# Anotaciones del proyecto de redes HDN

Objetivo

El objetivo de este trabajo es analizar la read Human Disease Network (HDN). En primer lugar, realizaremos un análisis exploratorio de la red (valores de su parámetros) y la compararemos con un conjunto de redes aleatorias. Aquí añadiremos los parámetros sigma y omega que tiene network x para decir si es una red aleatoria o si es de mundo pequeño. Concluir con qué tipo de red es.

Luego analizaríamos los nodos más importantes según esos parámetros. Por ejemplo, según la cercanía, la intermediación, los k cores etc.

El siguiente paso sería realizar diferentes ataques (me refiero a distintas aproximaciones de ataques) empezando por distintos nodos. El objetivo final es ver como de resistente es la red de enfermedades y cuál es el nodo más susceptible. Para ver como de resistente es, atacar también a la red aleatoria.

En este trabajo tenemos varias cosas.

* Un archivo .gexf (gephi) y un archivo .gml (cytoscape) para ver la red completa. Esta red tendría tanto genes como enfermedades como nodos. Esto es muy importante, porque nosotros queremos estudiar solo los nodos que sean enfermedades. Los nodos (enfermedades) están conectados si las enfermedades comparten un gen.
* El archivo “diseasome.edgelist.txt” o el “diseasome.edgelist.csv” son los mismo, solo que la extensión se cambia para que Gephi pueda leerlo. Tiene únicamente los nodos. En total 516 nodos. Estos serían los nodos de las enfermedades.
* El archivo “disease.label.txt” que contiene las etiquetas de las enfermedades.

#### Análisis inicial del grafo

Entre las cosas que quiero analizar es:

* Identificar los k-cores de 10 y 9, viendo que tipo de enfermedad son, el tamaño de los nodos me servirá para ver la facildad de curarla (nº genes)
* Identificar el camino más corto entre los k-cores.
* Identificar los nodos más importantes: mayor grado, mayor centralidad, mayor cercanía, mayor betweenness
* Hacer ataques a la red a los distintos nodos identificados anteriormente y ver cuál sería el óptimo para curar las enfermedades. Podemos seguir el plantemiento del CRISPR. El objetivo sería ver qué enfermedad me permitiría llegar más rápido a las demás y curar más rápidamente enfermedades.
* Una razón por la que empezar desde un nodo del k core 10 es porque el k core 10 está cerca del k core 9 y más cerca que éste del k core 7. Desde el k core 10 dominamos todo el grafo. Además, dentro del k core 10 hay varios grafos con propiedades de cluster, cercanía y betweenness altos.
* Tengo foto de los k cores 10, 9 y 7. También del camino más corto entre ellos.
* EVOLUCIÓN DE K-CORE: voy a dibujar como va cambiando el k core en el grafo. Crearlo en python y guardar eso como imagen ya montada.
* Por otro lado está estudiar que la red de enfermedades no es una red aleatoria. Para eso represento, el diagrama de k cores, la distribución de grado, la distribución del índice de clusterización. Los valores de ambos grafos y su comparación.
* Con distintos parámetros calculados saco el nodo top 1 de cada parámetro y estudio el ataque desde ese nodo. Objetivo: ver qué características son más importantes a la hora de infectar una red.
  + Empezar a atacar desde el nodo que es baricentro y centro: 736, Lipodistrofia (que esté en el centro no significa que esté conectado con todos)
  + Desde un nodo en la periferia: 829, 828, 962
  + Nodo con indice de clusterizacion 1
  + Nodo de k core 7
  + Nodo de k core 9
  + Nodo de k core 10
  + Nodo con mayor betweenness
  + Nodo con mayor centralidad de grado
  + Nodo con mayor cercanía

Vemos que el baricentro no es un nodo que aparezca en los k cores 7, 9 o 10. Pertenece a un k core de 4, tiene grado 8.

Vemos que en la periferia encontramos nodos pertenecientes al k core 9. El k core 9 no es una buena idea para empezar a atacar.

Vemos que el k core 10 está cerca de nodos con alta centralidad y tiene a colon cancer con mayor centralidad de grado. Éste también es el que mayor grado posee. Tiene un índice de clusterización igual a 0.124 pero el que más triángulos posee (152). La distancia del K core 10 al k core 7 es pequeña e intermedia al k core 9. Mientras que del k core 7 al 10 es corto pero al 9 es más largo. Y lo mismo pasa con el k core 9. El k core 10 se encuentra en el medio de los dos k cores 7 y 9. No es el nodo

#### Características de HDN

* Diámetro: 15
* Radio 8
* Conexo
* No dirigido
* Densidad: 0.0089
* Grado medio del grafo: 4.6047
* Orden: 516
* Tamaño: 1188
* Camino característico: 6.508
* Índice de clusterización: 0.6358
* Transitividad del grafo: 0.4305
* Máximo número de triángulos: 152
* Sigma (nx.sigma(hdn, niter = 10, nrand = 3)) = 10.058
* Omega (default) = - 0.719
* Non randomness: (186.3187863371036, -32.388568503923956)

#### Características del grafo aleatorio

* No conexa
* Componente: 6.25 +- 2.2
* Diámetro: 8.6 +- 0.58
* Densidad: 0.00894 +- 1e-18
* Radio: 5.65 +- 0.477
* Transitividad: 0.0085 +- 0.0023
* Triángulos: 2.2 +- 0.599
* Centralidad de grado: 0.0089 +- 5e-18
* Nodo con mayor centralidad de grado: 0.0238 +- 0.00228
* Cercanía: 0.233 +- 0.00163
* Nodo con mayor cercanía: 0.286 +- 0.00163
* Intermediación: 0.0062 +- 9e-5
* Nodo con mayor intermediación: 0.0344 +- 0.0048
* Clustering: 0.00761 +- 0.0028
* Nodo con mayor clustering: 0.475 +- 0.00279
* Max k core: 3 +- 0
* Camino característico: 4.234 +- 0.0269
* Sigma (nx.sigma(hdn, niter = 10, nrand = 3)) = 1.6743
* Omega (nx.omega(hdn, niter = 10, nrand = 3)) = 0.8008

Ataques

El aspecto que presenta nuestro grafo es el de una mezcla de small-world, sclae-free pero con algo más de regular. Lo apoya también el valor que sacamos en omega, que está cercano a 1 y eso son mallas. Por eso vemos que somos sensibles a ataques aleatorios también, porque en cuento nos golpean un grafo con muchos vecinos lo notamos, pero no se desploma del todo.

##### Modelos SIS y SIR de infección

En el modelo SIR hay tres estados: infectado, sano y recuperado. En el modelo SIS sólo hay dos fases: infectado y sano. En el SIS no hay inmunidad.

En nuestro caso nos da igual simular el SIR o SIS puesto que la tasa de recuperación es 0.

Quiero probar la