

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)
}

```

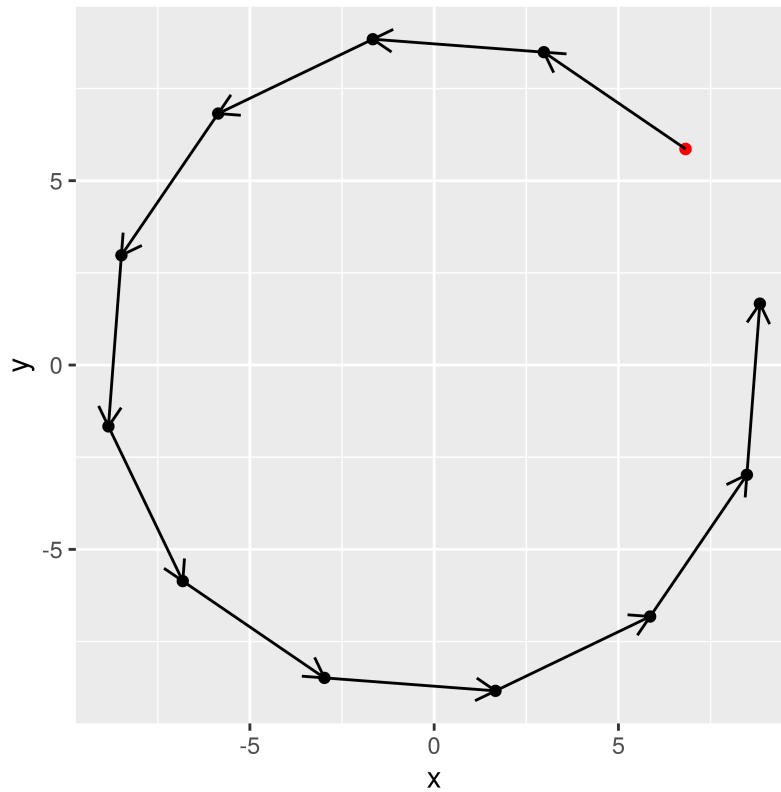
Berikut adalah uji coba dengan titik sembarang berikut ini:

```
# 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  6.821  5.861
## 2  2.977  8.486
## 3 -1.665  8.838
## 4 -5.861  6.821
## 5 -8.486  2.977
## 6 -8.838 -1.665
## 7 -6.821 -5.861
## 8 -2.977 -8.486
## 9  1.665 -8.838
## 10 5.861 -6.821
## 11 8.486 -2.977
## 12 8.838  1.665
```

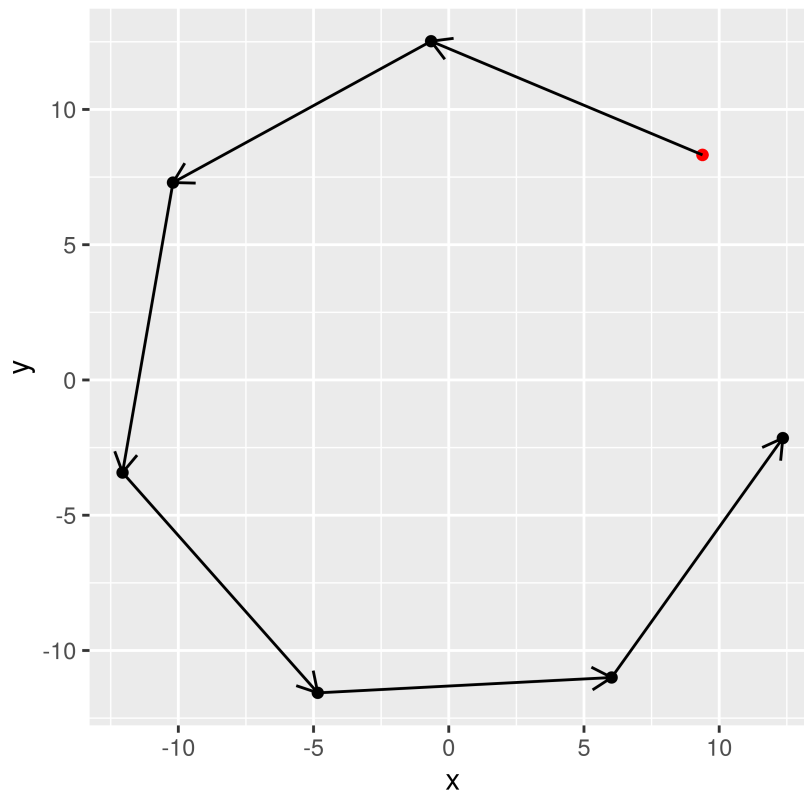
Uji coba kembali dengan titik sembarang lainnya berikut ini:

