Lista 2 - MAC0328

Matheus de Mello Santos Oliveira - 8642821

Agosto 2017

1 Objetivo

Gostaríamos de determinar empiricamente a probabilidade de dois vértices escolhidos aleatoriamente estarem ao alcance um do outro em grafos aleatórios.

2 Experiência

Com o objetivo de determinar experimentalmente a probabilidade de dois vértices escolhidos aleatoriamente estarem ao alcance um do outro, fizemos experimentos da seguinte forma:

1. Fixamos V vértices e T testes - a serem rodados para cada quantidade de arestas - que são entradas do programa.
2. Geramos um grafo não dirigido aleatório com número esperado de arestas E.
3. Executamos uma DFS para calcular as componentes conexas do grafo e também o tamanho de cada uma delas no vetor ccSize indexado pelas componentes.
4. Usamos a informação do tamanho de uma dada componente para calcular a quantidade de pares de vértices que estão ao alcance um do outro na mesma da seguinte forma:
$$\binom{ccSize[i]}{2}$$
Somamos esse valor para cada componente i e o total dividimos por $V(V-1)/2$, que é o total de arestas do grafo. Com isso temos a probabilidade neste grafo de dois vértices escolhidos aleatoriamente estarem ao alcance um do outro.
5. Agora repetimos o passo 4 T vezes e tiramos a média.
6. Repetimos passos 2-5 variando a quantidades de arestas esperadas de 0 até $20V$ ou $V(V-1)/2$), o que for menor, por motivos que veremos na próxima seção.

3 Resultados

Primeiramente é preciso notar que é esperado que o aumento na quantidade de testes nos traga um refinamento da informação adquirida, pois ao aumentar o número de testes, diminuimos os riscos de casos atípicos contaminarem o resultado, estabilizando tanto média, variância quanto desvio padrão.

E percebemos exatamente isso. Para V fixado em 100 temos abaixo gráfico dos resultados para $T = 10$ e para $T = 100$. O eixo y representa a probabilidade de dois vértices escolhidos aleatóriamente estarem ao alcance um do outro e o eixo x o número de arestas esperadas na criação do grafo aleatório.

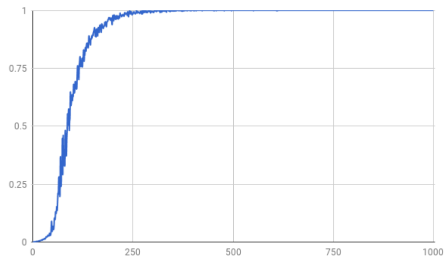


Figure 1: 10 testes.

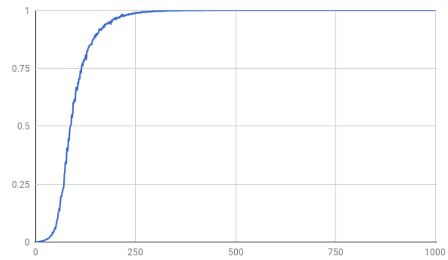
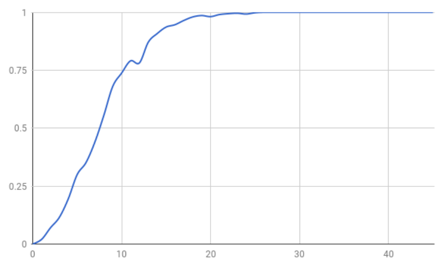
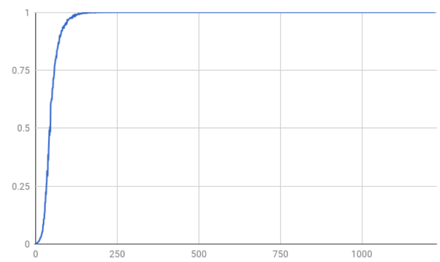


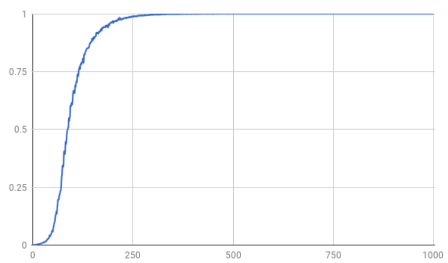
Figure 2: 100 testes.

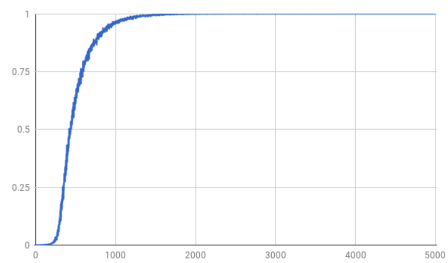
Os dois gráficos demonstram o esperado, ou seja, um refinamento da informação, e por esse motivo, adotamos como padrão uma bateria de 100 testes por quantidade de aresta para o desenvolvimento deste projeto.

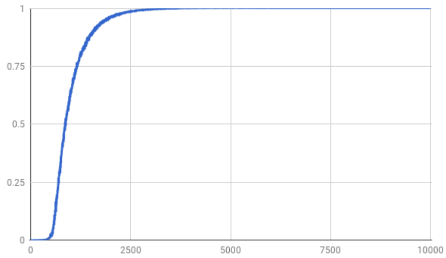
Eis os gráficos para 10, 50, 100, 500 e 1000 vértices.











Outro ponto que notamos é o fato de que o número de arestas que precisamos para que a probabilidade de dois vértices escolhidos aleatoriamente estarem ao alcance um do outro seja alta parece depender de V. Com os dados deste experimento podemos dizer empiricamente que para V menores ou iguais a 1000 o número de arestas necessárias para que o grafo tenha somente uma componente é superiormente limitado por $V \log_2(V)$. Daqui fica claro a idéia de limitar a quantidade de arestas em $20V$ implementada no algoritmo, pois $log_2(1000) \leq 10$ o fator 2 entra para garantir que estamos abrangendo a faixa necessária.

Número de vértices	Número de arestas para o grafo ter somente uma componente	$V \log_2(V)$
10	25	33
50	250	282
100	500	664
500	3000	4480
1000	5000	9960

4 Implementação

Este código foi implementado utilizando matriz de adjacência principalmente pelo fato de que o gargalo do desempenho está na própria criação do grafo, que tem consumo de tempo: $O(V^2)$. Não somente isto, matriz de adjacência permite inserção de aresta em tempo constante, ao passo que listas de adjacência tem pior caso linear no número de vértices o que tornaria a criação de grafos aleatórios $O(V^3)$

5 Uso do programa

Para compilar:

$ make

Para executar o código:

$ ./ep2 [V] [T]

Onde V é o número de vértices e T o número de testes a serem executados para cada quantidade de arestas.