

UNIVERSITAT POLITÈCNICA DE CATALUNYA

BÚSQUEDA Y ANÁLISIS DE INFORMACIÓN MASIVA

GRADO EN INGENIERÍA INFORMÁTICA

Introduction to igragh

Entregable 7

Autor:

Eric DACAL

Tutor

Jose Luis BALCAZÁR NAVARRO

Q1 Curs 2018-2019



**UNIVERSITAT POLITÈCNICA
DE CATALUNYA
BARCELONATECH**

1 Tarea 1: Reproducción de un grafo

El objetivo de esta tarea es la creación de un gráfico visto en teoría, en este caso correspondiente a la representación de el coeficiente de clustering y el camino mas corto promedio en función del parametro p del Modelo Watts y Strogatz.

Para este objetivo he creado el script de R *task1.R* en el cual todos los datos están normalizados en un intervalo $[0,1]$. El gráfico obtenido es el siguiente:

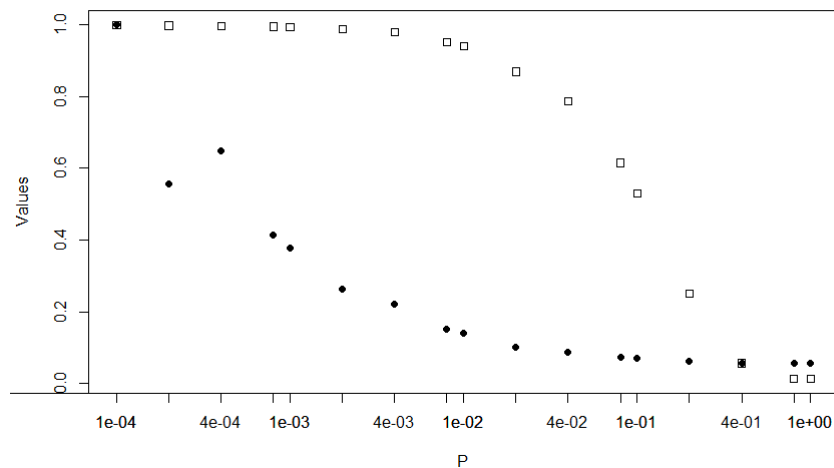


Figura 1: Reproducción del grafo a partir de script

Para adecuarme al gráfico origen he representado los valores del coeficiente de clustering como cuadrados blancos y los valores de camino mas cortos promedio con puntos negros.

Finalmente tras probar con diferentes valores para el modelo he encontrado que los valores mas adecuados son dimensión 1, tamaño 1000, constante nei valor 4 y finalmente la probabilidad p con valor 0.

```
graph <- watts.strogatz.game(1, 1000, 4, 0)
```

2 Tarea 2: Análisis de grafo

En esta tarea quiero realizar el análisis de un grafo que voy a crear a partir de una lista de aristas usando el fichero *edges.txt*. Para leerlo correctamente y asegurarme de que sea leído como un grafo no direccional uso la siguiente instrucción:

```
g <- read.graph("edges.txt", directed = FALSE)
```

Voy ahora a analizar el grafo en detalle, primeramente voy a dibujarlo y luego voy a sacar diferentes características de interés de este grafo.

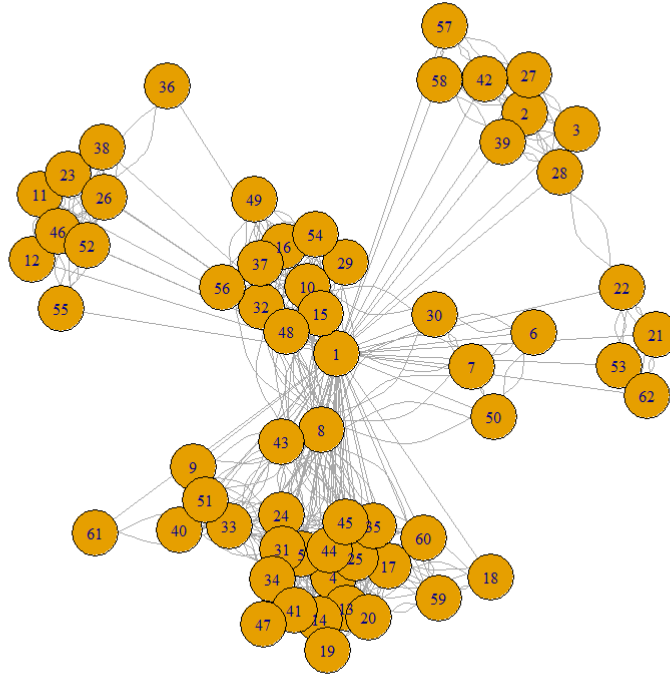


Figura 2: Representación del grafo

```
plot(g)
V(g)
E(g)
diameter(g)
transitivity(g)
degree.distribution(g)
```

