

SPIRAL OPTIMIZATION ALGORITHM

Tugas Kuliah
SK5001 Analisis Numerik Lanjut

Mohammad Rizka Fadhli
NIM: 20921004

16 October 2021

PENDAHULUAN

Bahasa yang Digunakan

Saya membuat program *spiral optimization algorithm* menggunakan bahasa **R** yang bisa dieksekusi pada versi minimal 3.5.3.

Spiral Optimization Algorithm

Spiral Optimization Algorithm adalah salah satu metode *meta heuristic* yang digunakan untuk mencari minimum global dari suatu sistem persamaan.

Algoritmanya mudah dipahami dan intuitif tanpa harus memiliki latar keilmuan tertentu. Proses kerjanya adalah dengan melakukan *random number generating* pada suatu selang dan melakukan rotasi sekaligus kontraksi dengan titik paling minimum pada setiap iterasi sebagai pusatnya.

Berikut adalah algoritmanya:

INPUT

```
m >= 2 # jumlah titik
theta # sudut rotasi (0 <= theta <= 2pi)
r      # kontraksi
k_max  # iterasi maksimum
```

PROCESS

```
1 generate m buah titik secara acak
  x_i

2 initial condition
  k = 0 # untuk keperluan iterasi

3 cari x_* yang memenuhi
  min(f(x_*))

4 lakukan rotasi dan kontraksi semua x_i
  x_* sebagai pusat rotasi
  k = k + 1

5 ulangi proses 3 dan 4
```

```
6 hentikan proses saat k = k_max
  output x_*
```

Berdasarkan algoritma di atas, salah satu proses yang penting adalah melakukan **rotasi** dan **konstraksi** terhadap semua titik yang telah di-*generate*.

Agar memudahkan, saya akan memberikan ilustrasi geometri beserta operasi matriks aljabar terkait kedua hal tersebut.

Penjelasan Geometri

Operasi Matriks Rotasi

Misalkan saya memiliki titik $x \in \mathbb{R}^2$. Untuk melakukan rotasi sebesar θ , saya bisa menggunakan suatu matriks $A_{2 \times 2}$ berisi fungsi-fungsi trigonometri sebagai berikut:

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix}$$

Berdasarkan operasi matriks di atas, saya membuat **program** di **R** dengan beberapa modifikasi. Sebagai contoh, saya akan membuat program yang bertujuan untuk melakukan rotasi suatu titik $x \in \mathbb{R}$ sebanyak n kali:

```
# mendefinisikan program
rotasi_kan = function(x0,rot){
  # menghitung theta
  theta = 2*pi/rot

  # definisi matriks rotasi
  A = matrix(c(cos(theta),-sin(theta),
               sin(theta),cos(theta)),
             ncol = 2,byrow = T)

  # membuat template
  temp = vector("list")
  temp[[1]] = x0

  # proses rotasi
  for(i in 2:rot){
    xk = A %*% x0
    temp[[i]] = xk
    x0 = xk
  }

  # membuat template data frame
  final = data.frame(x = rep(NA,rot),
                    y = rep(NA,rot))

  # gabung data dari list
  for(i in 1:rot){
    tempura = temp[[i]]
    final$x[i] = tempura[1]
    final$y[i] = tempura[2]
  }

  # membuat plot
```

```

plot =
  ggplot() +
    geom_point(aes(x,y),data = final) +
    geom_point(aes(x[1],y[1]),
              data = final,
              color = "red") +
    coord_equal() +
    labs(title = "titik merah adalah titik initial")

# enrich dengan garis panah
panah = data.frame(
  x_start = final$x[1:(rot-1)],
  x_end = final$x[2:rot],
  y_start = final$y[1:(rot-1)],
  y_end = final$y[2:rot]
)

# menambahkan garis panah ke plot
plot =
  plot +
    geom_segment(aes(x = x_start,
                    xend = x_end,
                    y = y_start,
                    yend = y_end),
                data = panah,
                arrow = arrow(length = unit(.3,"cm")))
)

# menyiapkan output
list("Grafik rotasi" = plot,
     "Titik-titik rotasi" = final)
}

