Федеральное агентство по образованию

Государственное образовательное учреждение высшего профессионального образования

«Нижегородский государственный архитектурно-строительный университет»

Международный факультет экономики, права и менеджмента

**Расчетно-графическая работа**

**по дисциплине «Исследование операций и методы оптимизации»**

**по теме «Методы нелинейной оптимизации»**

Выполнил студент

Курс III

Группа ПИэ 13.13

Преподаватель

Нижний Новгород

год

Содержание

Постановка задачи

Определение унимодальности функции

Точный метод поиска экстремума

Приближенные методы поиска экстремума

.1 Метод перебора

.2 Метод поразрядного поиска

.3 Метод дихотомии

.4 Метод золотого сечения

.5 Метод средней точки

.6 Метод хорд

.7 Метод Ньютона

Сравнение методов

1 Постановка задачи

Знакомство с оптимизационными задачами, изучение различных методов одномерной оптимизации и сравнение эффективности их применения для конкретных целевых функций.

Нахождение минимума функции 1/|x-3|3 методами перебора, поразрядного поиска, дихотомии, золотого сечения, средней точки, хорд и Ньютона на интервале [2;4] cточностью до e=0,05, а также сравнение методов по скорости вычисления и точности.

2 Определение унимодальности функции

нелинейный оптимизация минимум функция

Функция F(x) является унимодальной на отрезке [A, B] в том и только в том случае, если она монотонна по обе стороны от единственной на рассматриваемом интервале оптимальной точки х\* и принимаем значения f``(x)≥0.

Определим унимодальность заданной функции 1/|x-3|3 двумя способами.

Аналитический способ. Найдем последовательно первую и вторую производные функции:

3/|x-3|4

-12/|x-3|5

Для аналитического определения унимодальности необходимо решить уравнение .

Графический способ

|  |  |  |  |
| --- | --- | --- | --- |
| x | f(x) | f'(x) | f''(x) |
| 2 | 1 | 3 | 12 |
| 2,1 | 1,371742 | 4,572474 | 20,32211 |
| 2,2 | 1,953125 | 7,324219 | 36,62109 |
| 2,3 | 2,915452 | 12,49479 | 71,39882 |
| 2,4 | 4,62963 | 23,14815 | 154,321 |
| 2,5 | 8 | 48 | 384 |
| 2,6 | 15,625 | 117,1875 | 1171,875 |
| 2,7 | 37,03704 | 370,3704 | 4938,272 |
| 2,8 | 125 | 1875 | 37500 |
| 2,9 | 1000 | 30000 | 1200000 |
| 3 | #ДЕЛ/0! | #ДЕЛ/0! | #ДЕЛ/0! |
| 3,1 | 1000 | 30000 | -1200000 |
| 3,2 | 125 | 1875 | -37500 |
| 3,3 | 37,03704 | 370,3704 | -4938,27 |
| 3,4 | 15,625 | 117,1875 | -1171,88 |
| 3,5 | 8 | 48 | -384 |
| 3,6 | 4,62963 | 23,14815 | -154,321 |
| 3,7 | 2,915452 | 12,49479 | -71,3988 |
| 3,8 | 1,953125 | 7,324219 | -36,6211 |
| 3,9 | 1,371742 | 4,572474 | -20,3221 |
| 4 | 1 | 3 | -12 |
| Функция унимодальна на [2;2,9] |





3 Точный метод поиска экстремума

1) Найти производную функции .

) Найти стационарные точки (точки, подозрительные на экстремум), решив уравнение .Обратить внимание на точки, в которых не существует двусторонней конечной производной.

) Выяснить, меняет ли производная свой знак в точках, подозрительных на экстремум. Если она меняет знак с минуса на плюс, то в этой точке функция имеет свой минимум. Если с плюса на минус, то максимум, а если знак производной не меняется, то экстремума в этой точке нет.

) Найти значение функции в точках минимума (максимума).

3/|x-3|4;

/|x-3|4 ≠ 0;

 - глобальный минимум .

