

SME0130 – Redes Complexas

Projeto 02 – Caracterização de Redes Complexas

Felipe Scrochio Custódio – 9442688

Gabriel Henrique Scalici – 9292970

Introdução

O projeto consiste da análise de seis redes distintas (retiradas do site: <http://konect.uni-koblenz.de/networks>), onde serão calculadas diversas medidas vistas em aula, para que possam ser melhores classificadas e compreendidas, como por exemplo a assortividade e modularidade.

Os dados estão organizados em tabelas e gráficos gerados pelo programa que facilitam a compreensão da rede.

O projeto foi desenvolvido utilizando a linguagem Python e as bibliotecas *Igraph*, *NetworkX*, *Matplotlib*.

Foram utilizadas as duas bibliotecas para a construção das redes (*Igraph*, *NetworkX*) realizando a conversão de uma para outra quando necessário, como por exemplo para calcular a modularidade, onde *Igraph* possui funções que não estão presentes no *Networkx*.

Redes estudadas:

- EuroRoad
- US Airports
- Hamsterster
- Cortical - Humano
- Cortical - Gato
- Cortical - Macaco

Os códigos desenvolvidos para o projeto, assim como os gráficos na resolução original, estão disponíveis no repositório:

<https://github.com/felipecustodio/complex-networks/blob/master/assignment2/assignment2.py>

1 – Assortatividade

Rede	Coefficiente de Assortatividade
EuroRoad	0.900
US Airports	-0.1134
Hamster	-0.0889
Cortical Human	0.2910
Cortical Cat	-0.0917
Cortical Monkey	-0.1506

Sendo assortividade a tendência de um vértice se conectar com outro vértice de grau parecido, temos que a rede pode ser:

- Assortativa : Nós de grau elevado conectados à nós de grau elevado e nós de grau baixo conectados à nós de grau baixo.
- Disassortativa : O oposto de assortativa, onde nós de grau elevado tendem a se conectar com nós de grau baixo.

Pelo coeficiente de assortividade é possível verificar que:

- Assortativa : EuroRoad e Cortical Human
- Disassortativa : USAirports, Hamster, Cortical Cat e Cortical Monkey

Há uma relação entre a assortividade de o tipo de rede, visto que nas redes com coeficiente positivo como "*EuroRoad*" e "*Cortical Human*" há semelhança de conexão de um vértice com os graus dos vértices vizinhos, tendendo a formar mais comunidades, o que de fato acontece e pode ser visto nos cálculos e discussões dos próximos itens.

2 - Correlação entre k X $k_{nn}(k)$

Calculamos o Coeficiente de Pearson entre os vértices e seus knns ($k_{nn}(x)$) utilizando `average_neighbor_degree` da biblioteca NetworkX.

Rede	Coeficiente de correlação
EuroRoad	0.1071
US Airport	-0.0495
Hamster	-0.0456
C. Human	0.4505
C. Cat	-0.1399
C. Monkey	-0.3913

A primeira relação que podemos facilmente perceber nas tabelas de coeficiente de correlação e de assortatividade é que as redes analisadas tiveram valores positivos nas duas ou valores negativos nas duas.

