

Módulo 04 - Simulação de grafos aleatórios

Já vimos como calcular facilmente diversas propriedades de uma rede. Contudo, quando o objeto da análise é apenas uma rede (ou seja, não há outra rede para usar como parâmetro para comparações), pode ser difícil chegar a uma conclusão sobre aqueles valores. Por exemplo, uma rede de N vértices e densidade p possui uma transitividade x . Esse valor é normal ou mais alto ou baixo que o normal? Uma possível abordagem para responder a perguntas como essa é por meio de redes ou grafos aleatórios.

Um grafo aleatório é um grafo gerado por um processo aleatório e consiste de N vértices onde cada par de vértices é conectado por uma aresta com uma probabilidade p . Por isso, é representado pela notação $G(N, p)$.

Para construir manualmente uma rede aleatória, deve-se seguir estes passos:

1. Comece com um conjunto de N vértices isolados;
2. Selecione um par de vértices e gerar um número aleatório entre 0 a 1. Se o número for maior que p , conecte-os com uma aresta, caso contrário não os conecte;
3. Repita a etapa 2 para cada um dos $N(N - 1)/2$ pares de vértices.

Os matemáticos Pál Erdős e Alfréd Rényi, em um trabalho publicado em 1959, descreveram como entender as propriedades de um grafo aleatório. Por isso, esse modelo ficou conhecido como Rede de Erdős-Rényi.

Cada rede aleatória gerada com os mesmos parâmetros N e p possuem diferenças entre si devido à aleatoriedade do processo. Algumas propriedades que apresentam valores diferentes entre as redes são, para citar algumas, a quantidade L de arestas e o grau médio k . As equações para essas propriedades de um processo aleatório são mostradas abaixo:

$$L = p(N(N - 1))/2$$

$$k = p(N - 1)$$

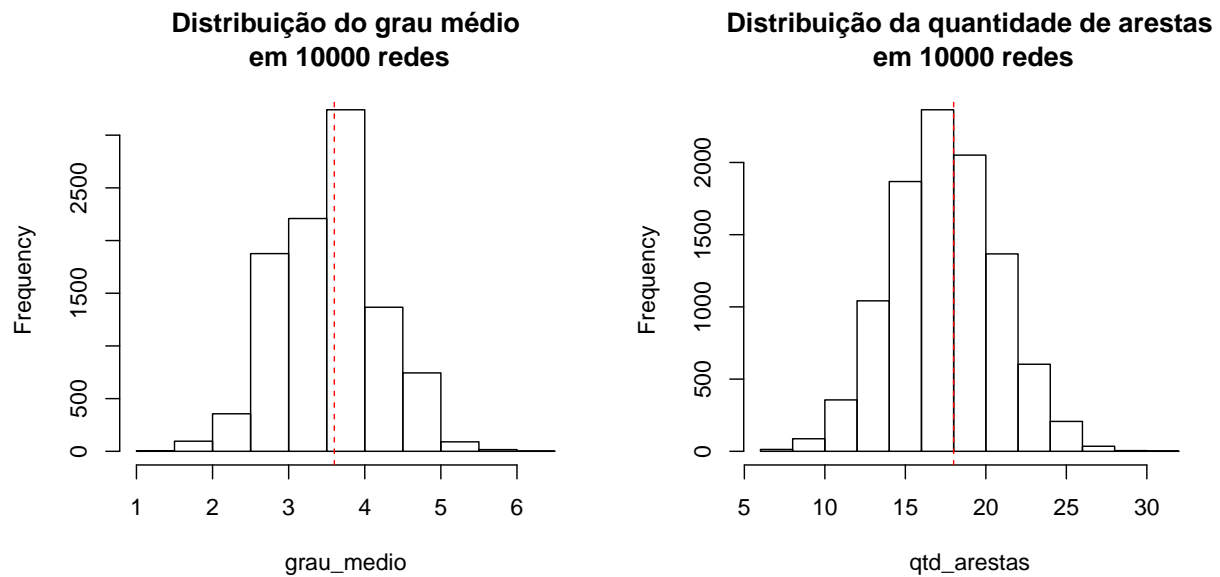
Por exemplo, para um conjunto de 10000 redes aleatórias com os parâmetros $N = 10$ e $p = 0.4$, a distribuição da quantidade de arestas e do grau médio é mostrada nos histogramas abaixo (junto com o código R usado para gerar os resultados):

```
# gerar 10000 redes aleatorias
N <- 10
p <- 0.4
lista_grafos <- replicate(10000, igraph::sample_gnp(n = N, p = p),
                          simplify = FALSE)
# calcular o grau medio em todos os grafos da lista
grau_medio <- sapply(lista_grafos, function(x) mean(degree(x)))
# calcular grau medio esperado de acordo com a equacao
grau_medio_esperado <- p * (N - 1)

# calcular a quantidade de arestas em todos os grafos da lista
qtd_arestas <- sapply(lista_grafos, ecount)
qtd_arestas_esperado <- p * N * (N-1)/2

# plotar histogramas
par(mfrow = c(1, 2))
# histograma do grau medio com linha vertical mostrando valor esperado
hist(grau_medio, main = "Distribuição do grau médio\n em 10000 redes")
abline(v = grau_medio_esperado, col = "red", lty = "dashed")
# histograma de L
```

```
hist(qtd_arestas, main = "Distribuição da quantidade de arestas\n em 10000 redes")
abline(v = qtd_arestas_esperado, col = "red", lty = "dashed")
```



Os histogramas mostram que, mesmo possuindo os mesmos parâmetros, redes podem divergir bastante entre si, pois podem apresentar desde 5 até 30 arestas.

Referências <http://barabasi.com/networksciencebook/> CAPÍTULO 03