```
# 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  9.3825  8.318
## 2 -0.6537 12.522
## 3 -10.1976  7.296
## 4 -12.0625 -3.424
## 5 -4.8441 -11.565
## 6  6.0220 -10.998
## 7 12.3534 -2.149
```

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)
}

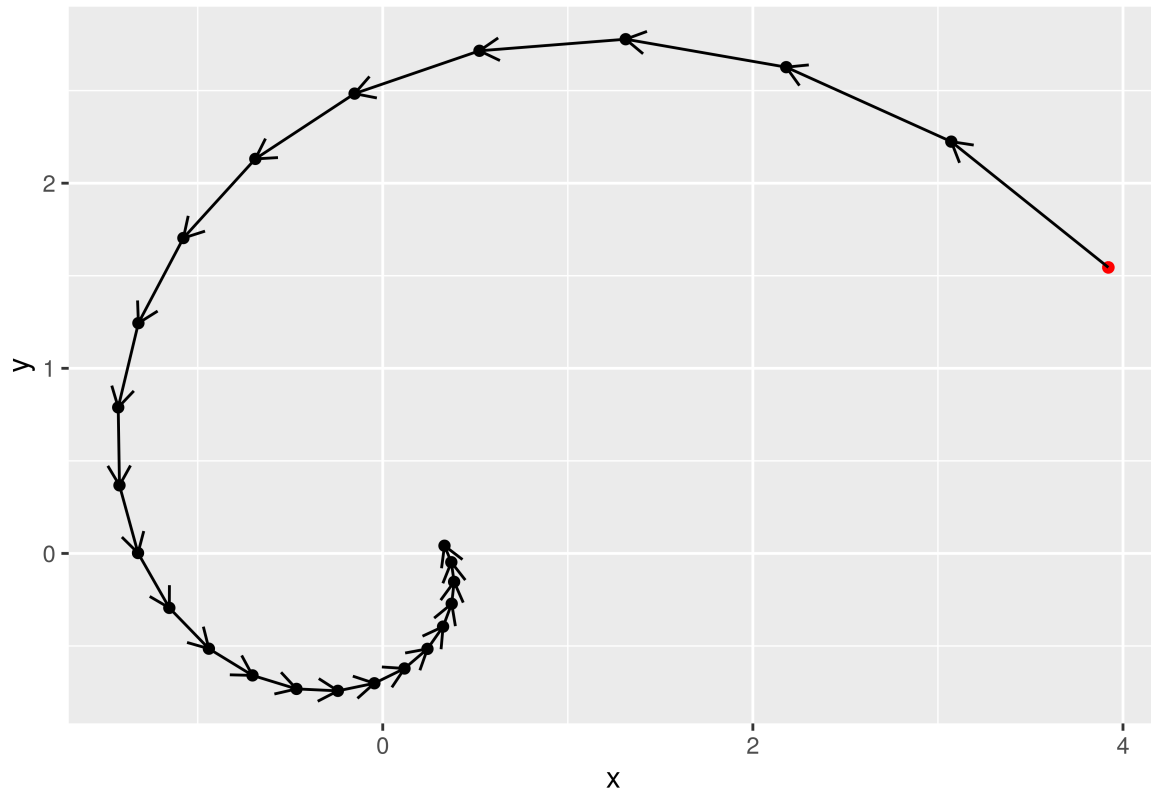
```

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 merah adalah titik initial



```
##
## $`Titik-titik rotasi`
##      x      y
## 1  3.92094  1.545002
## 2  3.07218  2.224405
## 3  2.18022  2.626686
## 4  1.31265  2.777728
## 5  0.52255  2.715212
## 6 -0.15220  2.483877
## 7 -0.68862  2.131192
## 8 -1.07729  1.703686
## 9 -1.32042  1.244025
## 10 -1.42948  0.788910
## 11 -1.42269  0.367764
## 12 -1.32251  0.002161
## 13 -1.15335 -0.294121
## 14 -0.93957 -0.514536
```