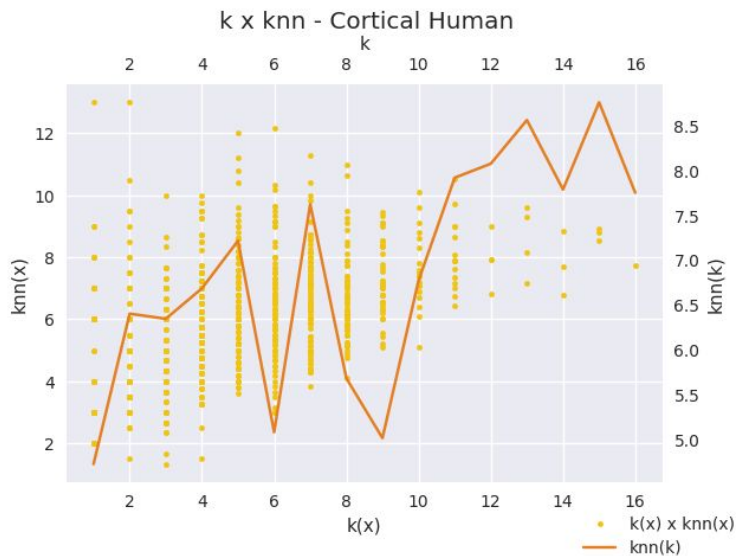
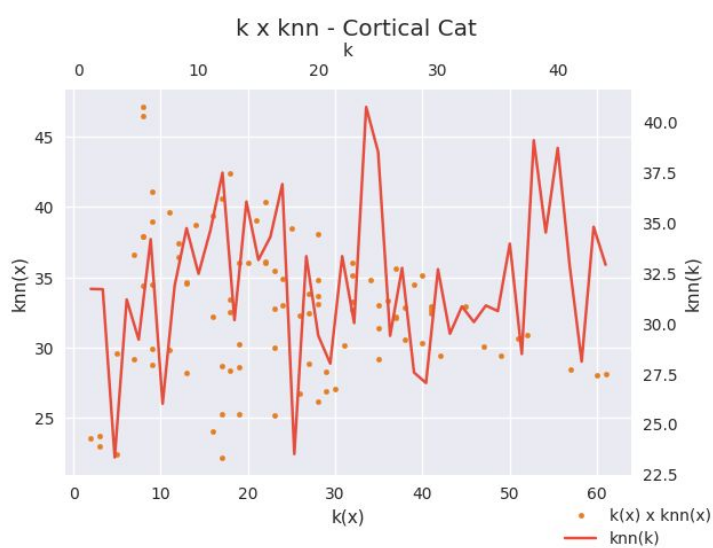
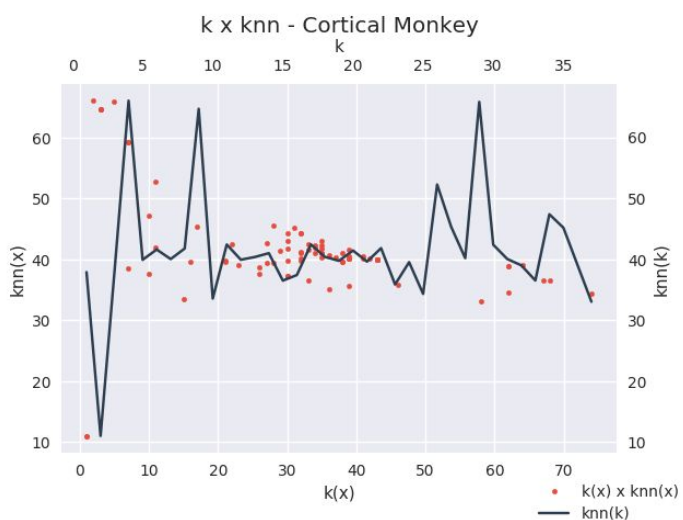
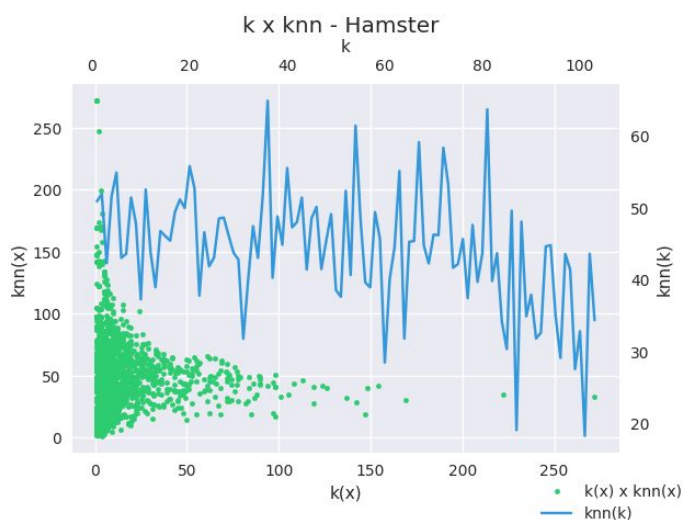
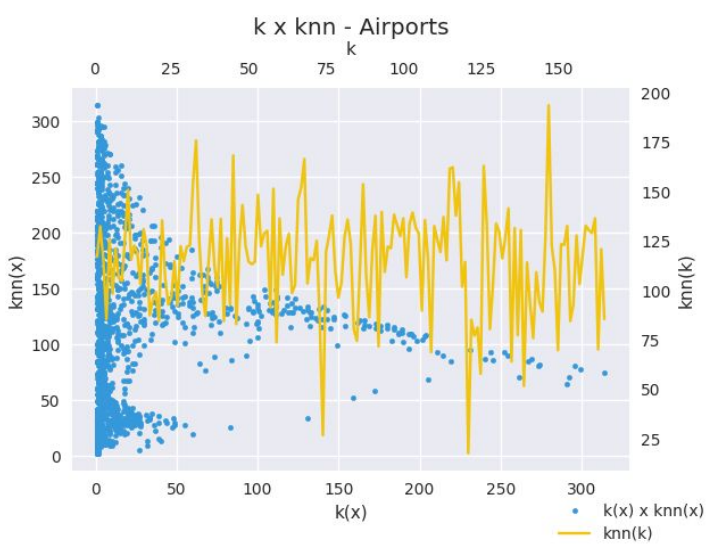
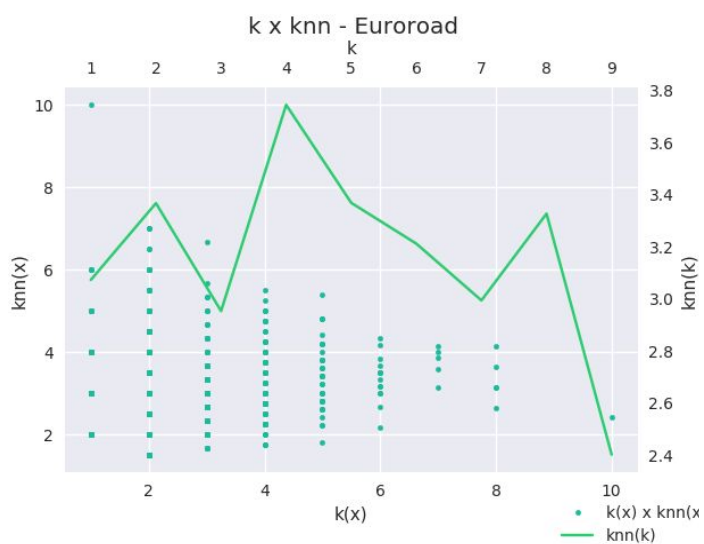
Essa relação entre as duas medições podem ser explicada pois das duas formas é possível obter a informações da semelhança de um vértice com os vértices vizinhos, podendo ajudar na visualização e funcionamento da rede, mostrando inclusive a tendência de haver comunidades (que serão explicadas mais adiante).