4 Приближенные методы поиска экстремума

.1 Метод перебора

Описание:

Метод перебора - простейший из методов поиска значений действительно-значных функций <http://ru.vlab.wikia.com/wiki/%D0%A4%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D1%8F\_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)?veaction=edit&redlink=1> по какому-либо из критериев сравнения (на максимум <http://ru.vlab.wikia.com/wiki/%D0%9C%D0%B0%D0%BA%D1%81%D0%B8%D0%BC%D1%83%D0%BC?veaction=edit&redlink=1>, на минимум <http://ru.vlab.wikia.com/wiki/%D0%9C%D0%B8%D0%BD%D0%B8%D0%BC%D1%83%D0%BC?action=edit&redlink=1>, на определённую константу).

Алгоритм:

)Разобьем отрезок [а, b] на 10 равных частей точками деления:

=a+i\*(b-a)/10, i=2,...2,9

)Вычислив значения F(x) в точках . путем сравнения найдем точку

. где m - это число от 2 до 2,9. такую, что F() = minF() для всех i от 2 до 2,9.

)Погрешность определения точки минимума  функции F(x) методом перебора не превосходит E=(b-a)/n.

Для заданной функции

|  |  |  |  |
| --- | --- | --- | --- |
| x | f(x) | Fmin= | 1 |
| 2 | 1 | x\*= | 2 |
| 2,1 | 1,371742 |  |  |
| 2,2 | 1,953125 |  |  |
| 2,3 | 2,915452 |  |  |
| 2,4 | 4,62963 |  |  |
| 2,5 | 8 |  |  |
| 2,6 | 15,625 |  |  |
| 2,7 | 37,03704 |  |  |
| 2,8 | 125 |  |  |
| 2,9 | 1000 |  |  |

4.2 Метод поразрядного поиска

Описание:

Метод поразрядного поиска. Этот метод представляет собой усовершенствование метода перебора. Поиск точки минимума функции осуществляется с переменным шагом.

Алгоритм:

) Выбрать начальный шаг h=(b-a)/4. Положить х0=а. Вычислить F(x0).

2) Положить =+h. Вычислить F().

) Сравнить F() и F(). Если F()>F(), то перейти к шагу 4, иначе - к шагу 5.

) Положить =и F()=F(). Проверить условие принадлежности хо интервалу [а, b]. Если а <<b, то перейти к шагу 2, иначе - к шагу 5.

) Проверка на окончание поиска: если |h| <= e, то вычисления завершить, полагая х\*=, Fmin=F(), иначе - перейти к шагу 6.

) Изменить направление поиска: положить =, F()=F(), h=-h/4.Перейти к шагу 2.

Для заданной функции:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  | а | b |
|  | E= | 0,05 |  | H= | 0,225 | 2 | 2,9 |
|  |  |  |  |  |  |  |  |
| шаг Н | x0 | x1 | x2 | x3 | х4 | х5 |  |
| 0,225 | 2 | 2,225 | СТОП |  |  |  |  |
|  | 1,000 | 2,148 |  |  |  |  |  |
| 0,05625 | 2,225 | 2,16875 | 2,1125 | 2,05625 | 2 | СТОП |  |
|  | 2,148 | 1,741 | 1,431 | 1,190 | 1,000 |   |   |
| 0,014063 | 2 | 2,014063 | СТОП |  |  |  |  |
|  | 2,014063 | СТОП |  |  |  |  |  |
| Точность достигнута |  |  Fmin= | 1,043402 |  |  |

Блок-схема:



#### 4.3 Метод дихотомии

Описание:

Этот метод является методом прямого поиска. В нем при поиске экстремума целевой функции используются только вычисленные значения целевой функции.

Алгоритм:

Дана функция F(x). Необходимо найти , доставляющий минимум (или максимум) функции F(x) на интервале [a,b] с заданной точностью .

) На каждом шаге процесса поиска делим отрезок [a,b] пополам, x=(a+b)/2 - координата середины отрезка [a,b].

2) Вычисляем значение функции F(x ±вычисленной точки x.

) Сравниваем F1 и F2 и отбрасываем одну из половинок отрезка [a,b].

Если F1<F2, то отрезок [x,b], тогда b=x. Иначе отрезок [a,x], тогда a=x.

) Деление отрезка [a,b] продолжается, пока его длина не станет меньше заданной точности , т.е. 