```

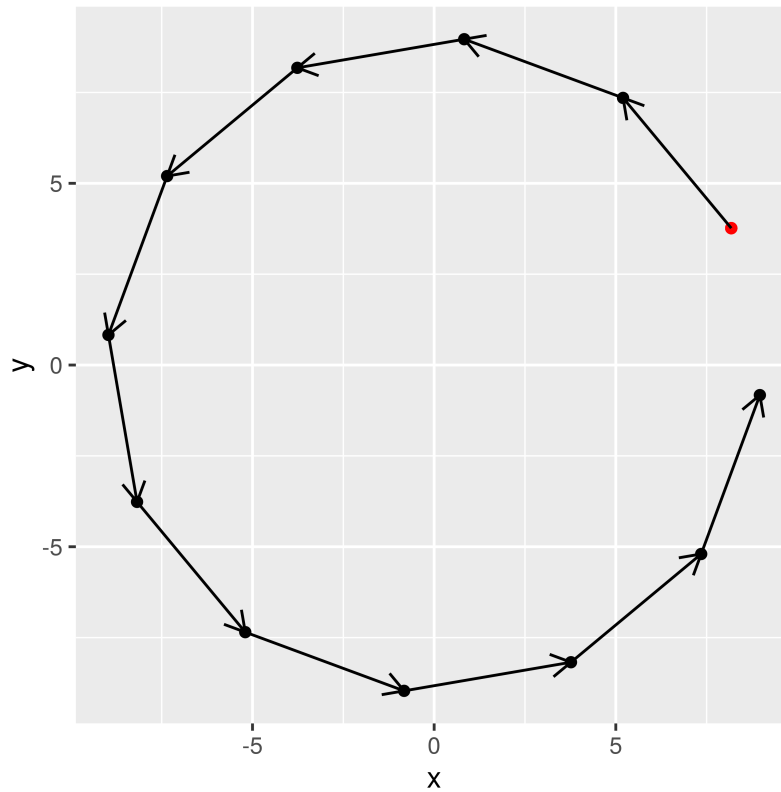
Berikut adalah uji coba dengan titik sembarang berikut ini:

```
# uji coba
rot = 12 # berapa banyak rotasi
x0 = rand_titik(0,10) # generate random titik

rotasi_kan(x0,rot)
```

```
## $`Grafik rotasi`
```

titik merah adalah titik initial



```
##
## $`Titik-titik rotasi`
##      x      y
## 1  8.1780  3.7660
## 2  5.1994  7.3505
## 3  0.8276  8.9654
## 4 -3.7660  8.1780
## 5 -7.3505  5.1994
## 6 -8.9654  0.8276
## 7 -8.1780 -3.7660
## 8 -5.1994 -7.3505
## 9 -0.8276 -8.9654
## 10  3.7660 -8.1780
## 11  7.3505 -5.1994
## 12  8.9654 -0.8276
```

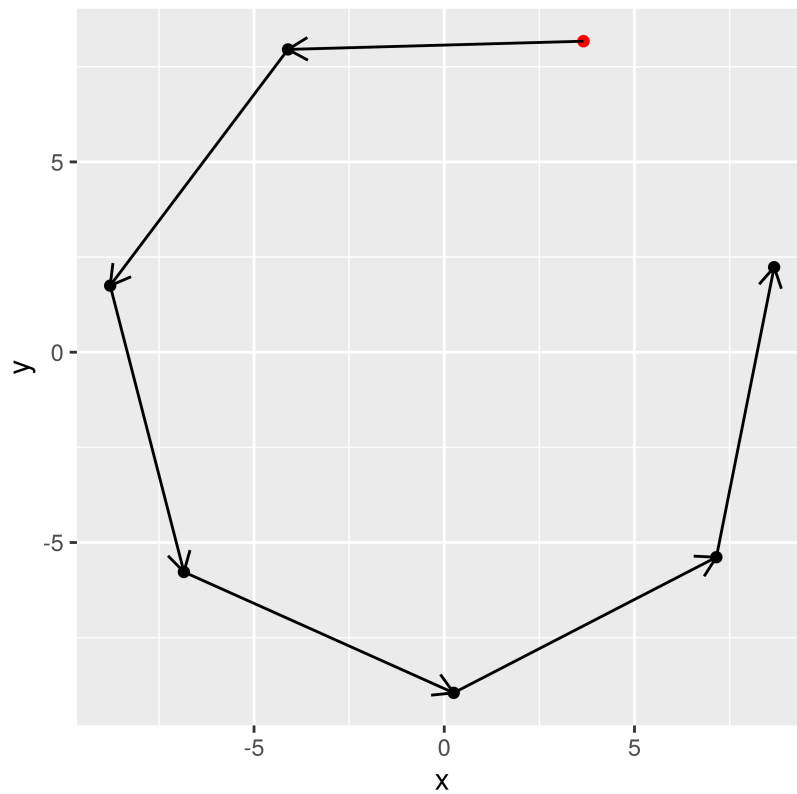
Uji coba kembali dengan titik sembarang lainnya berikut ini:

```
# uji coba
rot = 7 # berapa banyak rotasi
x0 = rand_titik(0,10) # generate random titik

rotasi_kan(x0,rot)
```

```
## $`Grafik rotasi`
```

