Лекция №2.

Безусловная оптимизация многих переменных. Методы нулевого порядка. Метод Хука-Дживса

В прикладных исследованиях часто встречаются задачи безусловная оптимизация многих переменных, в которых очень сложно вычислять производные целевой функции. Однако решение многих из этих задач позволяет получать значительный эффект или экономический выигрыш. Поэтому методы решения таких задач, для решения которых не используют производные и которые называют методами нулевого порядка, вызывают достаточно большой интерес.

Одним из таких методов является метод Хука-Дживса. Метод Хука-Дживса был разработан в 1961 году и до сих пор используется для решения подобных задач. Этот метод является эвристическим методом, и его сходимость не доказана.

Метод Хука-Дживса включает два основных этапа:

- «исследовательский поиск» вокруг базисной точки,

- «поиск по образцу».

В методе Хука-Дживса сначала задаётся начальная точка , которую называют базисной и начальное приращение . Затем вычисляется значение минимизируемой функции в базисной точке , т.е. вычисляется . После этого последовательно меняется каждая компонента вектора , начиная с . Величина  изменяется в соответствии со следующим соотношением:

.

После этого вычисляется значение функции  в точке , компоненты которой имеют вид . Если величина целевой функции в точке  окажется меньше, чем в базовой точке , т.е. , то выбранный шаг  считается удачным и пробная точка  принимает вид . В противном случае величина  изменяется следующим образом: . Затем вычисляется значение функции  в этой точке , компоненты которой имеют вид , и производится проверка. Если величина целевой функции в точке  окажется меньше, чем в базовой точке , т.е. , то выбранный шаг  считается удачным и пробная точка принимает вид . Если значение функции  в этой точке  окажется равным или большим значению , то пробная точка принимает вид .

После просмотра приведённых направлений для первой компоненты вектора  по описанной выше схеме производится просмотр направлений и формирование пробных точек для остальных компонент вектора .

По такой же схеме производится просмотр направлений и формирование пробных точек и на других итерациях метода.

Например, если значение функции  при просмотре -й компоненты в базисной точке  при  окажется меньше значения этой функции в предыдущей точке, т.е.

,

то -я компонента пробной точки будет равна . В противном случае производится вычисление значения функции  при значении -й компоненты пробной точки равном . Если окажется, что

,

то -я компонента пробной точки будет равна . В противном случае -я координата пробной точки будет равна .

После завершения формирования пробной точки, когда будут просмотрены все  компонент вектора , производится проверка. Если в пробной точке  уменьшения функции  не произошло, т. е. по построению она совпадает с базисной точкой , то из этой точки повторяется исследующий поиск, но с меньшими шагами , где  − постоянная величина больше 1. Если максимальная величина , при которой в пробной точке уменьшения функции  не произошло, оказалась меньше заданной точности , т.е. , то вычисления прекращаются и эта точка считается решением.

Если исследовательский поиск оказался удачным, и значение функции  оказалось строго меньше величины , т.е.

,

то производится поиск по образцу.

Поиск по образцу производится из точки  в направлении вектора . Координаты новой точки  определяются в соответствии с соотношением:

,

где  − величина шага.

Если в точке  значение функции  строго меньше значения , то точка  становится новой базисной точкой . В противном случае величина  может быть уменьшена, пока не будет выполнено условие  или в качестве новой базисной точкой может быть выбрана точка .

Вычислительная схема метода Хука-Дживса по шагам может быть представлена в следующем виде.

ШАГ 1. Положить , выбрать начальную точку , вычислить функцию  в точке , задать точность , по которой будет производиться прекращение вычислений, и вектор приращения компонент . Обычно все приращения компонент  выбираются одинаковыми. Положить . Следует переход к Шагу 2.

ШАГ 2. Определяется пробная точка  и вычисляется функция  в точке , в которой компоненты имеют вид. Следует переход к Шагу 3.

ШАГ 3. Производится проверка. Если окажется, что , то следует переход к Шагу 4. В противном случае следует к Шагу 5.

ШАГ 4. Производится проверка. Если , то следует переход к Шагу 9. В противном случае  полагается равным , и следует переход к Шагу 2.

ШАГ 5. Определяется пробная точка  и вычисляется функция  в точке , в которой компоненты имеют вид. Следует переход к Шагу 6.

ШАГ 6. Производится проверка. Если окажется, что , то следует переход к Шагу 7. В противном случае следует к Шагу 8.

ШАГ 7. Производится проверка. Если , то следует переход к Шагу 9. В противном случае  полагается равным , и следует переход к Шагу 2.

ШАГ 8. Производится проверка. Если , то следует переход к Шагу 10. В противном случае ,  полагается равным , и следует переход к Шагу 2.