Блок схема:



Для заданной функции

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Е= | 0,05 | E/2= | 0,025 |  |  |  |  |
| a | b | x1 | x2 | f(x1) | f(x2) | En | Критерий останова |
| 2 | 2,9 | 2,425 | 2,475 | 5,260 | 6,911 | 0,450 | Поиск |
| 2,000 | 2,475 | 2,213 | 2,263 | 0,580 | 0,698 | 0,238 | Поиск |
| 2,000 | 2,263 | 2,106 | 2,156 | 0,345 | 0,453 | 0,131 | Поиск |
| 2,000 | 2,156 | 2,053 | 2,103 | 0,237 | 0,339 | 0,078 | Поиск |
| 2,000 | 2,103 | 2,027 | 2,077 | 0,186 | 0,284 | 0,052 | Поиск |
| 2,000 | 2,077 | 2,013 | 2,063 | 0,160 | 0,258 | 0,038 | Точность достигнута |
| x\* | Fmin |  |  |  |  |  |  |
| 2,038 | 1,124 |  |  |  |  |  |  |

4.4 Метод золотого сечения

Описание:

Пусть задана функция . Тогда для того, чтобы найти определённое значение этой функции на заданном отрезке, отвечающее критерию поиска (пусть это будет минимум), рассматриваемый отрезок делится в пропорции золотого сечения в обоих направлениях.

Алгоритм:

) Задаются начальные границы отрезка [a ; b] и точность .

) Рассчитывают начальные точки:





и значения в них целевой функции <https://ru.wikipedia.org/wiki/%D0%A6%D0%B5%D0%BB%D0%B5%D0%B2%D0%B0%D1%8F\_%D1%84%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D1%8F>:



Если  (для поиска max изменить неравенство на ), то 

Иначе .

)Если , то  и останов. Иначе возврат к шагу 2.

Блок схема:



Для заданной функции

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | E= | 0,05 |   |  |  |  |  |  |  |  |  |
| номер итерации | a | b | x1 | x2 | f(x1) | f(x2) | anew | bnew | En | En<E |
| 1 | 2 | 2,9 | 2,344 | 2,556 | 3,539 | 11,440 | 2,000 | 2,556 | 0,278 | Продолжить поиск |
| 2 |   |   | 2,212 | 2,344 | 2,047 | 3,539 | 2,000 | 2,344 | 0,172 | Продолжить поиск |
| 3 |   |   | 2,131 | 2,212 | 1,526 | 2,047 | 2,000 | 2,212 | 0,106 | Продолжить поиск |
| 4 |   |   | 2,081 | 2,131 | 1,289 | 1,526 | 2,000 | 2,131 | 0,066 | Продолжить поиск |
| 5 |   |   | 2,050 | 2,081 | 1,167 | 1,289 | 2,000 | 2,081 | 0,041 | точность достигнута |
| x\* | Fmin |  |  |  |  |  |  |  |  |  |  |
| 2,041 | 1,132 |  |  |  |  |  |  |  |  |  |  |

.5 Метод средней точки

Описание: основан на алгоритме исключения интервалов, на каждой итерации которого рассматривается одна пробная точка R. Если в точке R выполняется неравенство f'(R) < 0, то в следствии унимодальности функции точка оптимума не может лежать левее точки R. Аналогично, если f'(R) > 0, то интервал x>R можно исключить.

Алгоритм:

1) Определить = , на заданном отрезке [a;b].

) Вычислить f '().

) Проверить критерий окончания вычислений. Если кf '() кЈe, ,перейти к шагу 5, иначе - к шагу 4.

) Перейти к новому отрезку локализации [a, b]. Если f '() > 0, то положить b = . Иначе положить a = . Перейти к шагу 2.

) Положить x\* ». Вычислить f(x\*).

Блок-схема:



Для заданной функции

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| e= | 0,05 |  |  |  |
| a | b | x\* | f`(x\*) | Критерий останова |
| 2 | 2,9 | 2,45 | 32,784646 |   |
| 2 | 2,45 | 2,225 | 8,315999 | Продолжить поиск |
| 2 | 2,225 | 2,1125 | 4,835571 | Продолжить поиск |
| 2 | 2,1125 | 2,05625 | 3,781755 | Продолжить поиск |
| 2 | 2,05625 | 2,028125 | 3,362634 | Продолжить поиск |
| 2 | 2,028125 | 2,014063 | 3,174854 | Продолжить поиск |
| 2 | 2,014063 | 2,007031 | 3,085879 | Продолжить поиск |
| 2 | 2,007031 | 2,003516 | 3,042561 | Точность достигнута |
| x\*= | 2,003516 |  |  |  |
| Fmin= | 1,010621 |  |  |  |

4.6 Метод хорд

Описание: Ориентирован на нахождение корня уравнения f'(x) в интервале [a;b], в котором имеются две точки N и P, в которых знаки производных различны. Алгоритм метода хорд позволяет аппроксимировать функцию f'(x) "хордой" и найти точку, в которой секущая графика f'(x) пересекает ось абсцисс.