titik merah adalah titik initial



```
##
## $`Titik-titik rotasi`
##      x      y
## 1  3.6595  8.171
## 2 -4.1069  7.956
## 3 -8.7807  1.749
## 4 -6.8424 -5.774
## 5  0.2483 -8.950
## 6  7.1520 -5.386
## 7  8.6702  2.234
```

Operasi Matriks Rotasi dan Konstraksi

Jika pada sebelumnya saya **hanya melakukan rotasi**, kali ini saya akan memodifikasi operasi matriks agar melakukan rotasi dan konstraksi secara bersamaan. Untuk melakukan hal tersebut, saya akan definisikan $r, 0 < r < 1$ dan melakukan operasi matriks sebagai berikut:

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \end{bmatrix} = \begin{bmatrix} r \\ r \end{bmatrix} \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix}$$

Oleh karena itu saya akan modifikasi program **R** sebelumnya menjadi sebagai berikut:

```
# mendefinisikan program
rotasi_konstraksi_kan = function(x0,rot,r){
  # menghitung theta
  theta = 2*pi/rot

  # definisi matriks rotasi
  A = matrix(c(cos(theta),-sin(theta),
               sin(theta),cos(theta)),
             ncol = 2,byrow = T)

  # membuat template
  temp = vector("list")
  temp[[1]] = x0

  # proses rotasi dan konstraksi
  for(i in 2:rot){
    xk = A %*% x0
    xk = r * xk
    temp[[i]] = xk
    x0 = xk
  }

  # membuat template data frame
  final = data.frame(x = rep(NA,rot),
                    y = rep(NA,rot))

  # gabung data dari list
  for(i in 1:rot){
    tempura = temp[[i]]
    final$x[i] = tempura[1]
    final$y[i] = tempura[2]
  }

  # membuat plot
  plot =
    ggplot() +
    geom_point(aes(x,y),data = final) +
    geom_point(aes(x[1],y[1]),
              data = final,
              color = "red") +
    coord_equal() +
    labs(title = "titik merah adalah titik initial")

  # enrich dengan garis panah
```

```

panah = data.frame(
  x_start = final$x[1:(rot-1)],
  x_end = final$x[2:rot],
  y_start = final$y[1:(rot-1)],
  y_end = final$y[2:rot]
)
# menambahkan garis panah ke plot
plot =
  plot +
  geom_segment(aes(x = x_start,
                   xend = x_end,
                   y = y_start,
                   yend = y_end),
               data = panah,
               arrow = arrow(length = unit(.3,"cm")))
)

# menyiapkan output
list("Grafik rotasi" = plot,
     "Titik-titik rotasi" = final)
}

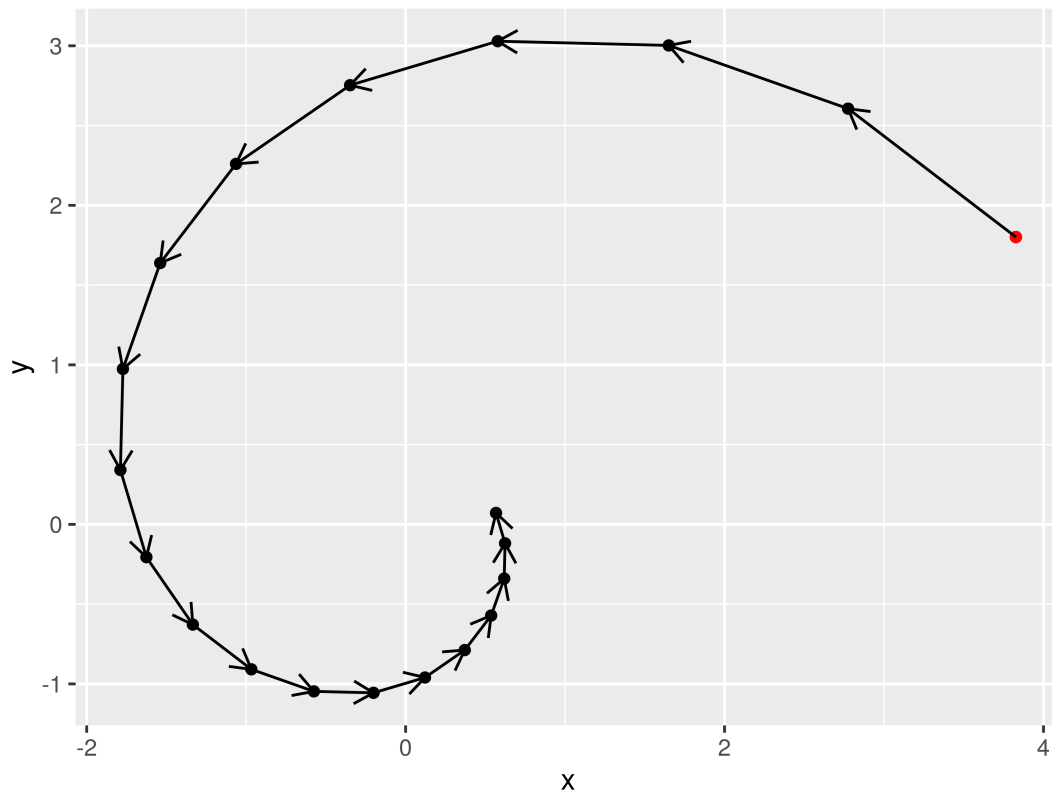
```