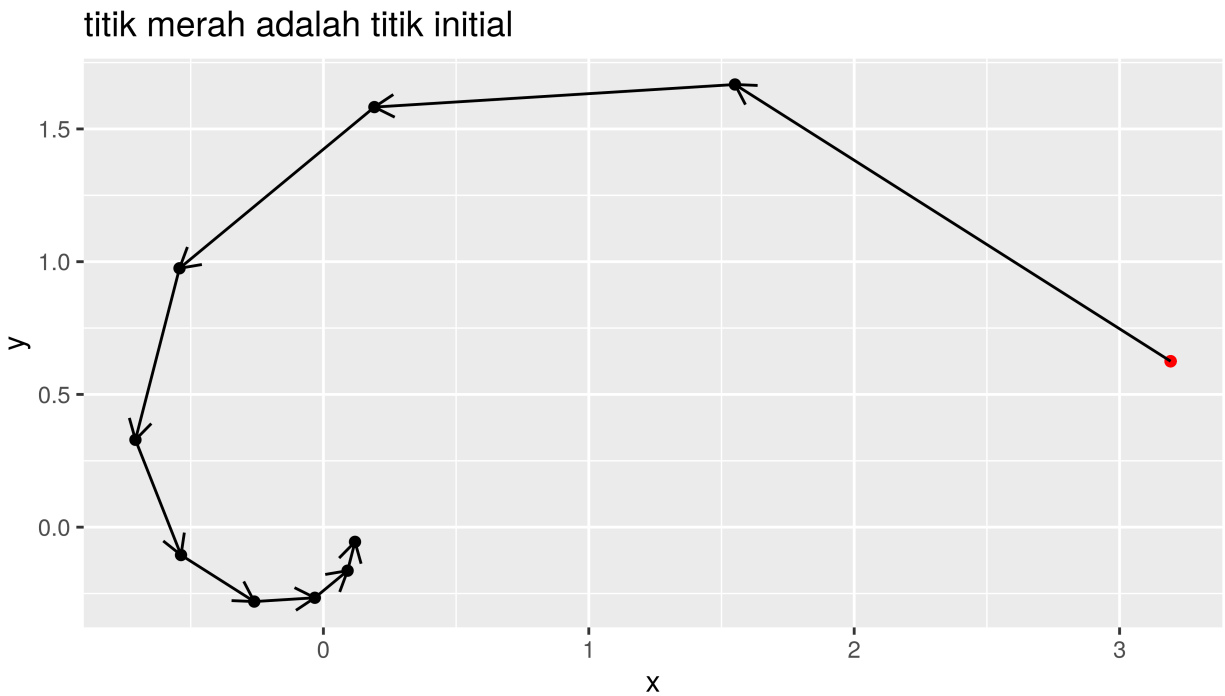


```
## 15 -0.70388 -0.658829
## 16 -0.46613 -0.731861
## 17 -0.24253 -0.742312
## 18 -0.04528 -0.701376
## 19  0.11751 -0.621541
## 20  0.24155 -0.515511
## 21  0.32595 -0.395319
## 22  0.37262 -0.271655
## 23  0.38562 -0.153409
## 24  0.37049 -0.047419
## 25  0.33358  0.041587
```

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  3.19230  0.62496
## 2  1.55070  1.66740
## 3  0.19213  1.58230
## 4 -0.54223  0.97513
## 5 -0.70829  0.32912
## 6 -0.53653 -0.10504
## 7 -0.26063 -0.28024
## 8 -0.03229 -0.26594
## 9  0.09113 -0.16389
## 10 0.11904 -0.05532
```

Catatan penting:

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. Operasi matriksnya adalah sebagai berikut:

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

Oleh karena itu kita akan modifikasi program bagian sebelumnya menjadi seperti ini:

```
# 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")))
)