Алгоритм:

В этом методе кривая f(x) заменяется прямой линией - хордой. В зависимости от знака выражения f(a)\*f //(a) метод хорд имеет два варианта.

Если f(a) f //(a)>0, то x0=b и

.

Если же f(a) f //(a)<0, то x0=a и

.

Окончание итерационного цикла в этом методе происходит по условию: |f(x1)| < ε или .

Блок-схема:



Для заданной функции:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | e= | 0,05 |  |  |  |  |  |
| номер итерации | a | b | f`(a) | f`(b) | x | f`(x) | Критерий останова |
| 1 | 2 | 2,9 | 3 | 30000 | 1,99991 | 2,99892 | Продолжить поиск |
| 2 | 2 | 1,99991 | 3 | 2,99892 | 1,749944 | 1,228579 | точность достигнута |
| x\*= | 1,749944 |  |  |  |  |  |  |
| Fmin= | 0,511931 |  |  |  |  |  |  |

Так как x\*=1,749944 не попадает на отрезок [2;2,9], то приравняем x к самому наиближайшему числу на отрезке.

x=2 , следовательно Fmin = 1.

4.7 Метод Ньютона

Описание: Ориентирован на нахождение корня уравнения f'(x) в интервале [a,b], в котором имеются две точки N и P, в которых знаки производных различны. Работа алгоритма начинается из точки xo, которая представляет начальное приближение корня уравнения f'(x)=0. Далее строится линейная аппроксимация функции f'(x) в точке x1, и точка, в которой аппроксимирующая линейная функция обращается в нуль, принимается в качестве следующего приближения. Если точка xk принята в качестве текущего приближения к оптимальной точке, то линейная функция, аппроксимирующая функцию f'(x) в точке xk, записывается в виде

f'(x,xk) = f'(xk) + f''(xk)(x-xk)

Алгоритм:

) В качестве начального приближения задается любой корень уравнения, который находится одним из прямых методов.

) В точке F(x0) строится касательная к кривой у = F(x) и ищется ее пересечение с осью х. Точка пересечения принимается за новую итерацию. Метод Ньютона самый быстрый способ нахождения корней уравнений.

) Высчитываем по формуле:



) Итерационный процесс проходит до того времени, пока не будет выполнено условиее . где е - заданная точность.

Блок схема:



Для заданной функции:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | e= | 0,05 |  |  |  |
| номер итерации | Xk | |Xk+1-Xk| | f`(x) | f``(x) | Критерий останова |
| 1 | 2,3 | - | 12,49479 | 71,39882 | - |
| 2 | 2,125 | -0,175 | 5,117868 | 23,39597 | точность достигнута |
| x= | 2,125 |  |  |  |  |
| Fmin= | 1,492711 |  |  |  |  |

Сравнение методов

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Метод | Кол-во итераций | Обращение к функции f(x) | x | f(x) |
| Точный метод | 1 | 1 | 2 | 1 |
| Метод перебора | 158 | 158 | 2 | 1 |
| Метод поразрядного поиска | 9 | 9 | 2,014063 | 1,043402 |
| Метод дихотомии | 6 | 12 | 2,038 | 1,124 |
| Метод золотого сечения | 5 | 10 | 2,041 | 1,132 |
| Метод средней точки | 8 | 8 | 2,003516 | 1,010621 |
| Метод Хорд | 2 | 6 | 2 | 1 |
| Метод Ньютона | 2 | 4 | 2,125 | 1,429711 |

Для данной функции самыми точными оказались: Точный метод , метод перебора, метод средней точки и метод поразрядного поиска

Так же метод дихотомии и метод средней точки довольно хорошо себя показали. Метод хорд и метод Ньютона для данной функции оказались самыми неэффективными.