

# 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  $\theta$ , 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)
}

```

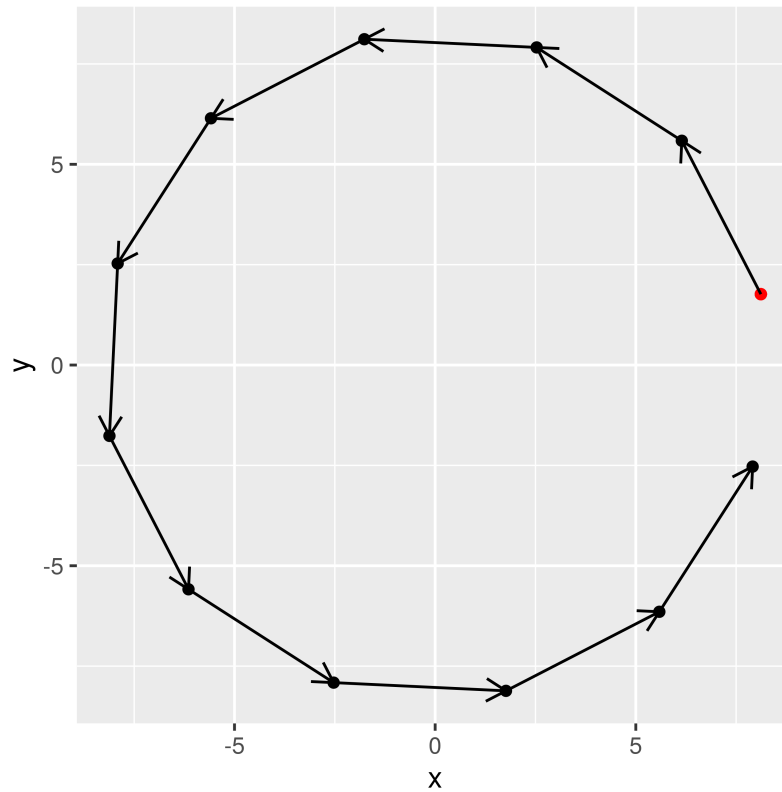
Berikut adalah uji coba dengan titik sembarang berikut ini:

```
# 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  8.116  1.765
## 2  6.146  5.586
## 3  2.530  7.911
## 4 -1.765  8.116
## 5 -5.586  6.146
## 6 -7.911  2.530
## 7 -8.116 -1.765
## 8 -6.146 -5.586
## 9 -2.530 -7.911
## 10  1.765 -8.116
## 11  5.586 -6.146
## 12  7.911 -2.530
```

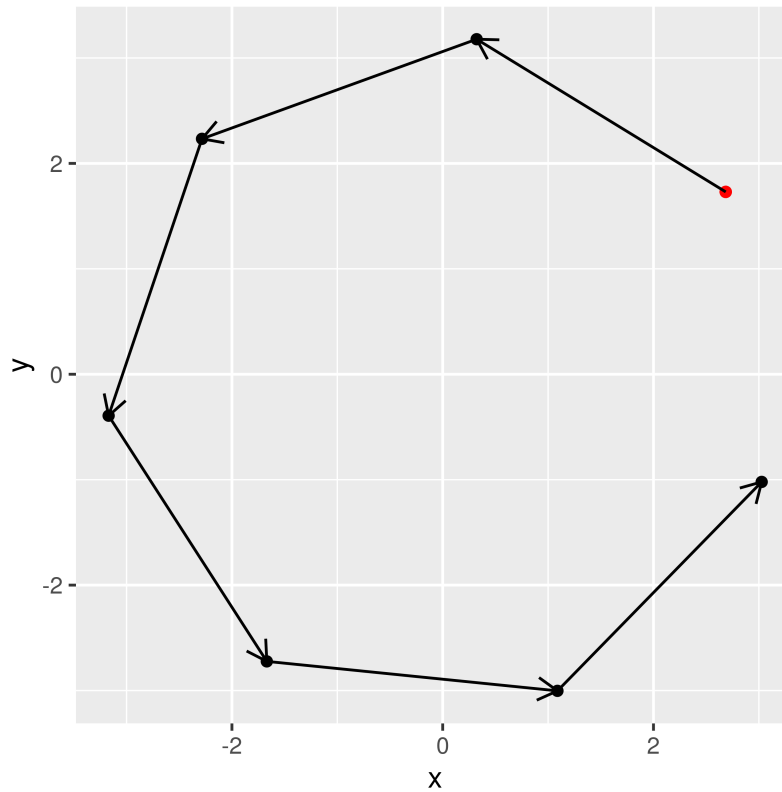
Uji coba kembali dengan titik sembarang lainnya berikut ini:

