IX Xornada de Usuarios de R en Galicia

Santiago de Compostela, 20 de outubro do 2022

Construyendo Mandalas con R/Construindo Mandalas com R

Luciane Ferreira Alcoforado[1]
[1] Academia da Força Aérea/Divisão de Ensino

RESUMO

A relação entre a Mandala e a matemática encontra-se no seu significado que vem do sânscrito e significa círculo. As mandalas são compostas por figuras geométricas que se repetem através de transformações matemáticos como isometrias e homotetias.
A partir das equações parametrizadas de uma seleção de curvas clássicas da matemática, combinados com as transformações matemáticas, é possível construir diversas formas de mandalas. Como recurso computacional a linguagem R.

Palabras e frases chave: curvas paramétricas, transformações, mandala, programação, linguagem R.

1. Introdução

Neste trabalho apresenta-se as equações parametrizadas de curvas clássicas da matemática, como gerar os pontos do eixo cartesiano e como programá-las no pacote ggplot2, Wickham (2016) da linguagem R, empregando transformações matemáticas com o objetivo de construir mandalas.
De acordo com Bezerra (2022), a mandala é, originalmente, um círculo que contém em seu interior desenhos de formas geométricas, figuras humanas e cores variadas e podem ser empregadas na educação como um recurso didático utilizado por professores de matemática, pois este símbolo serve para ensinar vários tópicos como geometría analítica, representação gráfica e lógica de programação.
Apresenta-se inicialmente as equações paramétricas de curvas como o círculo. Uma equação paramétrica, de acordo com a definição de MathWorld (2022), é um conjunto de equações que expressam um conjunto de pontos como funções explícitas de uma série de variáveis independentes, conhecidas como "parâmetros". A partir da equação paramétrica de uma curva, utiliza-se processos de transformações matemáticas gerando-se assim a mandala.

2. Transformações Matemáticas

Definição 1: Uma transformação no plano é uma função bijetora do conjunto dos pontos do plano sobre si mesmo.
Definição 2: Isometrias são transformações no plano que preservam distâncias, isto é, se T é uma isometria, para qualquer par de pontos A e B vale a relação $T(A)T(B) = AB$.

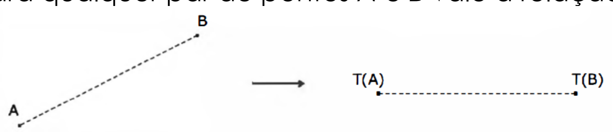


Figura 1: Isometria. Fonte: Rezende & Queiroz (2008).

Definição 3: Sejam A e B pontos distintos do plano. A translação T_{AB} é a isometria que leva um ponto X do plano ao ponto $T_{AB}(X) = X'$, tal que $ABX'X$ é um paralelogramo, se A, B e X não são colineares. Se A, B e X são colineares, então T_{AB} é tal que XX' está na reta

AB e os segmentos *AX'* e *BX* têm o mesmo ponto médio.

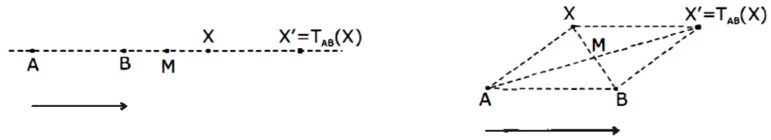


Figura 2: Translação. Fonte: Rezende & Queiroz (2008).

Considerando um ponto X de coordenadas (x,y), a translação para o ponto X´ de coordenadas (x´,y´) é obtido por

x´= x + a, a Є R; y´= y + b, b Є R

Definição 4: Seja O um ponto do plano e t um número real com $t \in [0,2\pi]$. A rotação de centro O e ângulo t é a isometria I_{Ot} que deixa fixo o ponto O e leva o ponto X, X ≠ O, no ponto X' = I_{Ot}(X), tal que OX = OX' e a medida do ângulo orientado (OX, OX') é igual a t, se t≠0 e t≠π. Além disso, OX' = OX, sendo O o ponto médio de XX', se t≠π; e X'= X se t =0.

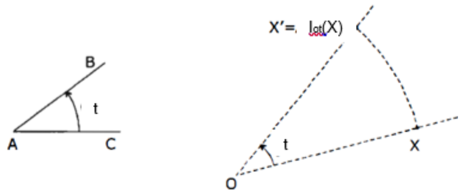


Figura 3: Rotação. Fonte: Rezende & Queiroz (2008).

Considerando um ponto X de coordenadas (x,y), a rotação por um ângulo t no sentido anti horário para o ponto X´ de coordenadas (x´,y´) é obtido por

x´= xcos(t) – ysin(t), t Є [0,2π]; y´= xsin(t) + ycos(t), t Є [0,2π]

Enquanto duas figuras isométricas têm a mesma forma e as mesmas dimensões, isto é, são congruentes, duas figuras homotéticas conservam apenas a mesma forma. Dizemos que duas figuras homotéticas são semelhantes.

