

Определение максимального и минимального значения функции двух переменных методом градиентного спуска (подъема).

При определении максимального или минимального значений функции двух или более переменных используется понятие «Градиента». Градиентом функции называется вектор, компоненты которого являются частными производными функции по соответствующим координатам. Так, например, для функции двух переменных $f(x,y)$ градиент определяется

как вектор с компонентами: $i \frac{\partial f}{\partial x}$ и $j \frac{\partial f}{\partial y}$. Обозначается градиент функции $\text{grad} f$ либо без значка вектора $\text{grad} f$.

Градиент функции $f(x_1, x_2, \dots, x_n)$ в каждой точке направлен в сторону наискорейшего локального возрастания (убывания) этой функции. Для поиска минимума необходимо спускаться в противоположном направлении.

При численных расчетах градиент функции можно вычислить приближенно $\text{grad} f \approx i \frac{\Delta_x f(x,y)}{\Delta x} + j \frac{\Delta_y f(x,y)}{\Delta y}$, где i, j – единичные вектора по направлениям осей x и y , $\Delta_x f(x,y) = f(x + \Delta x, y)$, $\Delta_y f(x,y) = f(x, y + \Delta y)$, Δx - приращение x , Δy - приращение y .

Вектор градиента $\text{grad} f$ указывает, подобно производной одномерной функции, направление на ближайший локальный максимум функции в данной точке, вектор $-\text{grad} f$ – направление на ближайший локальный минимум.

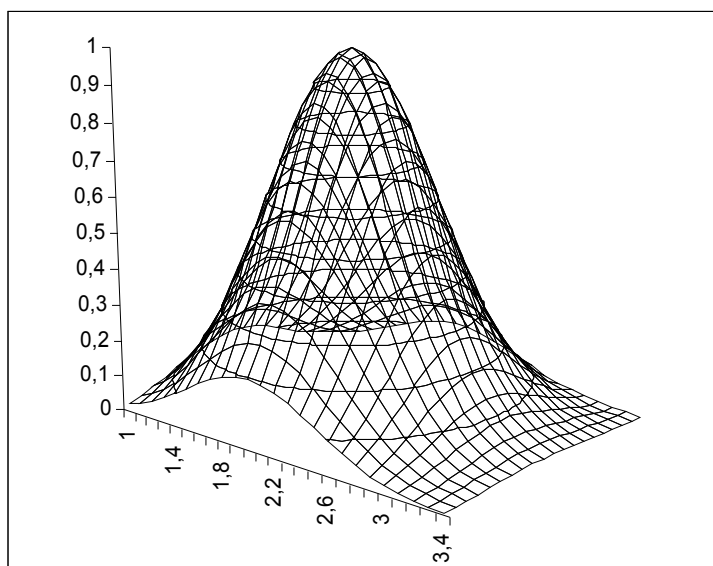
Параметр a определяет длину шага в направлении спуска. Длину шага можно выбирать из условия минимизации функции вдоль направления, противоположного градиенту. Такой метод называют методом наискорейшего спуска. В другом методе градиентного спуска длина шага a выбирается методом дробления.

Рассмотрим многомерную функцию, имеющую вид колокола:

$$f(x_1, x_2) = e^{-2((x_1 - x_{1c})^2 + (x_2 - x_{2c})^2)}, \quad (1)$$

где точка (x_{1c}, x_{2c}) – точка максимума функции.

В области изменений $(1 \leq x_1 \leq 3,4; 1,5 \leq x_2 \leq 3,4)$ функция имеет вид:



Аналогичные функции можно построить в многомерном пространстве. Например, если размерность пространства n , то функция « n -мерный колокол» будет иметь вид:

$$f(x_1, x_2, \dots, x_n) = e^{-2 \sum_{i=1}^n (x_i - x_{iC})^2} \quad (2)$$

Точка максимума этой функции - $(x_{1C}, x_{2C}, \dots, x_{nC})$. Функция « n -мерного колокола» (2) при $n=2$ превращается в (1).

Описание алгоритма нахождения максимального (минимального) значения функции двух переменных методом градиентного спуска (подъема):

1. Задаем размерность n массива аргументов минимизируемой (максимизируемой) многомерной функции, погрешность вычислений ϵ , начальные значения шага d по x , y , координаты начальной точки поиска x_0 , y_0 , определяем вспомогательный массив для запоминания значений аргументов при дроблении шага x_1 , y_1 , массив для составляющих градиента функции df .
2. Вычисляем значение функции в точке с координатами (x_0, y_0) .
3. Вычисляем x и y компоненты вектора градиента функции по формулам $g_x = (f(x_0+dx, y_0) - f(x_0, y_0)) / dx$; $g_y = (f(x_0, y_0+dy) - f(x_0, y_0)) / dy$.
4. Вычисляем модуль градиента функции.
5. Вычисляем новое значение функции и сравниваем с предыдущим. Если максимум (минимум) не найден, продолжаем итерации.
6. Если второе условие не выполнилось, уменьшаем шаг.

Алгоритм на естественном языке приведем для функции « n -мерного колокола».

Строки «комментарии» будем начинать с символа «*».

Для функции с координатами точки максимума: $(2, 3, 4, \dots, n+1)$ для пространства размерности n .

Определить n как целое, где n - количество аргументов функции.

