

# Verificando Computacionalmente o Teorema de Perron-Frobenius Parte II: Demonstração e Verificação

Pedro Cavalcante

2019-08-05

## Um Pequeno Aviso

Este post é - como o nome indica - uma continuação de outro. Sua leitura solitária pode fazer pouco ou nenhum sentido se o leitor não está familiarizado com os conceitos introduzidos na primeira parte.

## Plano de Vôo

Na primeira parte fomos apresentados a muita coisa então vale a pena refresca-las um pouco antes de entender para onde vamos. Primeiro conhecemos o conceito de *ponto fixo*. Depois de verificar que é relativamente simples encontrar pontos fixos em funções afins, percebemos que a tarefa fica bem complexa em pouco tempo e que por isso é interessante usar Teoremas de Ponto Fixo para garantir a existência destes objetos para certas funções que atendem a determinados critérios. Nos focamos no Teorema do Ponto Fixo de Brouwer, que garante que toda função contínua com domínio fechado, limitado e convexo e imagem neste mesmo conjunto admite pelo menos um ponto fixo.

Provamos esse resultado em três casos particulares, cada um se beneficiando de estratégias de provas bem diferentes. Primeiro vimos que com ferramentas simples do Cálculo conseguíamos demonstrar a existência de ponto fixo em funções que mapeiam o fechado  $[0, 1]$  nele próprio - bem como a interpretação geométrica associada. Depois usamos o Lema da Não-Retração para mostrar que não é tão difícil provar a validade do teorema em duas dimensões. Pelo contrário, era uma decorrência bem intuitiva de compor retas entre pontos em um círculo. Por fim, depois de apresentados o conceito de Simplex e de Colorações de Sperner, usamos um argumento de Combinatória para mostrar que até em dimensões arbitrariamente grandes o resultado valia.

E por quê eu dediquei algumas páginas de texto e semanas da minha vida escrevendo a primeira parte se o nosso objetivo é “apenas” usar o poder do R para simular matrizes, calcular seus autovalores e *ver o Teorema de Perron-Frobenius acontecendo*? Bem, porque eu acho que demonstra-lo é (i) interessante em si e deixa tudo menos “caído dos céus”, (ii) porque existe uma prova simples e elegante de um caso particular porém suficientemente geral para este post usando o Teorema do Ponto Fixo de Brouwer - ele próprio um resultado elegante - e (iii) porque no meio do caminho ganhei motivos para introduzir suave e contextualizadamente vários conceitos que usaremos na demonstração do Teorema de Perron-Frobenius.

Depois da longa parte I, precisamos agora conhecer apenas mais um conceito e estamos prontos para ver a magia acontecendo.

## Irreducibilidade

- *Definição:* Dizemos que uma matriz  $A_{n \times n} = [a_{ij}]$  é *reducível* se seus índices  $1, 2, \dots, n$  podem ser divididos em dois conjuntos não-vazios e disjuntos  $i_1, i_2, \dots, i_p$  e  $j_1, j_2, \dots, j_q$  (respeitando a limitação de que  $p + q = n$ , não podendo “criar dimensões”) de forma que qualquer elemento  $a_{ij}$  da matriz que tenha índices nesses conjuntos seja igual a 0.

Essa definição seca pode parecer estranha, mas tem uma interpretação mais visual muito intuitiva. Uma matriz é reducível se podemos rearranjar suas linhas e colunas de forma que ela tenha um “bloco” de zeros.

Imagine que temos uma matriz 2 por 2 em que cada entrada é uma matriz quadrada  $n$  por  $n$ .  $A$ ,  $B$  e  $C$  são matrizes com elementos quaisquer e  $\mathbf{0}$  é uma matriz preenchida de zeros. Se uma matriz  $D$  é reducível, então admite uma representação assim:

$$D = \begin{pmatrix} A & C \\ \mathbf{0} & B \end{pmatrix}$$

Em mais detalhes:

$$D = \begin{pmatrix} \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \ddots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix} & \begin{bmatrix} c_{11} & c_{12} & \dots & c_{1n} \\ c_{21} & c_{22} & \dots & c_{2n} \\ \vdots & \ddots & \ddots & \vdots \\ c_{n1} & c_{n2} & \dots & c_{nn} \end{bmatrix} \\ \begin{bmatrix} 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \end{bmatrix} & \begin{bmatrix} b_{11} & b_{12} & \dots & b_{1n} \\ b_{21} & b_{22} & \dots & b_{2n} \\ \vdots & \ddots & \ddots & \vdots \\ b_{n1} & b_{n2} & \dots & b_{nn} \end{bmatrix} \end{pmatrix}$$

Embora a matriz de zeros esteja na entrada esquerda inferior neste exemplo,  $D$  seria redutível se conseguíssemos rearranjar seus vetores de forma que a matriz de zeros fosse formada em qualquer canto.

## O Teorema de Perron-Frobenius

- *Teorema (Perron-Frobenius):*
- *Prova:*

## Verificando esse resultado computacionalmente

A função `eigen()` faz a decomposição espectral de matrizes com métodos numéricos então a usaremos para aproximar os autovalores e vetores de várias matrizes simuladas.

```
matriz = runif(n = m) %>%
  round(digits = 2) %>%
  matrix(ncol = 2)

DF = tibble(R1 = rep(NA, times = (n*t)),
            R2 = rep(NA, times = (n*t)),
            t = rep(NA, times = (n*t)))

for(j in 1:n) {

  vetor = c(runif(min = -5, max = 5, n = 1) %>% round(digits = 2),
            runif(min = -5, max = 5, n = 1) %>% round(digits = 2)) # inicializamos um vetor

  for(i in 1:t) {

    indice = if(j > 1) {ifelse(((j*t) + i) > 10000,
                              10000,
                              ifelse(j > 1, (j*t) + i, i))}
              else { indice = i} # trambicagem para o índice correr corretamente
```

```

vlinha = (matriz^i) %*% vetor # aplicamos a matriz

DF$R1[indice] = vlinha[1]
DF$R2[indice] = vlinha[2]
DF$t[indice] = i

}

rm(vetor)

}

DF %>% ggplot(aes(x = R1, y = R2, colour = t) +
  geom_point()

```