SALÃO DE INICIAÇÃO CIENTÍFICA XXIX SIC

UFRGS PROPESQ

Salão UFRGS 2017 CONHECIMENTO FORMAÇÃO INOVAÇÃO

múltipla UNIVERSIDADE inovadora inspiradora

Localização de Autovalores em Grafos Árvore

Rafael Jacobs Kehl
Orientador: Carlos Hoppen

Universidade Federal do Rio Grande do Sul

Instituto de Matemática e Estatística

Sumário

Definições

Grafo

Um grafo é formado pelos conjuntos (V, A) em que V é um conjunto arbitrário e A são pares não ordenados de V. Os elementos de V são chamados *vértices* e os de A de *arestas*.

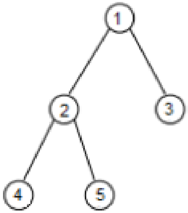


Figura: Exemplo de um grafo.

Definições

Grafo Conexo

Um grafo é conexo se, para qualquer par $\{v, w\}$ de seus vértices, existe um caminho com extremos v e w.

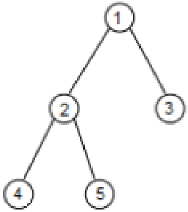


Figura: Grafo conexo.

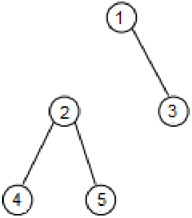


Figura: Grafo desconexo.

Definições

Grafo Árvore

Um grafo árvore, ou simplesmente árvore, é aquele que não possui ciclos e é conexo.

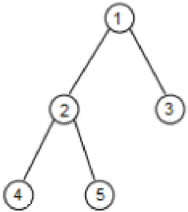


Figura: O grafo utilizado como exemplo é uma árvore.

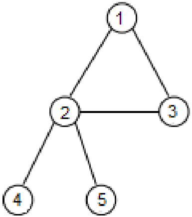


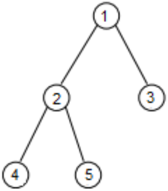
Figura: Grafos com ciclos não são árvores.

Definições

Matriz de Adjacência

A matriz de adjacência é uma das formas de representar um grafo. As entradas da matriz são tais que

$$a_{ij} = \begin{cases} 1, & \text{se } i \text{ é adjacente a } j \\ 0, & \text{se } i \text{ não é adjacente a } j \end{cases}$$



$$\begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

Figura: Grafo e sua matriz de adjacência.

Definições

Matrizes Congruentes

Dizemos que duas matrizes R e S são congruentes se existe uma matriz não singular P que satisfaz $R = P^T SP$.

Em resumo, duas matrizes são congruentes quando é possível obter uma fazendo as mesmas operações fundamentais nas linhas e colunas da outra.

Teorema

Lei da Inércia de Sylvester

Duas matrizes reais simétricas de ordem $n \times n$ são congruentes se e somente se elas têm o mesmo número de autovalores negativos e o mesmo número de autovalores positivos.

Motivação

Para a Teoria Espectral dos Grafos, é de grande interesse ter uma forma eficiente de verificar se um número é autovalor do grafo ou se há autovalores num dado intervalo.

Uma maneira de se localizar os autovalores de uma árvore T seria computando a sequência de Sturm para o polinômio característico de $\mathcal{X}(\lambda)$, mas só para obter $\mathcal{X}$ são necessárias $\mathcal{O}(nlog^2n)$ operações.

Por outro lado, há algoritmo de Jacobs e Trevisan para árvores que consegue o mesmo feito com somente $\mathcal{O}(n)$ operações.

Motivação

O algoritmo é baseado na diagonalização de $B_\alpha = A + \alpha I$, onde A é a matriz de adjacências de T.

Seja *Diagonalize*(A, α) a função que aplica o algoritmo de **Jacobs e Trevisan** e retorna a matriz D congruente a B_α, fazendo uso da **Lei da Inércia de Sylvester**, teremos então o **Teorema 1**, que será enunciado e demonstrado a seguir.

Teorema

Teorema 1

Seja $D = Diagonalize(A, -\alpha)$. Temos então:

a O número de entradas positivas de D é o número de autovalores de A maiores que α.

b O número de entradas negativas de D é o número de autovalores de A menores que α.

c O número de entradas nulas na diagonal de D é a multiplicidade de α como autovalor de A.

Demonstração

Seja A a matriz de adjacências de T, com autovalores λ_i, e sejam β_i os autovalores de $B_{-\alpha}$.