Analisando as curvas (k_{nn} por grau) do gráfico e o eixo x , é possível ver uma grande diferença entre as redes descritas, como pode ser observado por exemplo da "EuroRoad" e "Airports" onde o

eixo x da primeira vai até o grau 10 e o eixo x da segunda vai até o grau 300, o que mostra a quantidade de ligações que cada vértice pode assumir e até mesmo a presença de comunidades.

Nos itens seguintes trabalharemos mais com a questão das comunidades em uma rede, por enquanto temos que o grau médio em redes que possuem comunidades como *EuroRoad* e *Cortical Human* é relativamente baixo se comparado com redes como *Cortical Monkey*, *Cortical Cat* e *USAirports* e *Hamster* que não possuem comunidades.

Os gráficos abaixo estão armazenados na pasta "*plots*" do trabalho para que possam ser visualizados com maior qualidade.



3 - Modularidade

Rede	Bet. Centrality	Fast-Greedy	Eigenvectors of matrices	Walktrap
EuroRoad	0.8701	0.8660	0.8378	0.8237
US Airport	0.2199	0.3149	0.2772	0.2255
Hamster	0.3203	0.4072	0.3977	0.3737
C. Human	0.8276	0.7824	0.7851	0.8064
C. Cat	0.0229	0.2605	0.2231	0.1903
C. Monkey	0.3141	0.3552	0.3556	0.3470

Em todas as redes analisadas, as medições de modularidade de uma forma geral são bastante similares, isto é, o resultado de uma função de modularidade não se diferencia tanto de outra medição para a mesma rede. É perceptível uma mudança de desempenho em relação a cada algoritmo de detecção, o que será visto mais a fundo no item de comunidades.

