1 予備知識

2 階偏微分可能であり、2 階偏導関数が連続な（すなわち C^2 級）2 変数関数 $f(x, y)$ の場合は、Hessian

$$H(x, y) = f_{xx}(x, y)\, f_{yy}(x, y) - \left\{f_{xy}(x, y)\right\}^2 \tag{1}$$

の符号によって、極値をとるか否かの判定が出来ます。すなわち、臨界点（停留点）[*1] (a, b) に対して次が成り立ちます。

(1) $H(a, b) > 0$ かつ $f_{xx}(a, b) > 0$ ならば、点 (a, b) で極小値 $f(a, b)$ をとる。
(2) $H(a, b) > 0$ かつ $f_{xx}(a, b) < 0$ ならば、点 (a, b) で極大値 $f(a, b)$ をとる。
(3) $H(a, b) < 0$ ならば、点 (a, b) で極値をとらない。

残念ながら Hessian が 0 になってしまう場合は、この判定法では極値をとるか否か判定できません。

2 バッチ処理

x と y を変数とする関数を引数にとり、極値を返す関数を定義してみたいと思います。

– rmmpt_two.mac –

```
rmmpt_two(f) := block([fx, fy, fxx, fxy, fyy,
    H, s, t, u, Lz: [], Lp: [], Ln : [], realonly: true],
    fx: diff(f, x), fy: diff(f, y),
    fxx: diff(fx, x), fxy: diff(fx, y), fyy: diff(fy, y),
    H: fxx * fyy - fxy^2,
    s: algsys([fx, fy], [x, y]),
    if length(s) = 0 then
        return("Critical Point Not Found")
    else (
        for i: 1 thru length(s) do (
            t: subst(s[i], H),
            if t = 0 then
                Lz: cons(subst(s[i], [x, y, f]), Lz)
            elseif t > 0 then (
                u: subst(s[i], fxx),
                if u > 0 then
                    Lp: cons(subst(s[i], [x, y, f]), Lp)
```

[*1] $f_x(a, b) = f_y(a, b) = 0$ を満たす点

```
18                  else
19                      Ln: cons(subst(s[i], [x, y, f]), Ln)
20              )
21          ),
22          if length(Lz) # 0 then (
23              print("UNKNOWN:"),
24              for i in Lz do disp(i)
25          ),
26          if length(Lp) # 0 then (
27              print("Relative Minimum Point:"),
28              for i in Lp do disp(i)
29          ),
30          if length(Ln) # 0 then (
31              print("Relative Maximum Point:"),
32              for i in Ln do disp(i)
33          )
34      )
35  );
```

2 行目の Lz、Lp、Ln は、それぞれ Hessian が 0 の点、正の点、負の点を格納するリストです[*2]。また、臨界点（停留点）を求める際、求解の範囲を実数に制限するため、変数 `realonly` に `true` を代入しています。6 行目で連立方程式

$$\begin{cases} f_x(x, y) = 0 \\ f_y(x, y) = 0 \end{cases} \qquad [2]$$

を解いています。

7 行目と 8 行目は連立方程式 [2] の解を求められなかった場合の処理です。解が見つかった場合は、各解毎に Hessian を調べます（10 行目の for ループ）。

13 行目は、Hessian が 0 となる（すなわち極値判定不能な）解をリスト Lz に蓄積しています。17 行目は、Hessian が正で、かつ x についての 2 階偏微分係数が正の（すなわち極小となる）点をリスト Lp に蓄積しています。19 行目は、Hessian が正で、かつ x についての 2 階偏微分係数が負の（すなわち極大となる）点をリスト Ln に蓄積しています。

22 行目以降は、結果を出力する処理です。

```
(%i1) load("rmmpt_two.mac");
(%o1)                           rmmpt_two.mac
```

*2 リスト Lz、Lp、Ln の z、p、n は、それぞれ零（Zero)、正（Positive)、負（Negative）の頭文字のつもりです。

```
(%i2) rmmpt_two(x^4 + y^4 - 4*(x^2 - x*y + y^2));
UNKNOWN:
                                    [1, 1, - 2]

                                  [- 1, - 1, - 2]

Relative Minimum Point:
                            [sqrt(3), - sqrt(3), - 18]

                            [- sqrt(3), sqrt(3), - 18]

Relative Maximum Point:
                                     [0, 0, 0]

(%o2)                                   done
```

作成日：平成 20 年 8 月 21 日
ソフトウェア：Maxima 5.15.0cvs & CMU Common Lisp Snapshot 2008-08 (19E)