```
# 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  2.685  1.7297
## 2  0.322  3.1780
## 3 -2.284  2.2332
## 4 -3.170 -0.3932
## 5 -1.669 -2.7236
## 6  1.089 -3.0030
## 7  3.027 -1.0211
```

## 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")))
)

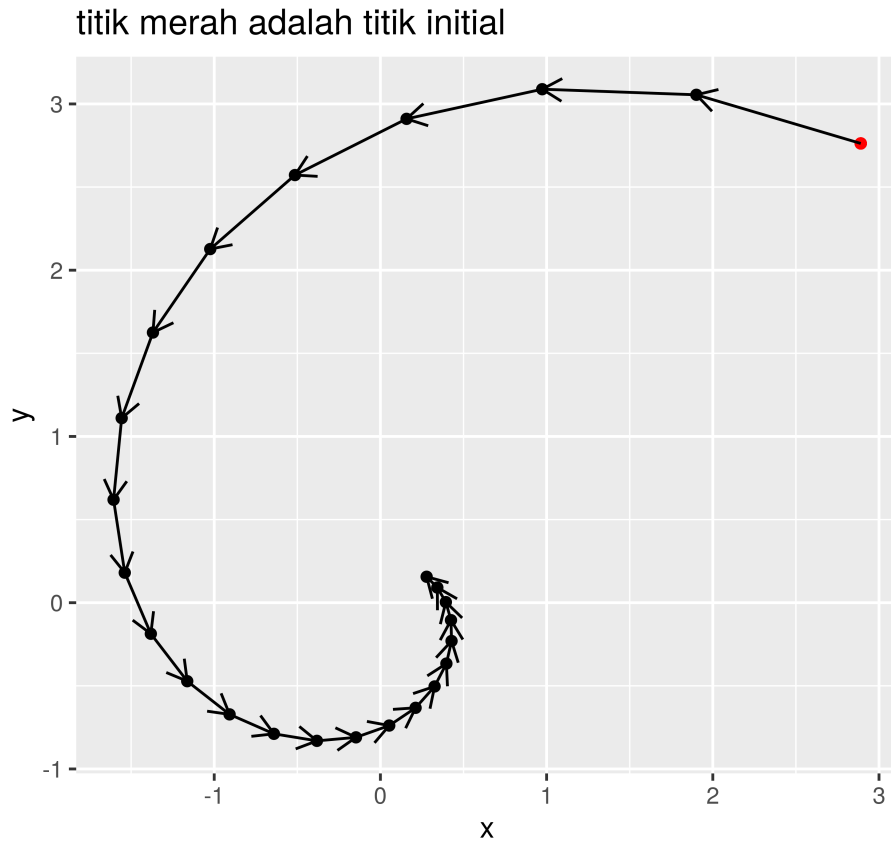
# menyiapkan output
list("Grafik rotasi" = plot,
     "Titik-titik rotasi" = final)
}

```

Saya akan uji coba untuk sembarang titik berikut ini:

```
# uji coba
rot = 25 # berapa banyak rotasi
x0 = rand_titik(0,4) # generate random titik
r = .9
rotasi_konstraksi_kan(x0,rot,r)
```