A partir de este código obtenemos la siguiente información:

Num. Aristas	Num. Vértices	Diámetro	Transitividad
602	63	2	0.522769

Taula 1: Información extraída del grafo

Paso ahora a analizar la conectividad de este grafo:

```

degree(g)
[1] 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
[40] 13 21 11 23 29 43 15 15 23 15 7 15 15 7 19 7 15 7 9 15 17 3 7

```

Listing 1: Distribución de la frecuencia de grados del grafo

```

degree.distribution(g)
[1] 0.00000000 0.00000000 0.00000000 0.03225806 0.00000000 0.00000000 0.00000000
    0.12903226 0.00000000 0.08064516
[11] 0.00000000 0.06451613 0.00000000 0.11290323 0.00000000 0.11290323 0.00000000
    0.04838710 0.00000000 0.03225806
[21] 0.00000000 0.04838710 0.00000000 0.06451613 0.00000000 0.03225806 0.00000000
    0.01612903 0.00000000 0.01612903
[31] 0.00000000 0.03225806 0.00000000 0.04838710 0.00000000 0.04838710 0.00000000
    0.01612903 0.00000000 0.01612903
[41] 0.00000000 0.00000000 0.00000000 0.01612903 0.00000000 0.00000000 0.00000000
    0.00000000 0.00000000 0.00000000
[51] 0.00000000 0.00000000 0.00000000 0.00000000 0.00000000 0.00000000 0.00000000
    0.00000000 0.00000000 0.01612903
[61] 0.00000000 0.00000000 0.00000000 0.01612903

```

Listing 2: Distribución de la frecuencia de grados del grafo

Para analizar con mayor profundidad voy a crear una nueva representación de la red de forma que las figuras que representan los vértices tengan un tamaño proporcional a la de su pagerank.

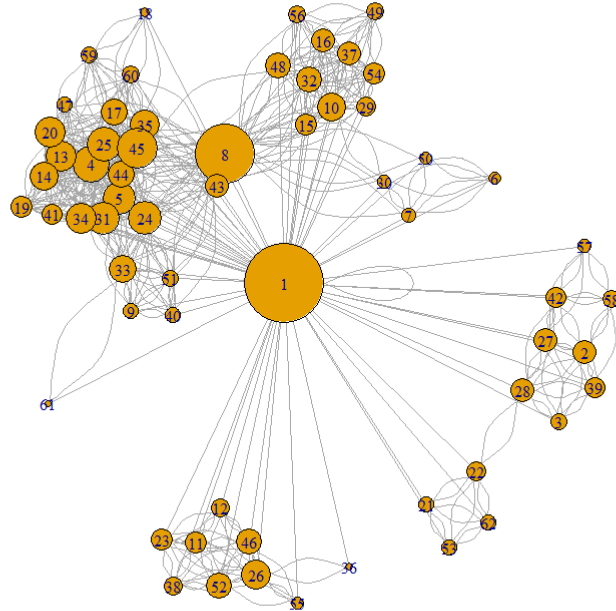


Figura 3: Grafo con nodos tamaño proporcional a su pagerank

A partir de este ultimo gráfico resulta fácil observar que este grafo no es completamente aleatorio, ya que se pueden ver diferentes grupos de nodos y un vértice central con un pagerank muy elevado al que se conectan todo los nodos.

