Universidad Técnica Federico Santa María
Departamento de Informática

Redes Complejas
Primer semestre de 2024

Tarea 2

Observaciones:

- Una pregunta requiere un par de archivos; en Aula encontrará el link a la carpeta que los contiene.
- En un par de preguntas tendrá que generar algo de código. **Inclúyalo** en su entrega.
- Dondequiera que vea un "comente" o "interprete", en esta tarea y las que sigan hágalo. Nos es de yapa; por lo general es lo más importante de una pregunta.

1. **[12 pt]** Sea G el grafo con matriz de adyacencia

   0 1 1 0 0 0 0
   1 0 1 0 0 0 0
   1 1 0 1 0 0 0
   0 0 1 0 1 1 0
   0 0 0 1 0 1 0
   0 0 0 1 1 0 1
   0 0 0 0 0 1 0

   Escriba la matriz laplaciana y determine el valor de Fiedler, junto al vector propio asociado. Grafique la red incluyendo los valores del vector propio, y úselo para determinar la (bastante evidente) partición de la red en dos comunidades.

2. **[15 pt]** Considere un grafo aleatorio con una gran cantidad de nodos (piénselo como el límite con $n \to \infty$) y con una distribución de grados dada por $P(k) = C \times \alpha^k$, para $k \geq 0$, donde $0 < \alpha < 1$ y C es una constante de normalización.

   (a) Dé una expresión cerrada para C en función de α (para que P efectivamente sea una distribución de probabilidad).
   (b) Dé una expresión cerrada para la función generadora de la distribución de grados.
   (c) Determine la condición sobre α que hará que el grafo tenga o no una componente gigante.

   ("Cerrada" aquí se refiere a que no sea una sumatoria, sino una expresión directa).

3. **[24 pt]** Para esta pregunta consideraremos cuatro redes: las de los archivos "gnutella.gdf" y "delfines.gml", y además un par de redes Erdös-Renyi que usted debe generar con la misma cantidad de nodos y la misma densidad de aristas.

   Ambas son redes no orientadas; la primera es un fragmento de la red p2p Gnutella, años ha, mientras que la segunda corresponde a la red de interacciones sociales de una comunidad de delfines en Nueva Zelandia.

   Ataque a piratas y delfines de las siguientes tres maneras:

   - Eliminando nodos al azar
   - Eliminando nodos en orden de grado decreciente
   - Eliminando nodos en orden de betweenness decreciente

Ojo: vaya recalculando grados y betweenness en la medida que vaya eliminando nodos.

Para cada red y para cada modo de ataque, determine el porcentaje de nodos que hace falta eliminar para que la componente gigante caiga a $1/2$ de su tamaño inicial.

Comente sus resultados.

4. **[15 pt]** Genere un grafo ER de 200 nodos, con probabilidad de conexión 0.2. Si llamamos a los nodos $\{a_1, a_2, \ldots, a_{200}\}$, entonces definamos ahora una partición de los nodos en dos grupos como $B_1 = \{a_1, \ldots, a_{100}\}$ y $B_2 = \{a_{101}, \ldots, a_{200}\}$, y otra partición, también en dos grupos, como $C_1 = \{a_1, \ldots, a_{50}, a_{101}, \ldots, a_{150}\}$ y $C_2 = \{a_{51}, \ldots, a_{100}, a_{151}, \ldots, a_{200}\}$.

   Convierta ahora las aristas en arcos orientados, escogiendo al azar cuál punta es cuál, excepto en el caso de las aristas entre B_1 y B_2 : esas oriéntelas con probabilidad p desde B_1 hacia B_2 (y con probabilidad $1 - p$ en dirección contraria).

   Repita esto para p desde 0 hasta 1, con pasos de 0.1 (o sea, p=0, 0.1, 0.2, 0.3, ..., 1). En cada ocasión evalúe la medida de modularidad Q^d, esto es, la medida modificada por Newman y Leicht para aplicarla a redes dirigidas[1]. Evalúela sobre las dos posibles particiones: (B_1, B_2), y (C_1, C_2). Grafique sus resultados (deberían ser dos curvas, cada una con 11 puntos), e interprételos[2].