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



```
##
## $`Titik-titik rotasi`
##      x      y
## 1  2.89014 2.762436
## 2  1.90112 3.054959
## 3  0.97349 3.088594
## 4  0.15732 2.910291
## 5 -0.51424 2.572186
## 6 -1.02399 2.127140
## 7 -1.36873 1.625091
## 8 -1.55689 1.110282
## 9 -1.60568 0.619396
## 10 -1.53835 0.180558
## 11 -1.38143 -0.186917
## 12 -1.16239 -0.472133
## 13 -0.90761 -0.671737
## 14 -0.64084 -0.788712
```

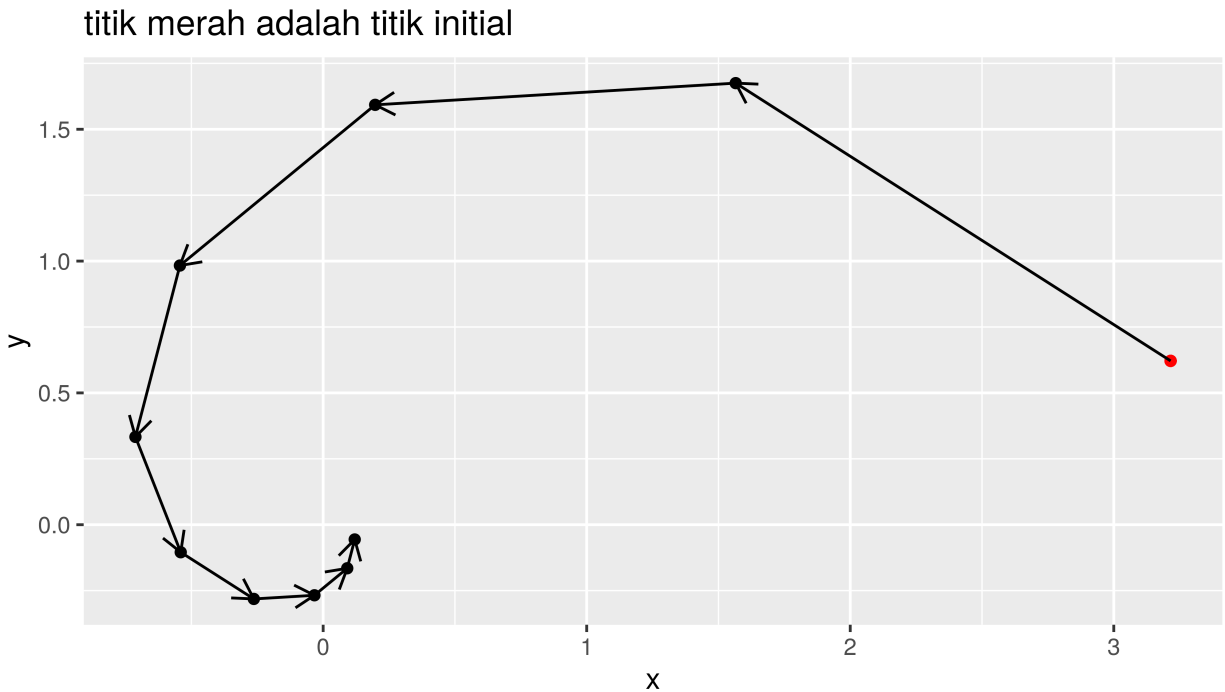


```
## 15 -0.38210 -0.830972
## 16 -0.14710 -0.809902
## 17  0.05304 -0.738936
## 18  0.21163 -0.632277
## 19  0.32600 -0.503805
## 20  0.39694 -0.366214
## 21  0.42799 -0.230394
## 22  0.42466 -0.105047
## 23  0.39370  0.003475
## 24  0.34242  0.091147
## 25  0.27809  0.156095
```