ШАГ 9. Производится поиск по образцу из точки  в направлении вектора , поскольку это направление привило к уменьшению значения целевой функции, в соответствии со следующим соотношением:

,

где  ‒ шаг на этапе поиска по образцу. Следует переход к Шагу 11.

ШАГ 10. Производится проверка. Если , то переход к Шагу 12. В противном случае полагаем , где  ̶ постоянное число, большее единицы,  и следует переход к Шагу 2.

ШАГ 11. Вычисляется функция  в точке . Производится проверка. Если функция , то , . Следует переход к Шагу 2. В противном случае , . Следует переход к Шагу 2.

ШАГ 12. Вычисления заканчиваются. В качестве решения принимается .

**Метод покоординатного спуска**

Рассмотрим функцию  двух переменных. На [рис. 1](https://intuit.ru/studies/courses/1020/188/lecture/4931?page=3#image.10.9) представлены линии уровня функции .

Минимального значения эта функция достигает в точке . Напомним, что линией уровня называется кривая в пространстве параметров (в данном случае в плоскости , значение функции на которой – постоянно). Простейшим методом поиска является метод покоординатного спуска. Из точки А мы производим поиск минимума вдоль направления оси х1 и находим точку B, в которой эта функция достигает минимального значения.

Затем, из точки B производится поиск минимума функции  в направлении оси x2.. В результате проведённого поиска получаем точку C. Из точки C производится поиск минимума функции  параллельно оси x1, и определяется точку , и т.д. Таким образом, при определённых условиях может быть определена точка минимума функции  .

Любой из одномерных методов поиска минимума функций может быть использован здесь для поиска вдоль осей. Аналогичным образом эту идею можно применить для функции  переменных.

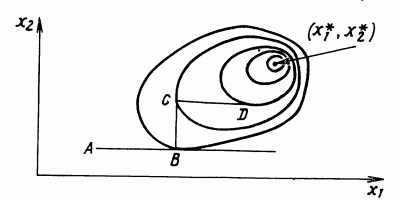


Рис. 2.1. Пример работы метода покоординатного спуска.

Рассмотрим данный метод более детально на примере некоторой целевой функции.

Пусть нужно найти наименьшее значение целевой функции . Выберем произвольную начальную точку  и рассмотрим эту функцию при фиксированных значениях всех переменных, кроме первой: . Тогда она превратится в функцию одной переменной x1. Изменяя эту переменную, будем двигаться от начальной точки  в сторону убывания функции, пока не дойдём до её минимума при , после которого она начинает возрастать. Точку с координатами  обозначим через , при этом .

Зафиксируем теперь переменные:  и рассмотрим данную функцию, как функцию одной переменной .

Тогда путём изменения x2 будем опять двигаться от начального значения  в сторону убывания функции, пока не дойдём до минимума функции  при  и фиксированных значениях .

Точку с координатами  обозначим через . Значение функции в точке  будет удовлетворять условию . Проведём такую же минимизацию целевой функции по переменным . Дойдя до переменной , снова вернёмся к  и продолжим процесс.

Эта процедура вполне оправдывает название метода. С ее помощью мы построим последовательность точек ,. которой соответствует монотонно невозрастающая последовательность значений функции  . Обрывая ее на некотором шаге k, можно приближённо принять значение функции  за ее наименьшее значение в рассматриваемой области ([рис.](https://intuit.ru/studies/courses/1020/188/lecture/4931?page=3#image.10.10) 2.1.).

Отметим, что данный метод сводит задачу поиска наименьшего значения функции нескольких переменных к многократному решению одномерных задач оптимизации.

Если функция  задана явной формулой и является дифференцируемой, то мы можем вычислить ее частные производные и использовать их для определения направления убывания функции по каждой переменной и поиска соответствующих одномерных минимумов.

На [рис.](https://intuit.ru/studies/courses/1020/188/lecture/4931?page=3#image.10.10) 2.2. изображены линии уровня некоторой функции двух переменных . Вдоль этих линий функция сохраняет постоянные значения, равные 1, 3, 5, 7, 9. Показана траектория поиска её наименьшего значения, которое достигается в точке O, с помощью метода покоординатного спуска.

При этом нужно ясно понимать, что рисунок служит только для иллюстрации метода. Когда мы приступаем к решению реальной задачи оптимизации, такого рисунка, содержащего в себе готовый ответ, у нас, конечно, нет.

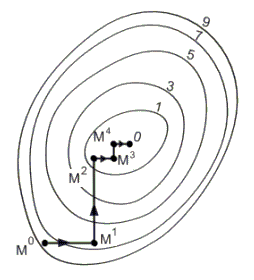


Рис. 2.2. Траектория поиска наименьшего значения функции двух переменных .