   Nota: para evitar mucho ruido en los resultados, tome el promedio de los $Q^d(p)$ sobre una cantidad grande de realizaciones del experimento (sugiero al menos 50).

5. **[9 pt]** Considere una red regular de grado k (es decir, una red en la que todos los nodos tienen el mismo grado, k). ¿Cuánto tiene que valer k para que, bajo fallas al azar, la componente gigante siga existiendo incluso si se borra el 95% de los nodos?

6. [15 pt] Baje la red "redchica.gdf".

   (a) Grafíquela, indicando junto a cada nodo su grado de entrada, su betweenness, y su valor de PageRank.

   (b) Haga un ranking de los nodos en función de cada uno de esos tres índices.

   (c) Comente sobre las posibles correlaciones entre esos valores, y sus divergencias. P. ej., ¿hay nodos con mejor PageRank que el esperable por su grado de entrada? ¿O con peor PageRank? ¿Y qué hay de su betweenness? Interprete, a la luz de la red, el por qué de esas discrepancias.

7. **[20 pt]** Baje "scientometrics.net". Se trata de citaciones entre papers en el journal *Scientometrics* a lo largo de dos décadas. Por lo tanto se trata de una red dirigida, y más aún, prácticamente sin ciclos, pues un paper sólo cita (por lo general) a papers cronológicamente previos.

   Determine la abundancia de todos los posibles motifs triangulares. Compare con los resultados de Milo et al que vimos en clases. ¿Cuáles parecen estar sobre/sub representados?

   Determine el vector de clustering dirigido definido por Ahnert & Fink, como lo vimos en clases. Comente sus resultados.

8. **[15 pt]** Implemente el siguiente sistema de generación de redes: Comenzamos con una malla regular de $N \times N$ nodos, donde cada nodo está conectado a sus 4 vecinos inmediatos (izquierda, derecha, arriba y abajo), si es que existen (en los bordes y esquinas faltarán vecinos). Luego, para cada nodo, escogemos un vecino extra mediante el siguiente procedimiento: partimos del

[1] Me da la impresión de que esto *no* está implementado en NetworkX (corríjanme si me equivoco). Pueden implementarla ustedes mismos, o intentar reciclar esfuerzos de terceros, como este (que no he probado): `https://zhiyzuo.github.io/python-modularity-maximization/`.

[2] De hecho, recomiendo iniciar la interpretación antes de hacer el experimento: piensen en cómo *deberían* ser las curvas. Así se darán cuenta si los resultados están saliendo mal. Es más: es posible -y legítimo- derivar analíticamente lo que debería salir, y ahorrarse el experimento.

nodo y damos un paso en una dirección al azar (izq, der, arriba, abajo). Con probabilidad p escogemos el nodo alcanzado como nuevo vecino (si ya era vecino, queda vecino); con probabilidad $1-p$, damos otro paso y aplicamos la misma regla. Por lo tanto, habrá una probabilidad $p(1-p)^k$ de que tengamos que dar $k+1$ pasos hasta escoger nuestro vecino.

Nos interesa estudiar la relación entre p y la distancia promedio entre los nodos de la red. Para esto, genere redes con $N = 50$ (o mayor, mientras el experimento sea viable en su PC) y experimente con distintos valores de p, calculando en cada caso la distancia promedio $\langle d \rangle$ entre los nodos. Produzca un gráfico p versus $\langle d \rangle$, para un rango de p en el que se note el cambio entre un mundo "grande" y un mundo pequeño.

Bonus, +5pt: Interprete sus resultados en comparación con los descritos por Kleinberg en `http://www.cs.cornell.edu/home/kleinber/nat00.pdf` (en lo que concierne a cuándo el mundo es pequeño o no, no miraremos la navegabilidad). Note que por el procedimiento usado, p no es directamente equivalente al parámetro usado allí.

Sí, suma más de 100.