

# ICSR: Trabajo evaluación Netlogo

*Alumno: Inmaculada Perea Fernández*

*Marzo 2017*

*Carga de librerías necesarias*

```
library(ggplot2)
```

## Ejercicio 1

*Analiza cómo afecta la probabilidad de añadir aristas al azar a las diversas medidas globales de centralidad de una red libre de escala*

### Script NetLogo

Se ha implementado el siguiente script para NetLogo que utiliza el generador Erdos-Rényi (ER-RN N p). Se parte de una red Barabási-Albert con 300 nodos y se varía la probabilidad de cableado (p) entre 0 y 1, con el generador de la red Erdos-Renyi. Para cada uno de los valores de p se imprime las medidas globales de cualquier red.

Para facilitar el trabajo de exportación de dichas medidas se han implementado dos funciones adicionales, una para imprimir las cabeceras que del fichero resultante (Measure\_headers) y otra para obtener las medidas globales más otra medida que recibe como parámetro (Measures\_p)

```
to ex1
  clear
  no-display
  Measure_headers
  foreach (range 0 1 .01)
  [ [p] ->
    clear
    BA-PA 300 2 1
    ER-RN 0 p
    compute-centralities
    print Measures_p p
  ]
end

to-report Measures_p [p]
  report (list Number-Nodes Number-Links Density
    Average-Degree Average-Path-Length
    Diameter Average-Clustering
    Average-Betweenness Average-Eigenvector
    Average-Closeness Average-PageRank p)
end

to Measure_headers
  let headers []
```

```

set headers store ("Number-Nodes Number-Links Density Average-Degree
Average-Path-Length Diameter Average-Clustering
Average-Betweenness Average-Eigenvector
Average-Closeness Average-PageRank p") headers

print headers
end

```

## Carga del fichero resultante

```

ex1<- read.table("ex1.txt", header=TRUE)
summary(ex1)

```

```

##   Number.Nodes   Number.Links      Density      Average.Degree
##   Min.    :300   Min.    : 299   Min.    :0.006667   Min.    : 1.993
##   1st Qu.:300   1st Qu.:11278   1st Qu.:0.251455   1st Qu.: 75.185
##   Median :300   Median :22259   Median :0.496299   Median :148.393
##   Mean   :300   Mean   :22341   Mean   :0.498119   Mean   :148.937
##   3rd Qu.:300   3rd Qu.:33440   3rd Qu.:0.745585   3rd Qu.:222.930
##   Max.    :300   Max.    :44393   Max.    :0.989811   Max.    :295.953
##   Average.Path.Length   Diameter   Average.Clustering   Average.Betweenness
##   Min.    :1.010         Min.    : 2.0   Min.    :0.0000         Min.    : 1.523
##   1st Qu.:1.254         1st Qu.: 2.0   1st Qu.:0.2517         1st Qu.: 38.035
##   Median :1.504         Median : 2.0   Median :0.4966         Median : 75.303
##   Mean   :1.583         Mean   : 2.3   Mean   :0.4985         Mean   : 87.107
##   3rd Qu.:1.749         3rd Qu.: 2.0   3rd Qu.:0.7457         3rd Qu.:111.907
##   Max.    :5.546         Max.    :11.0   Max.    :0.9898         Max.    :679.583
##   Average.Eigenvector   Average.Closeness   Average.PageRank      p
##   Min.    :0.03557       Min.    :0.1866       Min.    :0.003333   Min.    :0.0000
##   1st Qu.:0.74524       1st Qu.:0.5720       1st Qu.:0.003333   1st Qu.:0.2475
##   Median :0.85832       Median :0.6653       Median :0.003333   Median :0.4950
##   Mean   :0.78505       Mean   :0.6806       Mean   :0.003333   Mean   :0.4950
##   3rd Qu.:0.91547       3rd Qu.:0.7975       3rd Qu.:0.003333   3rd Qu.:0.7425
##   Max.    :0.98988       Max.    :0.9899       Max.    :0.003333   Max.    :0.9900

