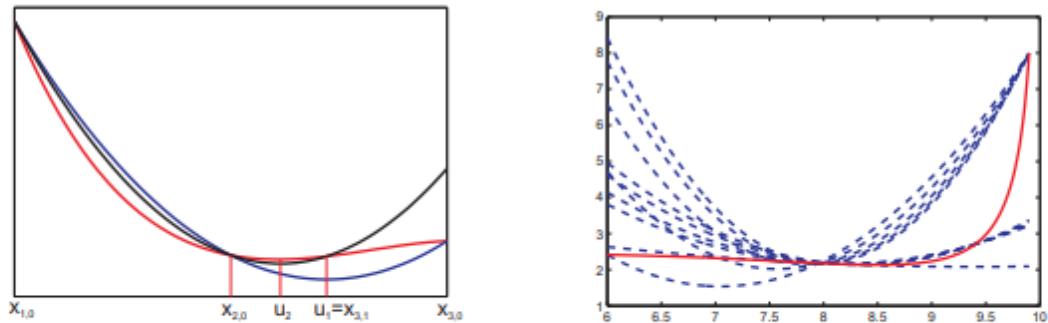
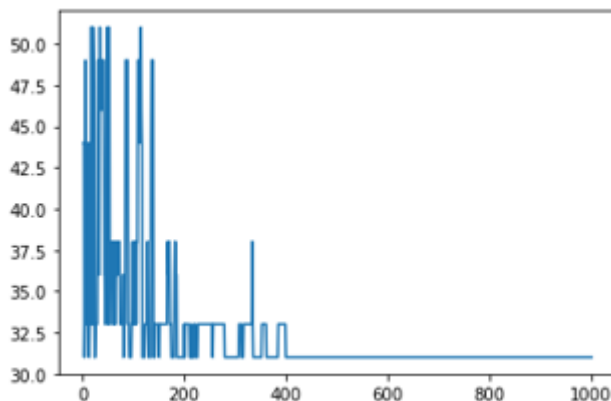


Рис. 2: Иллюстрация работы метода парабол. Слева: первые две итерации метода, красная кривая – оптимизируемая функция, синяя кривая – квадратичное приближение на первой итерации, черная кривая – квадратичное приближение на второй итерации. Справа: пример плохой сходимости метода парабол, красная кривая – оптимизируемая функция, синие кривые – итерационные квадратичные приближения.

** Возможны другие варианты визуализации работы алгоритмов, на усмотрение студентов.*

4. Оценить сходимость алгоритма. Для оценки сходимости метода предлагается построить график величины исследуемого интервала от номера итерации. Так же необходимо вывести списком все полученные величины на каждой итерации алгоритма.

Пример графика:



5. Сравнить производительность 4 разработанных алгоритмов.

Предлагается для одной и той же функции провести поиск минимума всеми 4-я алгоритмами. При этом необходимо замерить время выполнения алгоритма и количество итераций.

Результаты предлагается оформить в виде следующей таблицы:

Параметр	Алгоритм1	Алгоритм2	Алгоритм3	Алгоритм4
Полученное решение				
Время выполнения				
Количество итераций				

6. Сравнить производительность приближенных алгоритмов и точных алгоритмов.

Предлагается для одной и той же функции провести поиск минимума оптимальным алгоритмом из п.5 и одним из алгоритмов 1 темы данного курса. При этом необходимо замерить время выполнения алгоритма и количество итераций.

Результаты предлагается оформить в виде следующей таблицы:

Параметр	Оптимальный итерационный алгоритм	Оптимальный точный алгоритм
Полученное решение		
Время выполнения		

7. Необходимо оформить проектную документацию по проекту в формате файла «Описание структуры и разделов документации по проекту.docx»