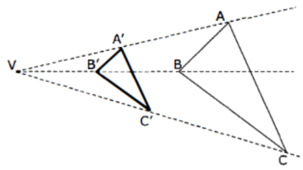


Figura 4: Homotetia de dois triângulos ABC e A'B'C'. Fonte: Rezende & Queiroz (2008).

Considerando um ponto X de coordenadas (x,y), a redução por um fator k do ponto X´ de coordenadas (x´,y´) é obtido por

x´= kx, k Є (0,1); y´= ky, k Є (0,1)

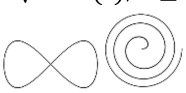
3. Equações Paramétricas de curvas clássicas

Equações paramétricas podem ser obtidas em Stover & Weisstein (2022):

Circulo: $x = rcos(t); y = rsin(t),\ t \in [0,2\pi]$; Elipse: $x = acos(t); y = bsin(t),\ t \in [0,2\pi]$

Cardióide: $x = 2r \cdot cos(t) - r \cdot cos(2t); y = 2 \cdot rsin(t) - r \cdot sin(2t),\ \ t \in [0,2\pi]$; Limaçon: $x = (b + a \cdot cos(t)) \cdot cos(t); y = (b + a \cdot cos(t)) \cdot sin(t),\ t \in [0,2\pi]$

Lemniscata: : $x = sin(t); y = \cos(t)\,.\,sin(t),\ t \in [0,2\pi]$; Espiral de Fermat: $x = a\sqrt{t}.cos(t);\ y = a\sqrt{t}.sin(t),\ t \geq 0$



4. Programando em R

Considerando as equações paramétricas das curvas clássicas em conjunto com as transformações matemáticas de isometria e homotetia, passaremos ao script na

linguagem R para construção e exibição das mandalas.
O procedimento está decomposto em duas partes, na primeira parte elabora-se a criação de uma base de dados contendo todos os pontos das coordenadas (x,y) que formam o desenho da mandala. Na segunda parte elabora-se a visualização do desenho utilizando-se o pacote ggplot2. O desenvolvimento dos scripts foi realizado com base em Alcoforado (2021).
Mandala 1: Elaborada a partir do círculo com translação e rotação
Inicialmente criamos em x e y os pontos que formam um círculo de raio 1, posteriormente criamos em xt e yt a translação dos pontos iniciais apenas no sentido do eixo x, figura 5:

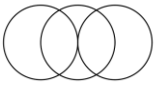


Figura 5: Resultado da translação do círculo na direção do eixo x. Fonte: autora, 2022.

Posteriormente é realizado a rotação da figura 5 nos ângulos de $\frac{\pi}{8}, \frac{\pi}{4}, \dots, 2\pi$, tais pontos são armazenados em um *dataframe* denominado dt. O número de linhas em dt é formado por 25500, ou seja, 25500 pontos a serem plotados.
Script – Parte 1 gerando pontos e armazenando em dt

```
#Parâmetros
n=500; raio=1; t=seq(0,2*pi, length.out = n)
#pontos para círculo inicial
x=raio*cos(t)
y=raio*sin(t)
#pontos para os 3 círculos
xt=c(x,x-raio,x-2*raio)
yt=c(y,y,y)
```

```
rotacao = (pi/8)*(1:16); n=length(xt); xt1=xt; yt1=yt
for(i in 1:length(rotacao))
  {
xt1=c(xt1,xt[1:n]*cos(rotacao[i])-yt[1:n]*sin(rotacao[i]))
yt1=c(yt1,xt[1:n]*sin(rotacao[i])+yt[1:n]*cos(rotacao[i]))
}
dt= data.frame(xt1,yt1,z="circulo")
```

Script – Parte 2 Visualizando a mandala

```
p= ggplot()+
coord_fixed()+
    theme_void()
p=
  p+
   geom_point(data=dt, aes(x=xt1, y=yt1), color='black')
p
```

Mandala 2: Elaborada a partir do cardióide com rotação
Inicialmente criamos em x e y os pontos que formam um cardióide de raio 1, posteriormente é realizado a rotação dos pontos do cardióide nos ângulos de $\frac{\pi}{4}, \frac{\pi}{2}, \dots, \frac{7\pi}{4}$, tais pontos são armazenados em um *dataframe* denominado dt. O número de linhas em dt é formado por 4000, ou seja, 4000 pontos a serem plotados.
Script – Parte 1 gerando pontos e armazenando em dt

```
n=500; t=seq(0, 2*pi, length.out = n); rotacao=pi/4*(1:7)
x=c(2*raio*cos(t)-raio*cos(2*t))
y=c(2*raio*sin(t)-raio*sin(2*t))
xt=x; yt=y #rotação dos pontos
for(i in 1:length(rotacao)){
xt=c(xt, x[1:n]*cos(rotacao[i])-y[1:n]*sin(rotacao[i]))
yt=c(yt, x[1:n]*sin(rotacao[i])+y[1:n]*cos(rotacao[i]))
}
dt= data.frame(xt, yt, z="cardióide")
```

