Математическое программирование, лекция 8

Метод Нелдера-Мида

```
Продолжение прошлой лекции
```

Достоинства:
1. Простота вычислений
2. Невысокие требования к объему памяти
3. Небольшое число задаваемых параметров α, β, γ, δ
4. Алгоритм оказывается эффективным даже в том случае, когда ошибка вычисления значений функции велика, поскольку при реализации оперирует наибольшими значениями функции в вершинах, а не наименьшими.
5. Достаточная гибкость
6. Учитывает различные ограничения

Недостаток:
1. Медленный

3. Теоретические основы метода сопряженных направлений Пауэлла

Метод ориентирован на решение задач безусловной минимизации с квадратичными функциями и основан на теоретических результатах.

Он использует понятие сопряженных векторов и свойство квадратичной функции, которое называют свойством параллельного подпространства.
Два направления d_i и d_j сопряжены относительно симметричной матрицы C, если $d_i^T C d_j = 0$.
Доказано, что минимум квадратичной функции может быть найден не более чем за n одномерных поисков при условии, что поиск ведется вдоль каждого из n сопряженных направлений.
Сформулируем свойством параллельного подпространства для случая 2-ух переменных.
Пусть заданы квадратичная функция $q(x)$
$q(x) = a + b^T x + 1/2 x^T C x,$
и 2 произвольные несовпадающие точки $x^{(1)}$ и $x^{(2)}$, а также направление d. Если точка $y^{(1)}$ найдена в результате поиска из точки $x^{(1)}$ вдоль каждого из m (m < n) сопряженных направлений, а точка $y^{(2)}$ получена в результате поиска...

Т.о. в 3-ехмерном случае для нахождения точного оптимума квадратичной функции требуется провести 9 поисков вдоль прямой с использованием только значений функции.
Вспомним, что для 2ухмерного случая нам потребовалось 4 поиска
Для n-мерного пространства алгоритм требует проведения последовательности n^2 одномерных поисков, котороя приводит к получению точки оптимума

квадратичной функции при отсутствии ошибок округления.

4. Алгоритм метода Пауэлла (n-мерный случай)

Шаг 1. Задаем $x^{(0)}$, начальную систему n линейно независимых направлений поиска $S^{(0)}$ = ($S_1^{(0)}$, ..., $S_n^{(0)}$)
Обычно за начальные направления принимают направления осей координат $S_1^{(0)} = e^{(1)}$, $S_2^{(0)} = e^{(2)}$ и т.д.;
i=0 (i - кол-во циклов)
Шаг 2. Решаем n+1 задачу минимизации по каждому направлению поиска. Здесь индекс k пробегает значения от 0 до n. При этом полученная ранее точка минимума берется в качестве исходной для следующего поиска. При первом поиске (k=0) минимизируем функцию f(x) по направлению $S_n^{(1)}$, т.е. находим min[f(x^{(0)})]...

Шаг 3. Определяем новое сопряженное направление
$s = (x^{(n+1)} - x^{(1)})$
Шаг 4. Заменяем направления $S_k^{(i+1)}$ = $S_{k+1}^{(i)}$ для всех k=1, 2, ..., n-1 т.е. $S_1^{(1)}$ = $S_2^{(0)}$, $S_2^{(1)}$= $S_3^{(0)}$, и т.д
Отбрасываем $S_1^{(0)}$, заменяя его новым сопряженным с $S_n^{(0)}$ направлением $S_n^{(1)}$ = s
Полученное на последнем поиске значение x станет новым начальным значением на новом цикле поисков $x^{(0)}$ = $x^{(n+1)}$, i = i+1
Если $i < n + 1$, переходим к **шагу 2**, для поиска следующего сопряженного направления
Если $i = n - 1$, переходим к **шагу 5**

Шаг 5. В результате выполнения n-1 цикла все n направлений станут сопряженными. на последнем n-ом цикле осуществляем поиск вдоль последнего найденного направления s , т.е. решаем задачу минимизации min[f(x^{(0)} + αs)], а затем вычисляем x* = $x^{(0)}$, то ...

Тема 3. Безусловная оптимизация функций многих переменных

Лекция 8. Методы поиска оптимума функций многих переменных первого порядка

1. Общая характеристика методов первого порядка
   Градиент функции - некий n-мерный вектор, компонентами которого являются частными производными функции f(x), вычисленными в точке $x^{(0)}$.
   ▼f(x^({0})) = $[(df(x^{(0)}))/dx_1)$ ... $(df(x^{(0)}))/dx_n$]
   Этот вектор перпендикулярен к плоскости, касательной к поверхности уровня функции f(x) проведенной через точку $x^{(0)}$
   Методы первого порядка (градиентные методы)

- метод наискорейшего спуска (метод Коши)
- метод сопряженных градиентов

- квазиньютоновские методы (методы переменной метрики)

2. Метод наискорейшего спуска
При использовании метода наискорейшего спуска на каждой итерации величина шага a_k выбирается из условия минимума функции f(x) в направлении наискорейшего убывания функции, т.е.
$f[x^{(k)} - \alpha_k^* \blacktriangledown f(x^{(k)})]$ = $min[f(x^{(k)} - \alpha_k \blacktriangledown f(x^{(k)}))]$
Это условие означает, что движение вдоль антиградиента происходит до тех пор, пока значение функции f(x) убывает. С математической точки зрения на каждой итерации необходимо решить задачу одномерной оптимизации по a_k функции
$\phi(\alpha_k)$ = $f(x^{(k)} - \alpha_k \blacktriangledown f(x^{(k)})$.
Значение a_k может быть найден с помощью любого из методов одномерного поиска, например, золотого сечения.
Алгоритм:
Шаг 1. Задаем начальную точку x^{(0)} и критерий остановки ε. k = 0
Шаг 2. В точке x^{(k)}, k = 0, 1, ... , вычисляем значение градиента ▼f(x^{(k)})
Шаг 3. Определяем величину шага путем одномерной минимизации по α_k функции $\phi(\alpha_k)$ = $f(x^{(k)} - \alpha_k \blacktriangledown f(x^{(k)}))$
Шаг 4. Определяем координаты точки $x_i^{(k+1)} = x_i^{(k)} - a_k^*(df(x^{(k)})/dx_i)$, i = 1, ... , n;
Шаг 5. Проверяем условие останова.

Достоинства метода: возможность использования самых эффективных методов одномерного поиска.
Недостатки:

3. Медленная скорость сходимости в окрестности минимума из-за малости градиента
4. Плохая эффективность работы в случае овражных функций, т.е. функций , линии уровня которых сильно вытянуты и изогнуты. Направление антиградиента этих функций существенно отклоняется от направления в точку минимума.
   В частности, алгоритм наискорейшего спуска сойдется за одну итерацию при любом начальном приближении для функции $f(x) = x_1^2 + x_2^2$, а в случае функции вида $f(x) = x_1^2 + 100x_2^2$ сходимость будет очень медленной