При использовании метода покоординатного спуска велика вероятность "застревания" поиска на дне ***оврага***вдали от точки экстремума. На рис. 2.3 видно, что после попадания в точку http://bigor.bmstu.ru/?frm/?doc=120_Opt/5005.mod/?n=8/?k=10, расположенную на дне оврага, дальнейшие шаги возможны лишь в направлениях http://bigor.bmstu.ru/?frm/?doc=120_Opt/5005.mod/?n=9/?k=10 или http://bigor.bmstu.ru/?frm/?doc=120_Opt/5005.mod/?n=10/?k=10, но они приводят к ухудшению целевой функции. Следовательно, поиск прекращается в точке http://bigor.bmstu.ru/?frm/?doc=120_Opt/5005.mod/?n=8/?k=10.

**Примечание 1**

Оврагом называют часть пространства управляемых параметров, в которой наблюдаются слабые изменения производных целевой функции по одним направлениям и значительные изменения с переменой знака — по некоторым другим направлениям. Знак производной меняется в точках, принадлежащих дну оврага.

|  |
| --- |
| http://bigor.bmstu.ru/?img/?doc=120_Opt/5005.mod/?n=2 |

Рис. 2.3. Пример «овражистой» функции

В то же время при благоприятной ориентации дна оврага, а именно при положении одной из координатных осей, близком к параллельности с дном оврага, поиск оказывается весьма быстрым. Эта ситуация показана на рис. 2.4.

|  |
| --- |
| http://bigor.bmstu.ru/?img/?doc=120_Opt/5005.mod/?n=3 |

Рис. 3. Траектория покоординатного спуска при благоприятной ориентации координатных осей

**Метод Розенброка**

**Метод Розенброка** заключается в таком повороте координатных осей, чтобы одна из них оказалась квазипараллельной дну оврага. Такой поворот осуществляют на основе данных, полученных после серии из  шагов покоординатного спуска.

Положение новых осей  может быть получено линейным преобразованием прежних осей .

Направление оси  обычно выбирают так, чтобы оно совпадало с направлением вектора .

Направление остальных осей выбирают из условия ортогональности к оси  и друг к другу.

**Методы поиска экстремума функции одной переменной.**

Рассмотрим кратко некоторые методы поиска экстремума функции одной переменной.

Необходимость рассмотрения численных методов, которые предназначены для поиска экстремумов функций одной переменной, обусловлена следующими обстоятельствами.

Во-первых, эти методы используются во многих алгоритмах поиска экстремума функций многих переменных.

Во-вторых, классы функций одной переменной служат удобной моделью для исследования эффективности методов оптимизации.

В-третьих, иногда удаётся с помощью некоторых приёмов, используя методы одномерной оптимизации получать решения многомерных задач.

Здесь следует отметить, что универсальных методов пригодных для минимизации произвольных функций не существует.

Рассмотрим кратко некоторые методы минимизации выпуклых функций одной переменной.

**Метод дихотомии.** Пусть  ‒ точка минимумафункции  и . Функция является непрерывной, выпуклой функцией на .















Рис. 1.

Вычислим точки  и , , . Величина  является параметром метода и при малых  точки  и  делят отрезок  почти пополам.

Далее вычисляем и сравниваем  и . Если , то полагаем , . Если , то полагаем , .

Если же , то полагаем , .

В результате получим отрезок , содержащий точку , которая является точкой минимума функции  на отрезке .

Причём

.

Если отрезок , содержащий точку  уже известен и

,

то дальше на этом отрезке поступаем аналогичным образом, а, именно, вычисляем точки

,

,

симметрично расположенные на отрезке и вычисляем значения функции  и 

Если , то полагаем , .

Если же , то полагаем , .

Отрезок содержит точку  минимума функции  на отрезке . Длина этого отрезка равна

.

Если сделано  вычислений, то точка  находится внутри отрезка , и в качестве решения для точки минимума функции  примем величину , допустив при этом погрешность

.

**Метод золотого сечения**

Золотым сечением отрезка называется деление отрезка на две неравные части. Деление производится, чтобы отношение длины всего отрезка к длине большей части равнялось отношению длин большей и меньшей частей, т.е.

.

Отсюда получаем  

,

, .

Точка  в свою очередь производит золотое сечение отрезка 

,

.

Аналогично точка  производит золотое сечение отрезка .

В связи с этим, используя данное свойство золотого сечения, можно построить метод поиска точки , в которой достигается минимум функции  на .

Схема метода аналогична схеме рассмотренного ранее метода дихотомии или деления. Сначала на отрезке  определяются две точки  и , которые задают золотое сечение отрезка  и вычисляем значение функции  в этих точках. Если , то  и . Если , то  и .

.

После ‘этого производится золотое сечение отрезка и весь описанный процесс повторяется.

На итерации  точка минимума  функции  на отрезке будет принадлежать отрезку  и

.