Saya akan uji coba untuk sembarang titik berikut ini:

```
# uji coba
rot = 20 # berapa banyak rotasi
x0 = rand_titik(0,4) # generate random titik
r = .9
rotasi_konstraksi_kan(x0,rot,r)
```

```
## $`Grafik rotasi`
```

titik merah adalah titik initial



```
##
## $`Titik-titik rotasi`
##      x      y
## 1  3.8268  1.80122
## 2  2.7746  2.60606
## 3  1.6502  3.00232
## 4  0.5775  3.02878
## 5 -0.3481  2.75309
## 6 -1.0636  2.25971
## 7 -1.5388  1.63840
## 8 -1.7728  0.97441
## 9 -1.7885  0.34099
## 10 -1.6257 -0.20553
## 11 -1.3343 -0.62805
## 12 -0.9675 -0.90868
## 13 -0.5754 -1.04685
## 14 -0.2014 -1.05607
```

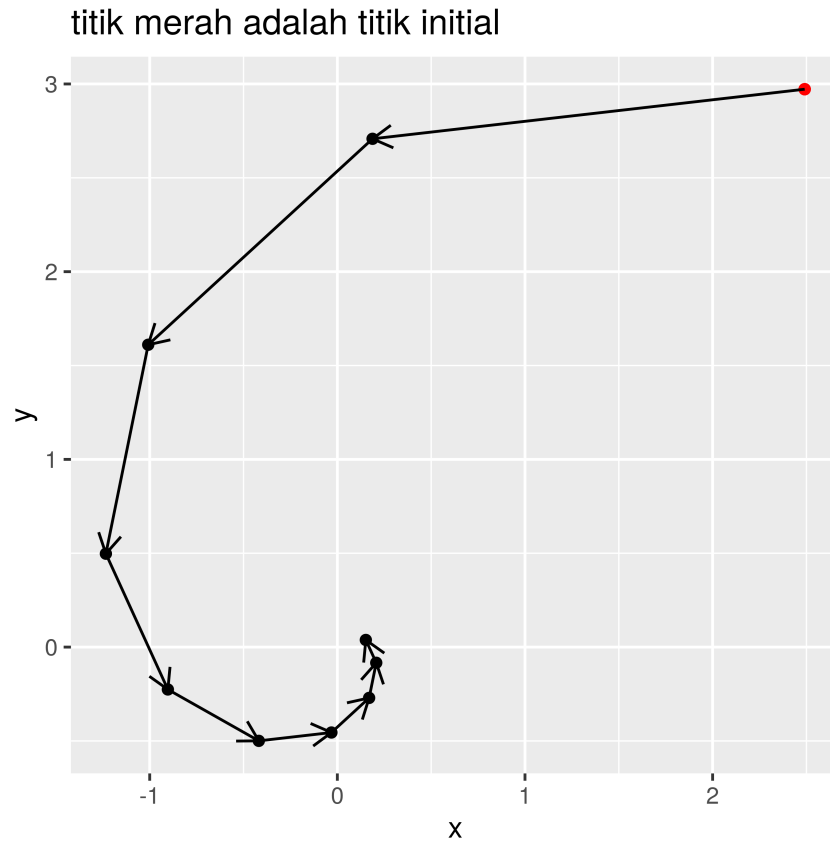


```
## 15  0.1214 -0.95994
## 16  0.3709 -0.78791
## 17  0.5366 -0.57127
## 18  0.6182 -0.33976
## 19  0.6236 -0.11890
## 20  0.5668  0.07166
```

