

CAIM LAB7

Yimin Pan

December 2019

1 Introducción

En esta práctica aprenderemos a analizar redes, así como algunas propiedades como diámetro, transitividad y distribución de grado de los nodos... Haciendo uso de la librería **iGraph** en Python.

2 Tarea 1

En la primera parte de la práctica se nos pidió generar una gráfica, la cual el eje X representa la probabilidad de **rewiring**, y el eje Y el coeficiente de **clustering** y **average shortest-path (asp)** de los grafos generados con dicha probabilidad. Para poder visualizar estas dos propiedades en una misma gráfica normalizamos los valores de estos, para así tenerlos en $[0, 1]$, y el eje X lo pasamos a escala logarítmica.

En la gráfica del enunciado, vemos que los valores de p están entre 0.0001 y 1, y son 14 puntos equidistantes en escala logarítmica. Por lo que creamos una lista con estos valores. Luego generamos los grafos **Watts Strogatz** con dichas p .

Después de hacer unos cuantos experimentos, observamos que para redes de este tipo, cuando incrementamos la p de 0 a 1, tenemos un decremento rápido en **asp** y un bajo decremento en **transitividad**. Dicho esto, hay otros dos parámetros que influyen en el modelo de **Watts Strogatz**, el número de vértices y las conexiones iniciales del anillo.

Para el número de conexiones iniciales, si escogemos un número muy alto, vemos que la velocidad de decremento del **asp** se reduce, la razón es que al inicio el grado de los vértices ya eran muy altos por lo que el **shortest-path** no se ve afectado por los **rewirings**.

En cuanto al número de vértices, cuando éste es grande, estabilizamos la variabilidad producida por los modelos aleatorios. Si generamos dos grafos WS con los mismos parámetros podemos obtener resultados bastante diferentes cuando el número de vértices es pequeño.

Dicho esto, escogemos $n = 5000$ y número de vecinos = 4, con lo que obtenemos el siguiente gráfico:

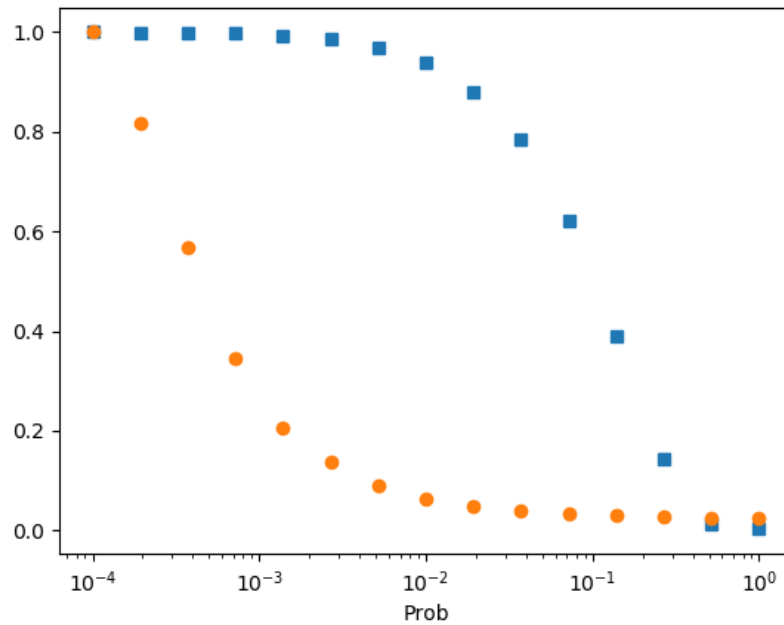


Figure 1: Gráfica asp/transitividad - p

3 Tarea 2

En la segunda sección se nos pidió cargar un grafo no dirigido desde el fichero `edges.txt` y luego estudiar algunas de sus propiedades