Script – Parte 2 Visualizando a mandala (o mesmo já descrito)

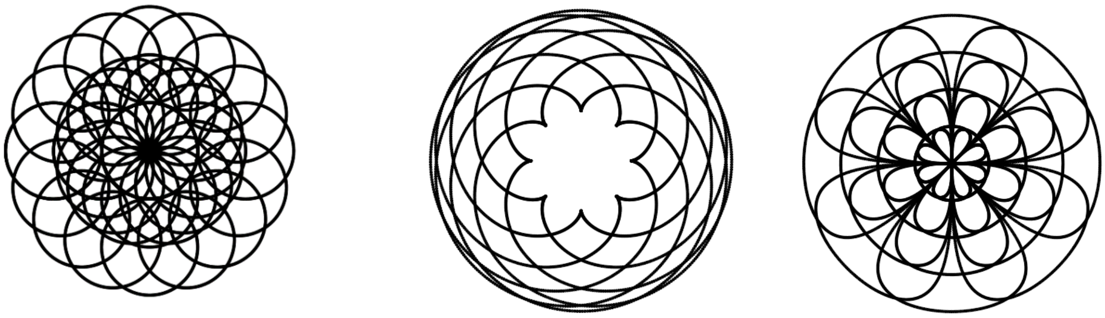


Figura 6: Mandala feita de círculo transladado e rotacionado (esquerda) Mandala feita de cardióide rotacionado (centro) e Mandala feita da Lemniscata com rotação e contração (direita). Fonte: autora, 2022.

Mandala 3: Elaborada a partir da lemniscata com rotação e contração

Inicialmente criamos em x e y os pontos que formam uma lemniscata, posteriormente é realizado a rotação dos pontos da lemniscata nos ângulos de $\frac{\pi}{4}, \frac{\pi}{2}$ $e \frac{3\pi}{4}$, tais pontos são armazenados em xt, yt e finalmente aplica-se uma redução de fator 0.25, 0.5 e 0.75 nos pontos gerados anteriormente, armazenando todos eles em um *dataframe* denominado dt. Em dt há 8000 pontos a serem plotados.

Script – Parte 1 gerando pontos e armazenando em dt

```
n=500; t=seq(0, 2*pi, length.out = n); rotacao=pi/4*(1:3)
x=sin(t);  y=sin(t)*cos(t)
xt=x; yt=y#rotações
for(i in 1:length(rotacao)){
xt=c(xt, x[1:n]*cos(rotacao[i])-y[1:n]*sin(rotacao[i]))
yt=c(yt, x[1:n]*sin(rotacao[i])+y[1:n]*cos(rotacao[i])) }
xtt=NULL; ytt=NULL; red=c(0.25, 0.5, 0.75) #redução
for(i in 1:length(red)){
provx=paste0("x",i); provy=paste0("y",i)
xtt=c(xtt, assign(provx, xt*red[i]))
ytt=c(ytt, assign(provy, yt*red[i])) }
dt=data.frame(x=c(xt, xtt), y=c(yt, ytt), z="lemniscata")
```

Script – Parte 2 Visualizando a mandala (o mesmo já descrito)

Outras combinações a partir de figuras geométricas podem ser realizadas, figura 7:

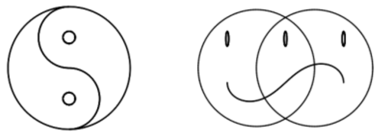


Figura 7: Mandalas feitas com a combinação de figuras geométricas como círculo, elipse e lemiscata. Fonte: autora, 2022.

5. Aplicativo Shiny para mandalas

Esta experiência de programar as mandalas levou a produção de um aplicativo shiny disponível em https://lucianealcoforado.shinyapps.io/Mandala/, Alcoforado (2022).

O aplicativo possui uma base de pontos para gerar diversas opções de mandalas e o usuário pode optar por colorir os pontos e o plano de fundo.

Referencias

[1] Alcoforado, L.F. (2022), *Mandala*. Aplicativo Disponível em https://lucianealcoforado.shinyapps.io/Mandala/. Acesso em 11/09/2022.

[2] Alcoforado, L.F. (2021) *Utilizando a Linguagem R: conceitos, manipulação, visualização, Modelagem e Elaboração de Relatório*, Alta Books, Rio de Janeiro.

[3] Bezerra, J. (2022), *Mandalas*, Toda Materia. Disponível em https://www.todamateria.com.br/mandala/. Acesso em 11/09/2022.

[4] Rezende, E.Q.F, Queiroz, M.L.B. (2008). *Geometria Euclidiana Plana de Construções Geométricas*, 2a. ed., Unicamp, Campinas.

[5] Stover, C., Weisstein, E.W. (2022). *Parametric Equations*. From MathWorld - A Wolfram Web Resource. https://mathworld.wolfram.com/ParametricEquations.html, Acesso em 11/09/2022.

[6] Wickham, H. (2016), *ggplot2: Elegant Graphics for Data Analysis*, Springer-Verlag, New York