Saya akan uji coba kembali untuk sembarang titik lainnya berikut ini:

```
# uji coba
rot = 10 # berapa banyak rotasi
x0 = rand_titik(0,4) # generate random titik
r = .7
rotasi_konstraksi_kan(x0,rot,r)
```

```
## $`Grafik rotasi`
```



```
##
## $`Titik-titik rotasi`
##      x      y
## 1  2.49062  2.97209
## 2  0.18760  2.70789
## 3 -1.00792  1.61070
## 4 -1.23352  0.49745
## 5 -0.90323 -0.22582
## 6 -0.41860 -0.49952
## 7 -0.03153 -0.45512
## 8  0.16940 -0.27071
## 9  0.20732 -0.08361
## 10 0.15181  0.03795
```

Operasi Matriks Rotasi dan Konstraksi dengan Titik x^* Sebagai Pusatnya

Salah satu prinsip utama dari *spiral optimization algorithm* adalah menjadikan titik x^* sebagai pusat rotasi di setiap iterasinya. Oleh karena itu kita akan modifikasi program bagian sebelumnya menjadi sebagai berikut:

```
# mendefinisikan program
rotasi_konstraksi_pusat_kan = function(x0,rot,r,x_bin){
  # menghitung theta
  theta = 2*pi/rot

  # definisi matriks rotasi
  A = matrix(c(cos(theta),-sin(theta),
               sin(theta),cos(theta)),
             ncol = 2,byrow = T)

  # membuat template
  temp = vector("list")
  temp[[1]] = x0

  # proses rotasi dan konstraksi
  for(i in 2:rot){
    xk = A %*% (x0-x_bin) # diputar dengan x_bin sebagai pusat
    xk = x_bin + (r * xk)
    temp[[i]] = xk
    x0 = xk
  }

  # membuat template data frame
  final = data.frame(x = rep(NA,rot),
                    y = rep(NA,rot))

  # gabung data dari list
  for(i in 1:rot){
    tempura = temp[[i]]
    final$x[i] = tempura[1]
    final$y[i] = tempura[2]
  }

  # membuat plot
  plot =
    ggplot() +
    geom_point(aes(x,y),data = final) +
    geom_point(aes(x[1],y[1]),
              data = final,
              color = "red") +
    geom_point(aes(x = x_bin[1],
                  y = x_bin[1]),
              color = "blue") +
    labs(title = "titik merah adalah titik initial\ntitik biru adalah pusat rotasi")

  # enrich dengan garis panah
  panah = data.frame(
    x_start = final$x[1:(rot-1)],
    x_end = final$x[2:rot],
    y_start = final$y[1:(rot-1)],
```

```

    y_end = final$y[2:rot]
  )
  # menambahkan garis panah ke plot
  plot =
    plot +
      geom_segment(aes(x = x_start,
                        xend = x_end,
                        y = y_start,
                        yend = y_end),
                    data = panah,
                    arrow = arrow(length = unit(.3, "cm")))
  )

  # menyiapkan output
  list("Grafik rotasi" = plot,
       "Titik-titik rotasi" = final)
}

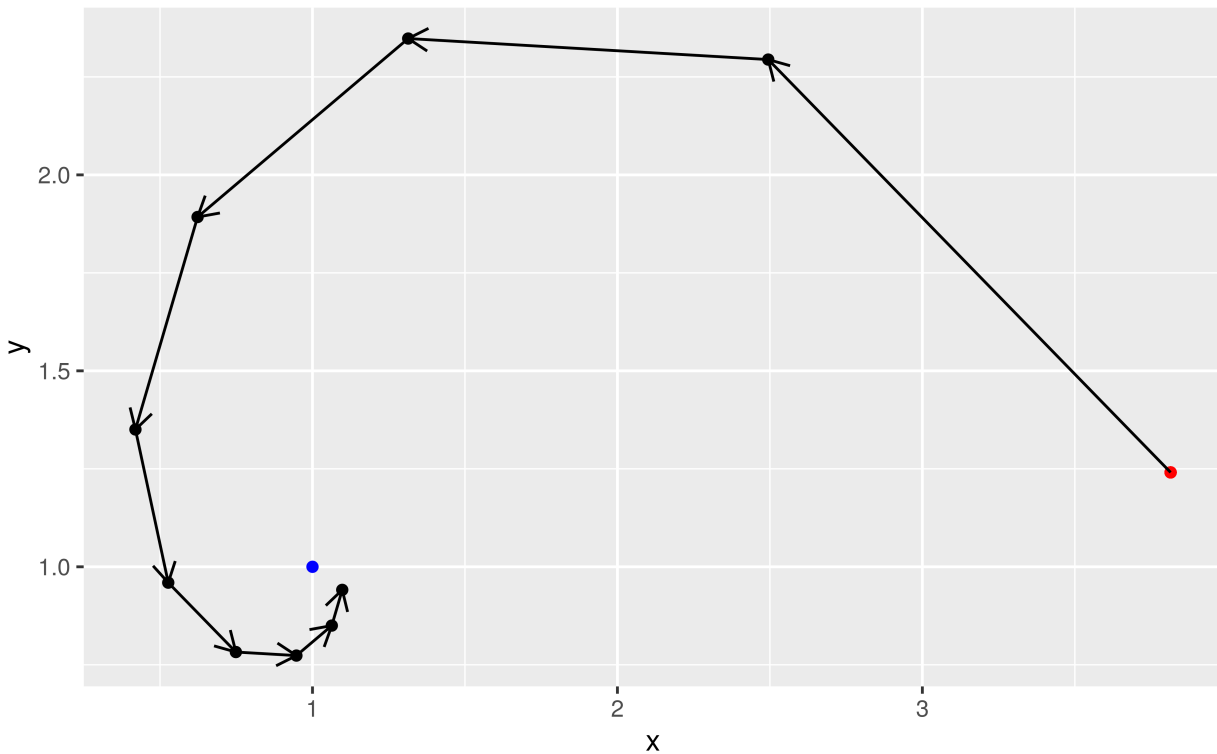
```