- num. aristas: 602.
- num. vertices: 62.
- diámetro: 2.
- transitividad: 0.55.
- grado de nodos: [63, 13, 9, 39, 35, 7, 9, 59, 13, 25, 13, 11, 33, 31, 19, 21, 27, 7, 23, 33, 7, 9, 13, 35, 37, 17, 13, 13, 17, 9, 35, 23, 25, 33, 31, 3, 21, 11, 11, 13, 21, 11, 23, 29, 43, 15, 15, 23, 15, 7, 15, 15, 7, 19, 7, 15, 7, 9, 15, 17, 3, 7].
- distribución de grado:

```

degree distr. : N = 62, mean +- sd: 19.4194 +- 12.5300
[ 3, 4]: ** (2)
[ 4, 5]: (0)
[ 5, 6]: (0)
[ 6, 7]: (0)
[ 7, 8]: ***** (8)
[ 8, 9]: (0)
[ 9, 10]: ***** (5)
[10, 11]: (0)
[11, 12]: **** (4)
[12, 13]: (0)
[13, 14]: ***** (7)
[14, 15]: (0)
[15, 16]: ***** (7)
[16, 17]: (0)
[17, 18]: *** (3)
[18, 19]: (0)
[19, 20]: ** (2)
[20, 21]: (0)
[21, 22]: *** (3)
[22, 23]: (0)
[23, 24]: **** (4)
[24, 25]: (0)
[25, 26]: ** (2)
[26, 27]: (0)
[27, 28]: * (1)
[28, 29]: (0)
[29, 30]: * (1)
[30, 31]: (0)
[31, 32]: ** (2)
[32, 33]: (0)
[33, 34]: *** (3)
[34, 35]: (0)
[35, 36]: *** (3)
[36, 37]: (0)
[37, 38]: * (1)
[38, 39]: (0)
[39, 40]: * (1)
[40, 41]: (0)
[41, 42]: (0)
[42, 43]: (0)
[43, 44]: * (1)
[44, 45]: (0)
[45, 46]: (0)
[46, 47]: (0)
[47, 48]: (0)
[48, 49]: (0)
[49, 50]: (0)
[50, 51]: (0)

```

Figure 2: degree distribution

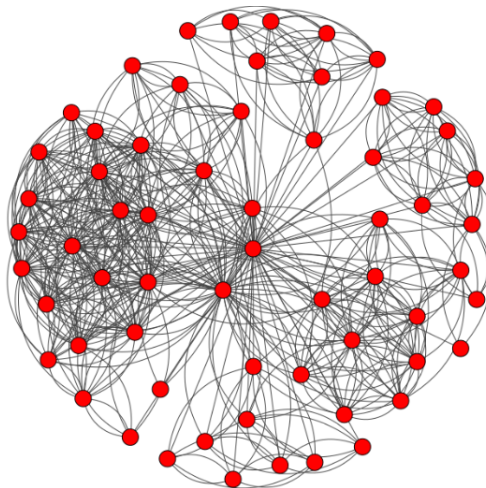


Figure 3: edges.txt

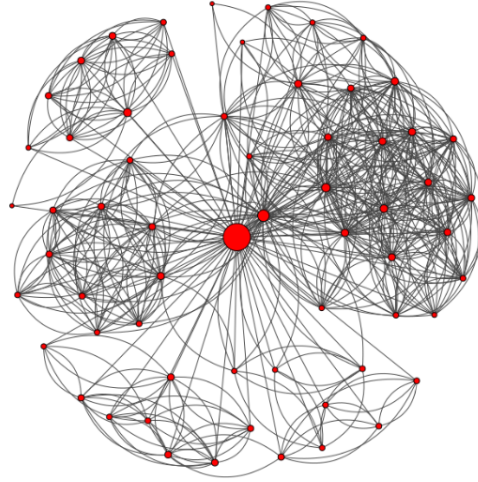


Figure 4: pagerank.txt

Al observar la distribución de los grados y el pagerank, nos da la impresión de que no se trata de un grafo aleatorio. En la gráfica de PR vemos que tenemos un nodo central que hace de **hub**, su grado es mucho mayor que los demás, y esta propiedad lo vemos en modelos de **Barabási-Albert** donde la probabilidad de que un nodo nuevo se conecte a otro ya existente es proporcional a su grado. De este modo, tenemos unos pocos nodos con grado elevado y los otros siguiendo una distribución uniforme.