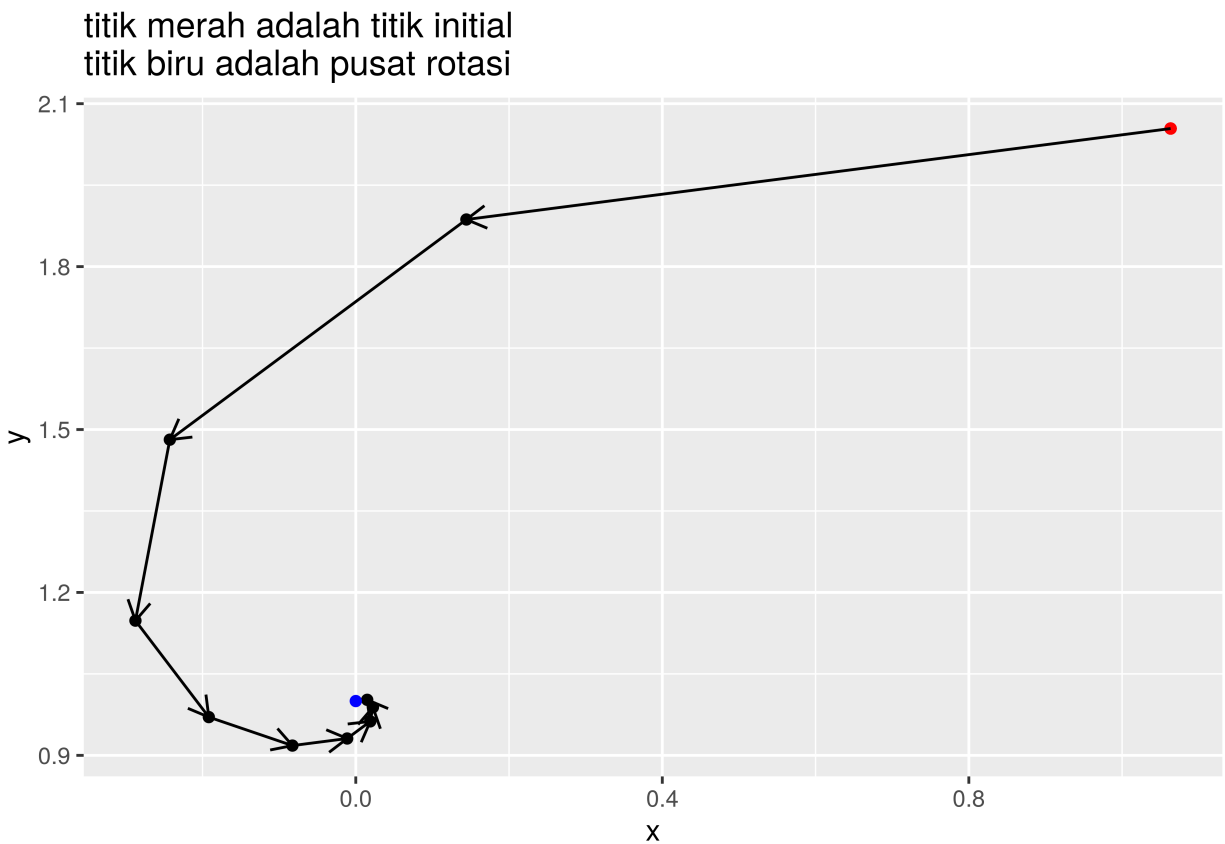
# menyiapkan output
list("Grafik rotasi" = plot,
     "Titik-titik rotasi" = final)
}

```

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`
```