Saya akan coba dengan sembarang titik berikut:

```
# uji coba
rot = 10 # berapa banyak rotasi
x0 = rand_titik(0,4) # generate random titik
x_bintang = c(1,1) # contoh pusat rotasi
r = .7
rotasi_konstraksi_pusat_kan(x0,rot,r,x_bintang)
```

```
## $`Grafik rotasi`
```

titik merah adalah titik initial
titik biru adalah pusat rotasi



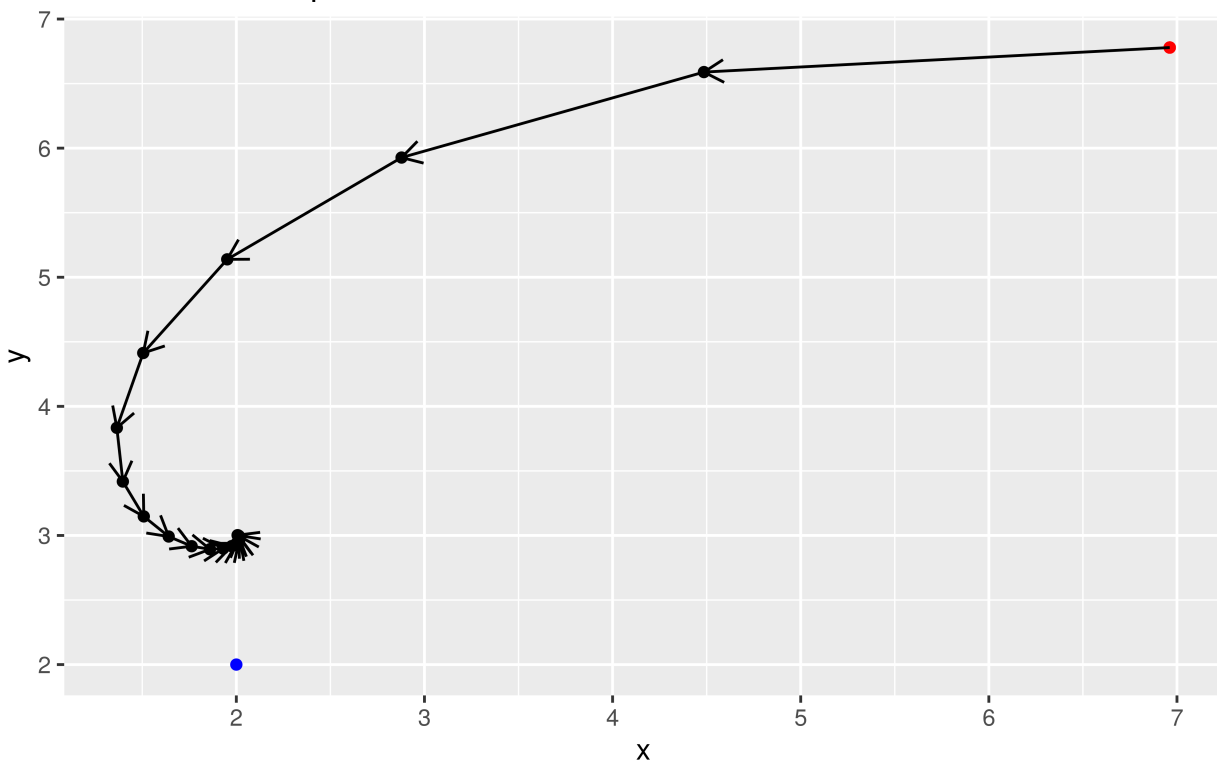
```
##
## $`Titik-titik rotasi`
##      x      y
## 1  3.8141 1.2408
## 2  2.4946 2.2943
## 3  1.3139 2.3479
## 4  0.6232 1.8925
## 5  0.4194 1.3504
## 6  0.5270 0.9595
## 7  0.7488 0.7825
## 8  0.9472 0.7735
## 9  1.0633 0.8500
## 10 1.0976 0.9411
```

Saya akan coba kembali dengan sembarang titik lainnya:

```
# uji coba
rot = 20 # berapa banyak rotasi
x0 = rand_titik(0,10) # generate random titik
x_bintang = c(2,3) # contoh pusat rotasi
r = .7
rotasi_konstraksi_pusat_kan(x0,rot,r,x_bintang)
```

```
## $`Grafik rotasi`
```

titik merah adalah titik initial
titik biru adalah pusat rotasi



```