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  3.21567  0.62143
## 2  1.56538  1.67501
## 3  0.19731  1.59265
## 4 -0.54356  0.98312
## 5 -0.71233  0.33311
## 6 -0.54046 -0.10444
## 7 -0.26309 -0.28152
## 8 -0.03316 -0.26768
## 9  0.09136 -0.16523
## 10 0.11972 -0.05599
```

Terlihat bahwa semakin banyak rotasi dan kontraksi yang dilakukan akan membuat titik *initial* menuju pusat (0,0).

## Operasi Matriks Rotasi dan Kontraksi 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){
  # pusat rotasi
  pusat = 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 kontraksi
  for(i in 2:rot){
    xk = A %*% (x0-pusat) # diputar dengan x_bin sebagai pusat
    xk = pusat + (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 = pusat[1],
                  y = pusat[2]),
              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)
}

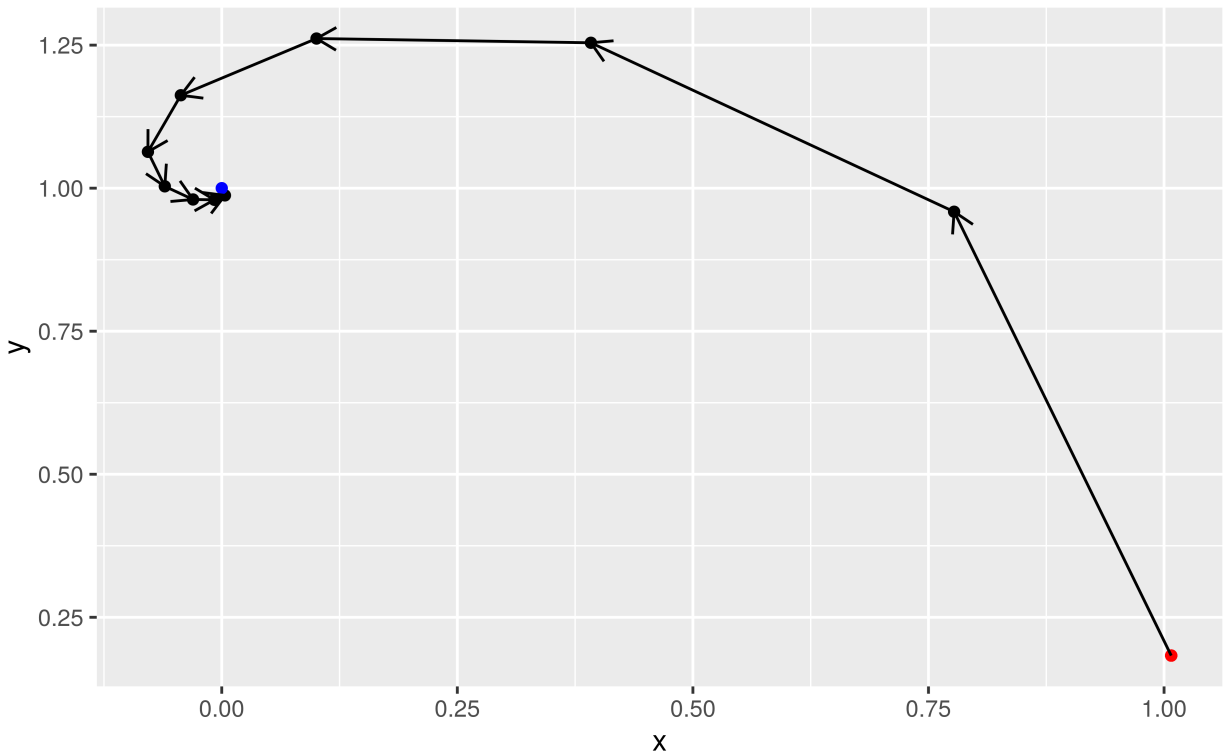
```