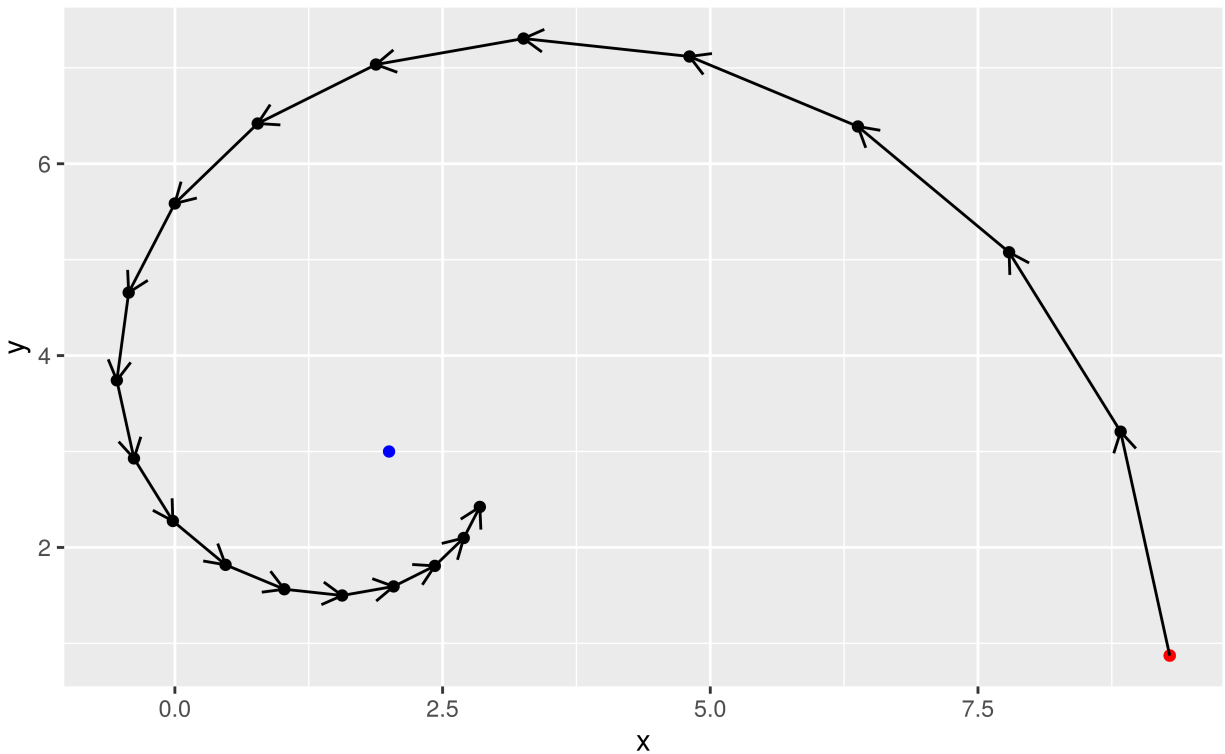
```
##
## $`Titik-titik rotasi`
##      x      y
## 1  1.06331 2.0542
## 2  0.14435 1.8867
## 3 -0.24266 1.4813
## 4 -0.28754 1.1481
## 5 -0.19180 0.9705
## 6 -0.08268 0.9180
## 7 -0.01122 0.9310
## 8  0.01887 0.9626
## 9  0.02236 0.9885
## 10 0.01491 1.0023
```

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  9.2908142 0.8725
## 2  8.8322615 3.2067
## 3  7.7906007 5.0771
## 4  6.3788076 6.3883
## 5  4.8057017 7.1180
## 6  3.2562515 7.3052
## 7  1.8779607 7.0344
## 8  0.7735170 6.4193
## 9 -0.0007666 5.5856
## 10 -0.4316645 4.6567
## 11 -0.5421497 3.7418
## 12 -0.3822623 2.9279
## 13 -0.0190576 2.2758
```

##	14	0.4732042	1.8186
##	15	1.0217123	1.5641
##	16	1.5619722	1.4989
##	17	2.0425525	1.5933
##	18	2.4276482	1.8078
##	19	2.6976242	2.0984
##	20	2.8478690	2.4223

Program *Spiral Optimization Algorithm*

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

```
# definisi
N = 50
a = -4
b = 4
k_max = 50
r = .9
rot = 16
f = function(x1,x2){
  ((x1^4 - 16 * x1^2 + 5 * x1)/2) + ((x2^4 - 16 * x2^2 + 5 * x2)/2)
}

# N pasang titik random di selang [a,b] di R2
x1 = runif(N,a,b)
x2 = runif(N,a,b)

# hitung theta
theta = 2*pi / rot

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

# bikin data frame
temp = data.frame(x1,x2) %>% mutate(f = f(x1,x2))

# proses iterasi
for(i in 1:k_max){
  # mencari titik x* dengan min(f)
  f_min = temp %>% filter(f == min(f))
  pusat = c(f_min$x1,f_min$x2)

  for(j in 1:N){
    # kita akan ambil titiknya satu persatu
    x0 = c(temp$x1[j],temp$x2[j])

    # proses rotasi dan kontraksi terhadap pusat x*
    xk = A %*% (x0-pusat) # diputar dengan x_bin sebagai pusat
    xk = pusat + (r * xk)

    # proses mengembalikan nilai ke temp
    temp$x1[j] = xk[1]
    temp$x2[j] = xk[2]
  }

  # hitung kembali nilai f(x1,x2)
  temp = temp %>% mutate(f = f(x1,x2))
}

temp %>% filter(f == min(f))
```


##		x1	x2	f
##	1	-2.915	-2.927	-78.32

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