Os resultados mostram por exemplo que na rede de rodovias "*EuroRoad*" é mais comum que se tenha comunidades, o que faz muito sentido pois existem mais rodovias (arestas) que vão para cidades (vértices) próximas do que para cidades distantes, o que já não ocorre em redes como "*USAirports*", pois aeroportos distantes estão ligados da mesma forma que aeroportos próximos, o que evita a formação de comunidades.

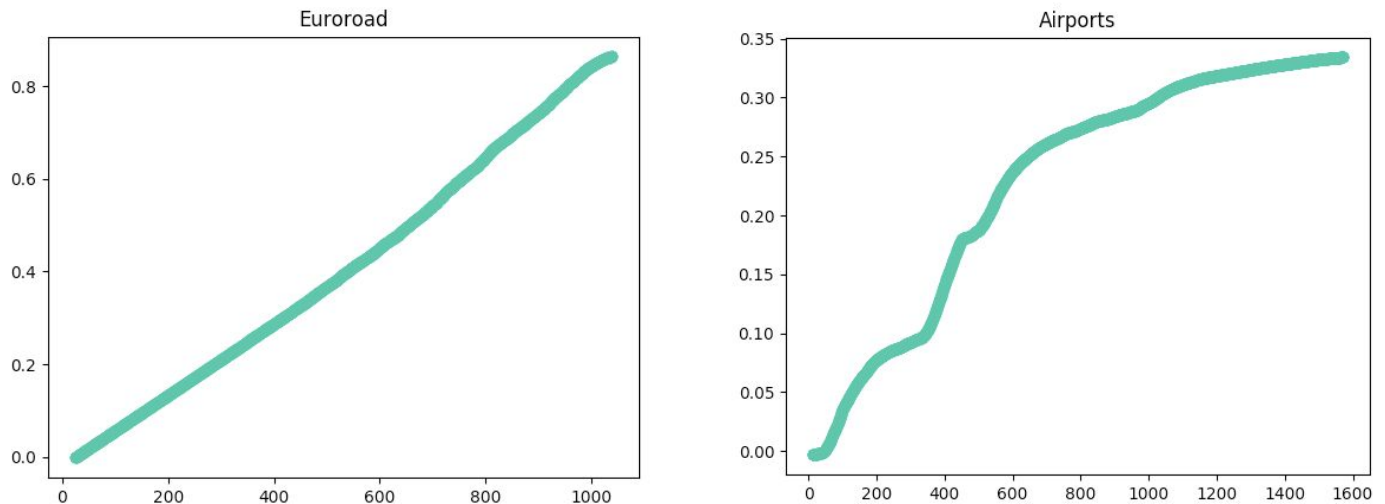
No **item 6**, focamos nas diferenças entre as **redes corticais** (Gato, Humano e Macaco).

Podemos observar que as redes corticais de seres humanos possuem a maior modularidade. Podemos relacionar o maior número de comunidades com um maior número de possíveis sinapses, o que aumenta muito a capacidade de aprendizado e processamento de informações nos cérebros humanos, comparado com macacos e gatos, com valores muito menores de modularidade.

Em termos de performance, o algoritmo de Edge Betweenness Centrality demorou cerca de 2h para a rede Hamster e 1:30h para a rede Airports, por serem redes muito grandes e muito conectadas.

4 – Evolução da modularidade (Fastgreedy)