Agora sejam d_i os elementos na diagonal de D.

Para todos acima, temos que $i = 1, ..., n$.

Demonstração

Note que $\beta_i = \lambda_i - \alpha$. Então, $\lambda_i > \alpha$ se, e somente se, $\beta_i > 0$.

Então, basta observar que para cada $d_i > 0$ há um $\beta_i > 0$. Logo, há o mesmo número de $\lambda_i > \alpha$. Isso demonstra **a**.

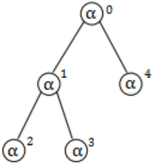
Podemos demonstrar **b** e **c** de forma análoga, basta trocar $>$ por $<$ ou $=$, respectivamente.

$\square$

Algoritmo

Seja T uma árvore com n vértices e A sua matriz de adjacência. Escolha um vértice v_0 arbitrário para ser a raiz da árvore, então enumere os demais vértices $v_1, ..., v_{n-1}$ tal que se v_i é filho de v_j então $i > j$.

Dessa forma a raiz será sempre v_0. Considere agora a matriz B_α para um escalar α. Note que o valor diagonal $d(v)$ de todos os vértices é $d(v) = \alpha$.



$$\begin{bmatrix} \alpha & 1 & 1 & 0 & 0 \\ 1 & \alpha & 0 & 1 & 1 \\ 1 & 0 & \alpha & 0 & 0 \\ 0 & 1 & 0 & \alpha & 0 \\ 0 & 1 & 0 & 0 & \alpha \end{bmatrix}$$

Figura: Árvore do exemplo com enumeração válida e sua matriz B_α.

Algoritmo

A principal característica do algoritmo é que pode ser **executado diretamente na árvore**, procedendo das folhas para a raiz.

Aplicando o algoritmo (das folhas para a raiz), temos então dois casos, se todos os filhos v_i do vértice v_j possuem $d(v_i) \neq 0$ e se algum dos filhos de v_j possui $d(v_i) = 0$.

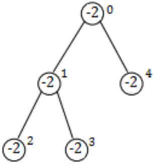


Figura: A execução do algoritmo para $\alpha = 2$ recai no primeiro caso.

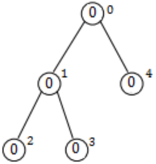


Figura: A execução do algoritmo para $\alpha = 0$ recai no segundo caso.

Algoritmo

No primeiro caso a entrada de $d(v_j)$ recebe $d(v_j) - \sum_{c \in \mathcal{C}} \frac{1}{d(c)}$, onde $\mathcal{C}$ é o conjunto dos vértices filhos de v_j.

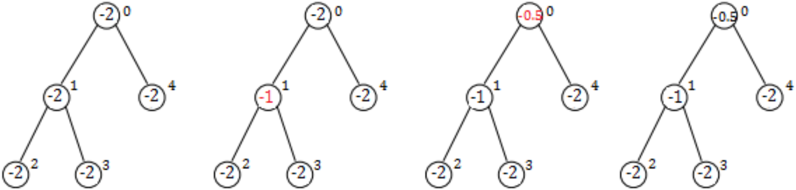


Figura: A execução do algoritmo para $\alpha = 2$. Temos que todos autovalores são menores que 2.

Algoritmo

O segundo caso é mais complexo. Quando um ou mais filhos de v_j possui $d(v) = 0$, então escolhe-se um v_i filho de v_j com $d(v_i) = 0$ e atribuímos então $d(v_j) = -\frac{1}{2}$ e $d(v_i) = 2$. Após isso, desconectamos v_j de seu pai, caso não seja a raiz, e continuamos a execução do algoritmo.

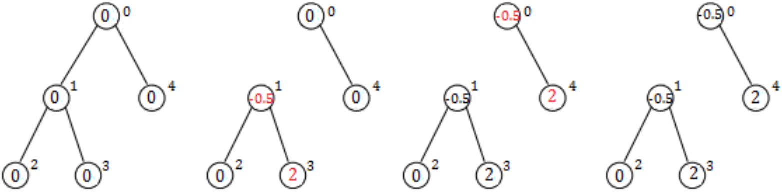


Figura: A execução do algoritmo para $\alpha = 0$. Temos dois autovalores menores, dois maiores e um igual a 0.

Algoritmo

O Teorema 1 fornece um meio de utilizar o algoritmo de Jacobs e Trevisan para verificar quantos autovalores há em um intervalo qualquer.