En la segunda parte de esta tarea se pide aplicar un algoritmo de community detection sobre el grafo anterior. He optado por el algoritmo *Walktrap* y he generado el siguiente gráfico:

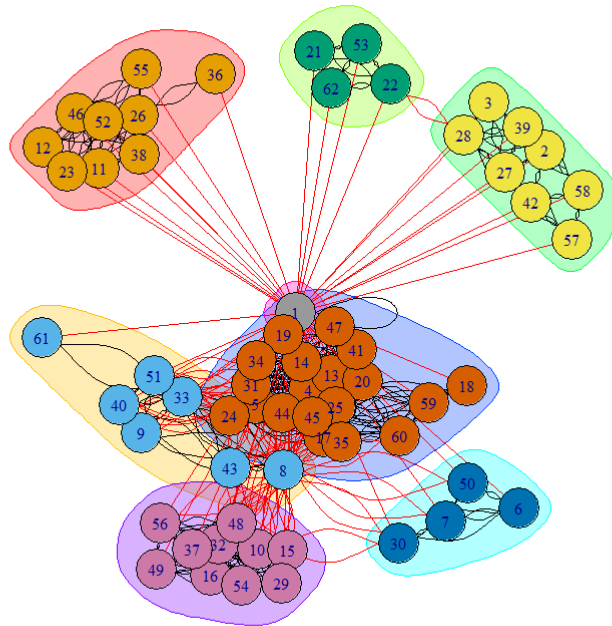


Figura 4: Grafo con nodos tamaño proporcional a su pagerank

Se puede observar claramente que hay una serie de grupos diferenciados en este grafo como ya había supuesto previamente. Finalmente voy a utilizar R para sacar el tamaño de las diferentes comunidades y un histograma del tamaño de estos grupos.

```
sizes(wal)
Community sizes
1 2 3 4 5 6 7 8
9 7 4 8 4 19 10 1
```

Listing 3: Tamaño de las diferentes comunidades

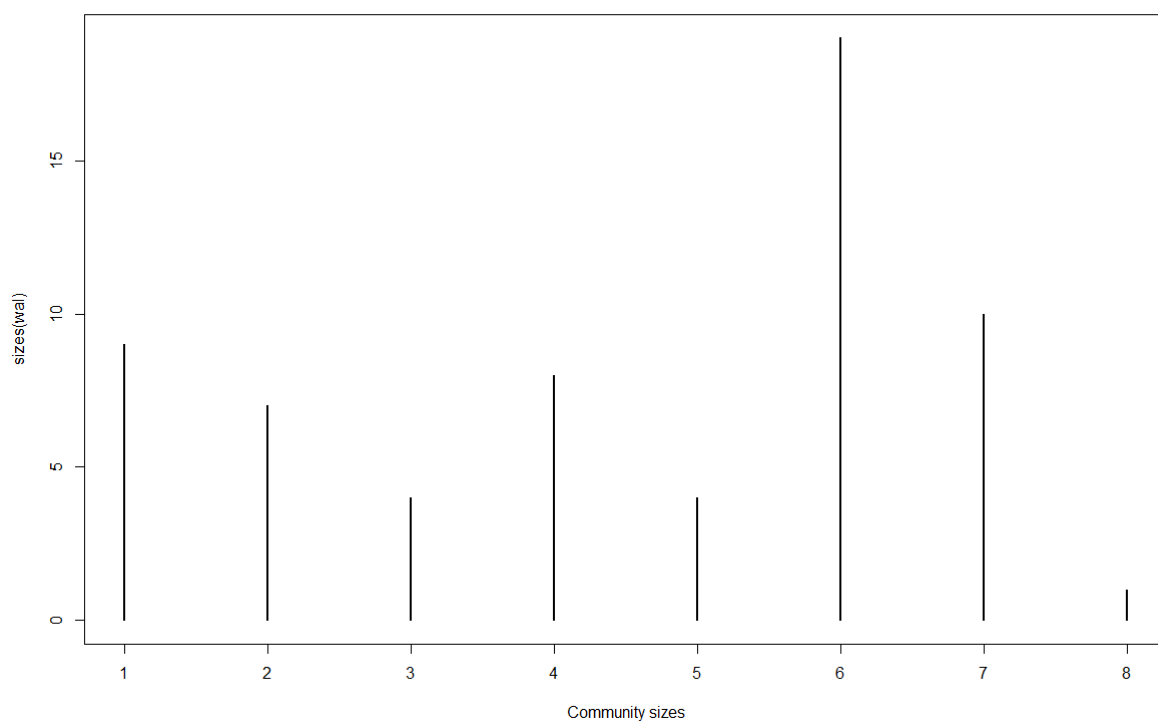


Figura 5: Plot de los diferentes tamaños de las comunidades