4.1 – USAirports e EuroRoads

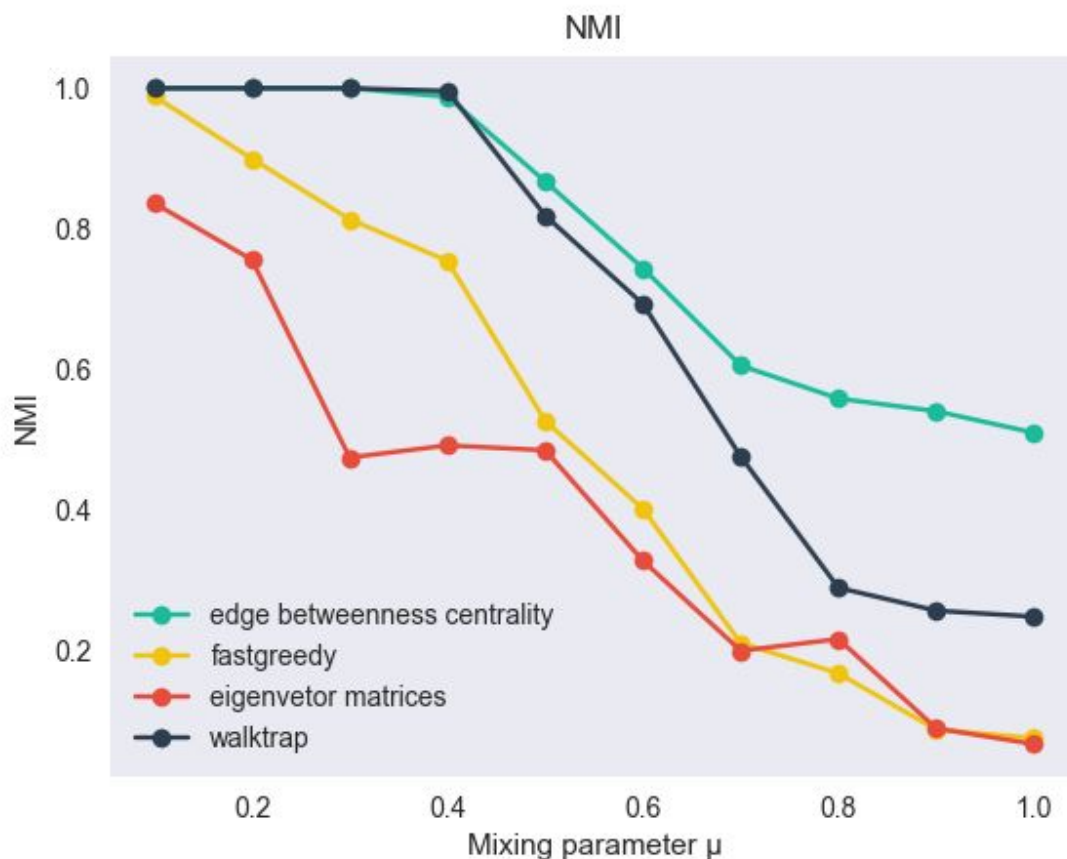


Os gráficos "*EuroRoad*" e "*USAirports*" apresentam apenas um pico, que corresponde ao valor de modularidade obtido na tabela do item anterior.

Lembrando que o algoritmo *fastgreedy* do iGraph que foi utilizado para obter a informação da evolução da modularidade a cada interação armazena em seu histórico somente valores anteriores ao máximo global.

Analisando com mais cuidado os dois gráficos, temos que a evolução da modularidade em *EuroRoad* chega ao pico em bem menos passos que na rede *USAirports*.

5 – Comunidades



Geramos as redes incrementando o mixing parameter de 0.1 em 0.1, indo de 0.1 a 1.0.

Para cada rede, pegamos as comunidades do arquivo *communities.dat*, gerada pelo programa do pacote *binary_networks*. Esse arquivo contém a informação real de quais comunidades existem e quais vértices estão nela.

Para avaliar e visualizar o desempenho dos algoritmos de detecção usados no passo acima (modularidade), rodamos todos eles para cada rede e calculamos o NMI, para ver o quão precisamente esses algoritmos conseguem detectar essas comunidades.

Como pode ser observado pelo gráfico, a curva que menos sofre queda e mantém maior precisão é a de *betweenness centrality*, sendo dessa forma a mais precisa para análise de comunidades em relação às quatro formas testada.

As menos precisas, isto é, que sofrem mais queda no NMI são *Eigenvetor matrices* e *Fast-greedy*, que atingem o mesmo valor de aproximadamente 0.5 NMI da curva mais precisa, quando o "*mixing parameter*" está em um valor aproximado de 0.4. Para redes muito densas e conectadas, com muitas comunidades, não seriam boas escolhas como algoritmos de detecção. Eles, porém, rodam muito mais rápido.

Podemos ver claramente que o custo de uma precisão maior vêm em termos de eficiência, com o algoritmo de *betweenness centrality* chegando a demorar horas para as redes maiores, comparado com minutos dos outros algoritmos.