##
## $`Titik-titik rotasi`
##      x      y
## 1  6.962  6.779
## 2  4.486  6.589
## 3  2.878  5.927
## 4  1.952  5.139
## 5  1.505  4.413
## 6  1.365  3.834
## 7  1.397  3.418
## 8  1.508  3.148
## 9  1.641  2.992
## 10 1.762  2.917
## 11 1.860  2.893
## 12 1.930  2.899
## 13 1.975  2.917
```

```
## 14 2.001 2.940
## 15 2.014 2.960
## 16 2.018 2.976
## 17 2.017 2.988
## 18 2.014 2.996
## 19 2.010 3.000
## 20 2.007 3.002
```

Program *Spiral Optimization Algorithm*

Berbekal program yang telah dituliskan di bagian sebelumnya, kita akan sempurnakan program untuk melakukan *spiral optimization* sebagai berikut:

SOAL 1

Tentukanlah akar-akar sistem persamaan berikut dengan **SOA**. Buatlah terlebih dahulu *contour plot*-nya:

$$f_1(x_1, x_2) = \cos(2x_1) - \cos(2x_2) - 0.4 = 0$$

$$f_2(x_1, x_2) = 2(x_2 - x_1) + \sin(x_2) - \sin(x_1) - 1.2 = 0$$

dengan $-10 \leq x_1, x_2 \leq 10$

JAWAB

Contour Plot

SOAL 2

Tentukanlah akar-akar sistem persamaan berikut dengan **SOA**. Buatlah terlebih dahulu *contour plot*-nya:

$$f_1(x_1, x_2) = \sin(x_1) \cos(x_2) + 2 \cos(x_1) \sin(x_2) = 0$$

$$f_2(x_1, x_2) = \cos(x_1) \sin(x_2) + 2 \sin(x_1) \cos(x_2) = 0$$

dengan $0 \leq x_1, x_2 \leq 2\pi$