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(0,1) # contoh pusat rotasi
r = .6
rotasi_konstraksi_pusat_kan(x0,rot,r,x_bintang)
```

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

titik merah adalah titik initial  
titik biru adalah pusat rotasi



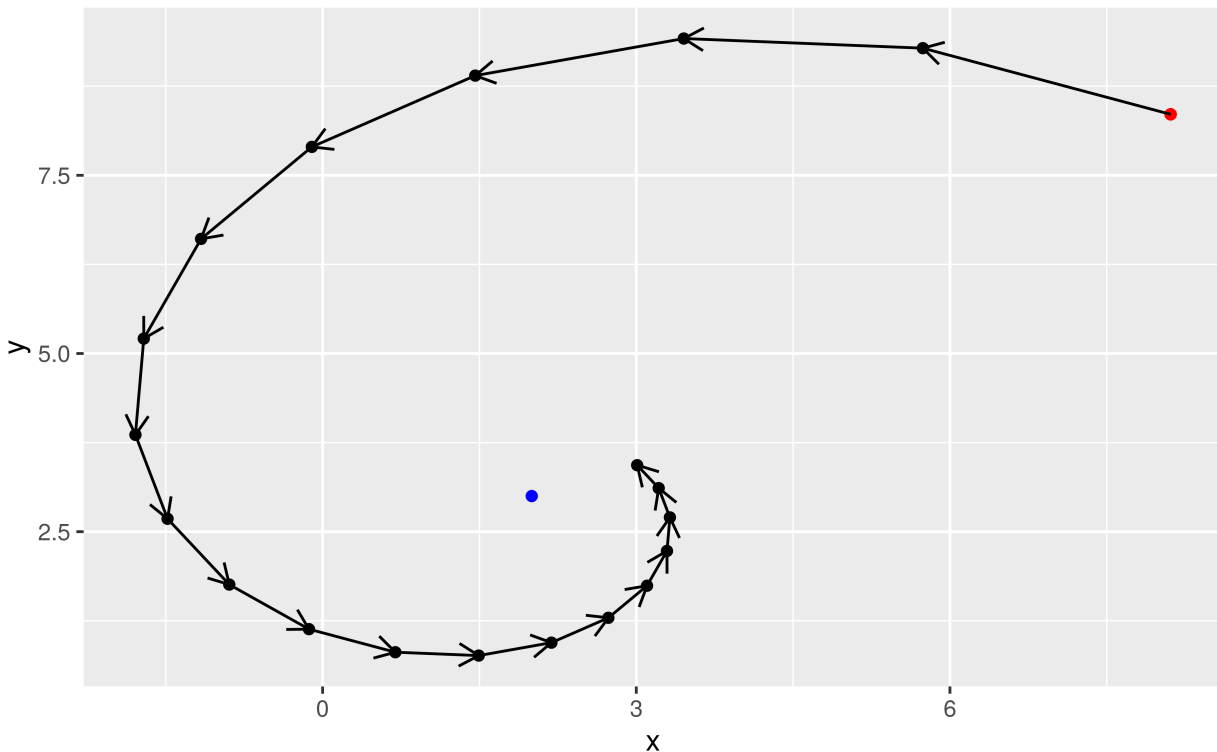
```
##
## $`Titik-titik rotasi`
##      x      y
## 1  1.007586 0.1831
## 2  0.777191 0.9588
## 3  0.391782 1.2541
## 4  0.100561 1.2615
## 5 -0.043415 1.1624
## 6 -0.078350 1.0635
## 7 -0.060434 1.0032
## 8 -0.030465 0.9802
## 9 -0.007820 0.9797
## 10 0.003376 0.9874
```

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 = .9
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  8.1107 8.3556
## 2  5.7410 9.2836
## 3  3.4545 9.4189
## 4  1.4598 8.8988
## 5 -0.1029 7.8988
## 6 -1.1624 6.6083
## 7 -1.7104 5.2090
## 8 -1.7903 3.8589
## 9 -1.4832 2.6810
## 10 -0.8927 1.7582
## 11 -0.1307 1.1326
## 12  0.6956 0.8090
## 13  1.4928 0.7619
```

##	14	2.1884	0.9432
##	15	2.7332	1.2919
##	16	3.1027	1.7419
##	17	3.2937	2.2298
##	18	3.3216	2.7005
##	19	3.2145	3.1112
##	20	3.0086	3.4330

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