$n=2$

Определить массивы $x_0(n)$, $x_1(n)$, $x_2(n)$, $df(n)$, $h(n)$

* массивы x_0 , x_1 , x_2 - рабочие массивы координат точек (0), (1),

* (2). Массивы df - компоненты вектора градиента функции,

* h - компоненты векторов шага

```

Определить глобальный массив хс(n)
* массив хс-массив координат точки максимума функции

eps=0.01
d=0.01

* в данном цикле задаются координаты точки максимума
цикл по i от 1 до n
хс(i)=1+i
конец цикла

* здесь задаются координаты начальной точки для определения
* максимума
цикл по i от 1 до n
х0(i)=0.1+i
конец цикла

h0=0.5

f0=колокол(@x0,n)

цикл по k от 1 до 10000
печать "k=",x0(1),x0(2),f0,h0

цикл по j от 1 до n
    цикл по i от 1 до n
        x1(i)=x0(i)
    конец цикла
    x1(j)=x0(j)+d

    f1=колокол(@x1,n)
    df(j)=(f1-f0)/d
конец цикла
* нормирование градиента
lendf=0
цикл по i от 1 до n
lendf = lendf + df(i)^2
конец цикла
lendf = Корень(lendf)
цикл по i от 1 до n
df(i)=df(i)/lendf
конец цикла

цикл по i от 1 до n
x0(i)=x0(i)+df(i)*h0
конец цикла

f0_1=колокол(@x0,n)

если f0_1<=f0
печать " нашли max"

если h0<eps

```

```
печать "найден max x0=",
цикл по i от 1 до n
  печать x0(i)
конец цикла
печать ")"
печать " f0= ", f0
печать "за ", k, " итераций"
выход из программы
конец если
```

```
h0=h0/2
```

```
конец если
```

```
f0=f0_1
```

```
конец цикла
```

```
печать "максимум не найден x="
цикл по i от 1 до n
  печать x0(i)
конец цикла
```

```
ОПРЕДЕЛИТЬ ФУНКЦИЮ колокол
параметры x, n
определить массив x(n)
определить локальную переменную r2
r2=0
цикл по i от 1 до n
  r2=r2+(x(i)-xc(i))^2
конец цикла
если r2>10
  r2=10
конец если
вернуть 5*(EXP(-0.1*r2))
```

Пример решения на языке VFP:

```
CLEAR
SET DECIMALS TO 15
n=100
DIMENSION x0(n), x1(n), df(n), h(n), x2(n)
PUBLIC ARRAY xc(n)

FOR i=1 TO n
  xc(i)=1+i
ENDFOR

eps=0.01
d=0.01

FOR i=1 TO n
  x0(i)=1.1+i
```

```

ENDFOR

h0=0.5

f0=функция (@x0,n)

FOR k=1 TO 10000
? "k=",STR(x0(1),5,2),STR(x0(2),5,2),f0,h0

FOR j=1 TO n
    FOR i=1 TO n
        x1(i)=x0(i)
    ENDFOR
    x1(j)=x0(j)+d

    f1=функция (@x1,n)
    df(j)=(f1-f0)/d
ENDFOR
* нормирование градиента
lendf=0
FOR i=1 TO n
lendf = lendf + df(i)^2
ENDFOR
lendf = SQRT(lendf)
FOR i=1 TO n
df(i)=df(i)/lendf
ENDFOR

FOR i=1 TO n
x0(i)=x0(i)+df(i)*h0
ENDFOR

f0_1=функция (@x0,n)

IF f0_1<f0
? " нашли max"

IF h0<eps
? "найден max  x0=(",
FOR i=1 TO n
??x0(i)
ENDFOR
??")"
?" f0= ", f0
? "за ",k," итераций"
RETURN
endif

h0=h0/2

ENDIF

f0=f0_1

```

```

endifor

? "максимум не найден  x="
FOR i=1 TO n
  ??x0(i)
ENDFOR

FUNCTION функция
LPARAMETERS x, n
DIMENSION x(n)
LOCAL r2
r2=0
FOR i=1 TO n
r2=r2+(x(i)-xc(i))^2
ENDFOR
IF r2>10
r2=10
ENDIF
RETURN 5*(EXP(-0.1*r2))

```

Пример решения на языке VBA:

```

Sub grd()
Dim z(8), y(8), p(8)
n = 2
e = 0.0001
a = 0.2
For i = 1 To n
Debug.Print z(i)
z(i) = 0
Next
GoSub 1

For i = 1 To n
Debug.Print z(i)
Next
Debug.Print f, k

1: k = 1
GoSub 2
4: f1 = f
GoSub 3
For i = 1 To n
y(i) = z(i)
Next
5: l = 1
For i = 1 To n
r = a * p(i)
z(i) = y(i) - r
If Abs(r) > e Then
l = 0
End If

```

```
Next
k = k + 1
GoSub 2
If l = 1 Then Return

If f < f1 Then GoTo 4

a = a / 2
GoTo 5
Return
2: r = Exp(z(1) ^ 2 + z(2) ^ 2)
f = r + 2 * z(1) - 3.5 * z(2)
Return
3: r = 2 * r
p(1) = z(1) * r + 2
p(2) = z(2) * r - 3.5
Return
End Sub
```