No trabalho, foi implementado na linguagem C o algoritmo para intervalos ou para um valor α dado. Além disso, foi implementado o algoritmo aplicado à matriz, mostrando passo a passo do que ocorre durante a diagonalização.

Algoritmo

Input

Como input, o programa recebe a matriz de adjacência, através de um arquivo de texto. Após isso, o usuário escolhe como deseja executar o algoritmo.

```
Digite o nome completo do arquivo com os dados da matriz:
arvore_ex.txt
```

Algoritmo

Output

O output para a execução do intervalo é o seguinte:

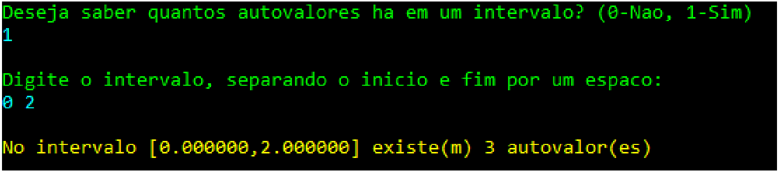


Figura: O espectro do grafo do exemplo é $[-1.84776, -0.76537, 0, 0.76537, 1.84776]$

Algoritmo

Output

O output da execução para um valor de α é o seguinte:

```
Digite o valor de alpha:
0

A arvore dada possui:
2 autovalor(es) maior(es) que 0.000000
2 autovalor(es) menor(es) que 0.000000
0.000000 e autovalor com multiplicidade 1
```

```
Digite o valor de alpha:
2

A arvore dada possui:
0 autovalor(es) maior(es) que 2.000000
5 autovalor(es) menor(es) que 2.000000
```

Figura: O resultado corrobora o que foi obtido acima.

Agradecimentos

Professores: Carlos Hoppen, Luiz Emílio Allem e Vilmar Trevisan.

Referências

D. P. Jacobs and V. Trevisan
Locating the Eigenvalues of Trees
Elsevier, 2001.

Reinhardt Diestel
Graph Theory (2^{nd} edition)
Springer-Verlag, 1:2–26, 12:251–277, 2000.

N. Abreu, R. Del-Vecchio, V. Trevisan, C. Vinagre.
Teoria Espectral de Grafos – Uma Introdução.
III^o Colóquio de Matemática da Região Sul., 6:145–172, 2014.

Muito obrigado!

$$B_\alpha v = (A + \alpha I)v = \beta v \Rightarrow (\lambda + \alpha)v = \beta v$$

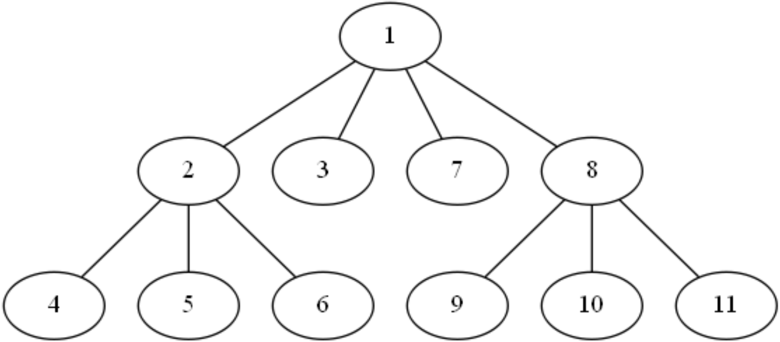


Figura: [-2.44949, -1.73205, -1, 0, 0, 0, 0, 0, 1, 1.73205, 2.44949]

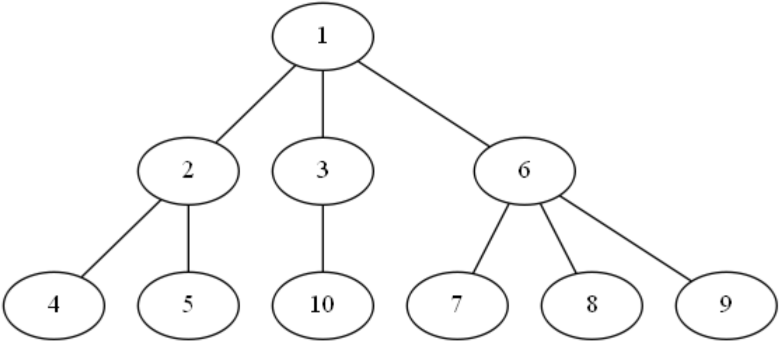


Figura: [-2.28349, -1.56870, -1.15103, 0, 0, 0, 0, 1.15103, 1.56870, 2.28349]