3.1 Detección de Comunidades

Para este experimento, usamos el algoritmo de clustering `community_edge_betweenness()` para la detección de comunidades.

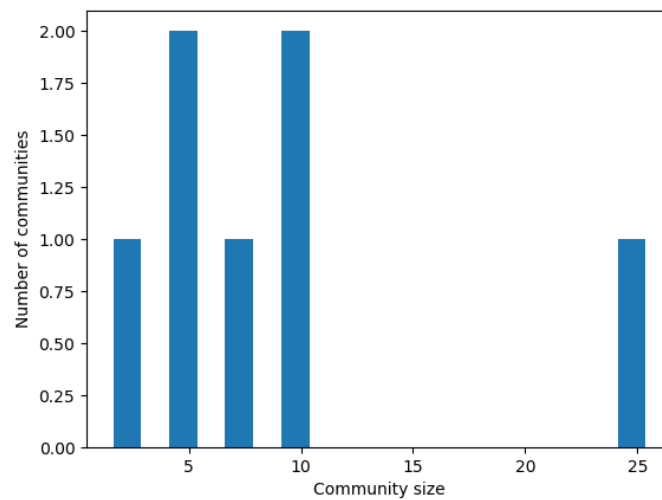


Figure 5: histograma de tamaño de clusters

Desde el histograma podemos ver que el tamaño de la comunidad más grande es 25. Los otros clusters, tienen un tamaño alrededor de 5 o 10, siendo en total 7 clusters.

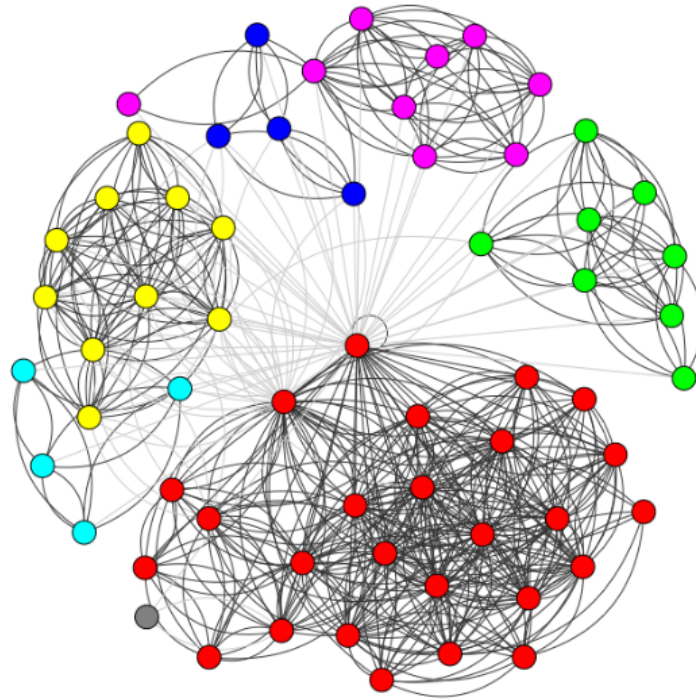


Figure 6: histograma de tamaño de clusters

En este plot, donde cada comunidad se representa con un color diferente podemos ver también lo dicho anteriormente. En realidad, desde el grafo original tampoco es difícil intuir esta distribución de clusters.

4 Problemas

A lo largo de la práctica no hemos encontrado grandes dificultades. Aunque nunca habíamos usado antes **iGraph**, el notebook ya era bastante completo como para cubrir lo que se pide. En comparación con prácticas anteriores, en esta, básicamente es hacer "prints" y "plots", no hay mucho que pensar en el código.