```

## Representación gráfica

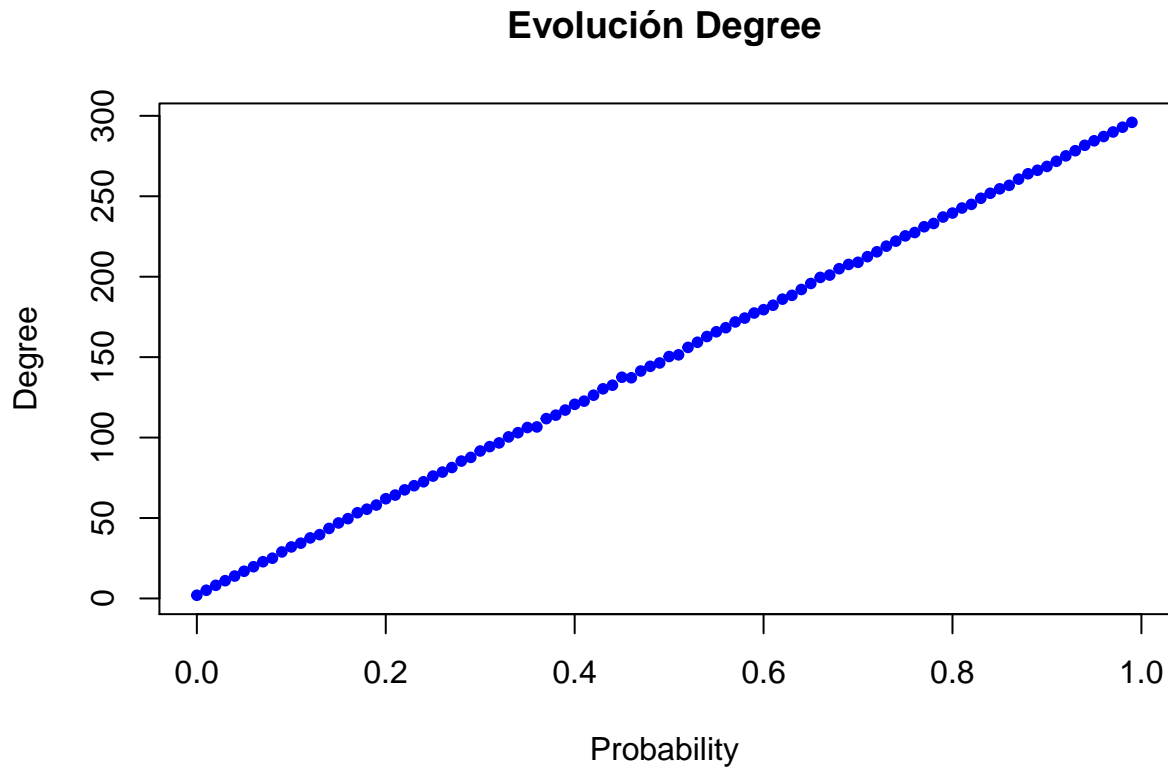
### Degree

Número de enlaces que posee un nodo con los demás.

```

plot(ex1$p, ex1$Average.Degree,
      xlab="Probability", ylab="Degree",
      main="Evolución Degree",
      col="blue",
      pch=20)

```



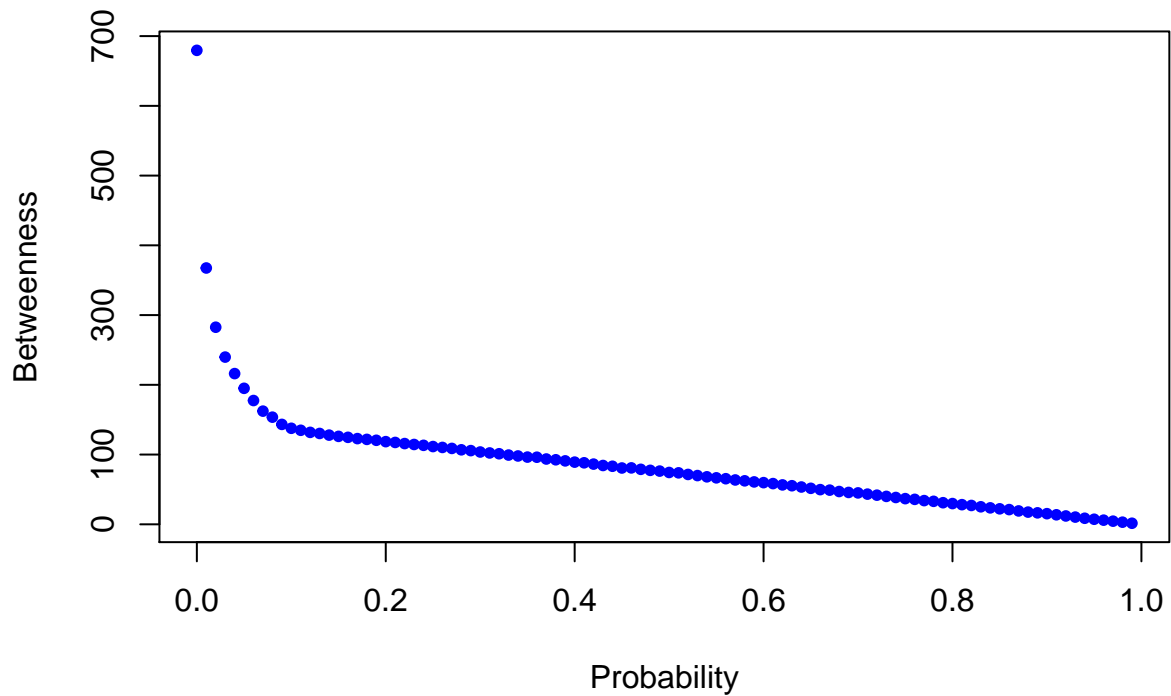
A la vista de la gráfica obtenida, se observa que como cabía esperar el grado aumenta si aumenta la probabilidad de añadir una arista. La relación es directamente proporcional y la pendiente de la recta es casi constante.

### Betweenness

Cuantifica el número de veces que un nodo actúa como un puente a lo largo del camino más corto entre otros dos nodos.

```
plot(ex1$p, ex1$Average.Betweenness,  
     xlab="Probability", ylab="Betweenness",  
     main="Evolución Betweenness",  
     col="blue",  
     pch=20)
```

## Evolución Betweenness

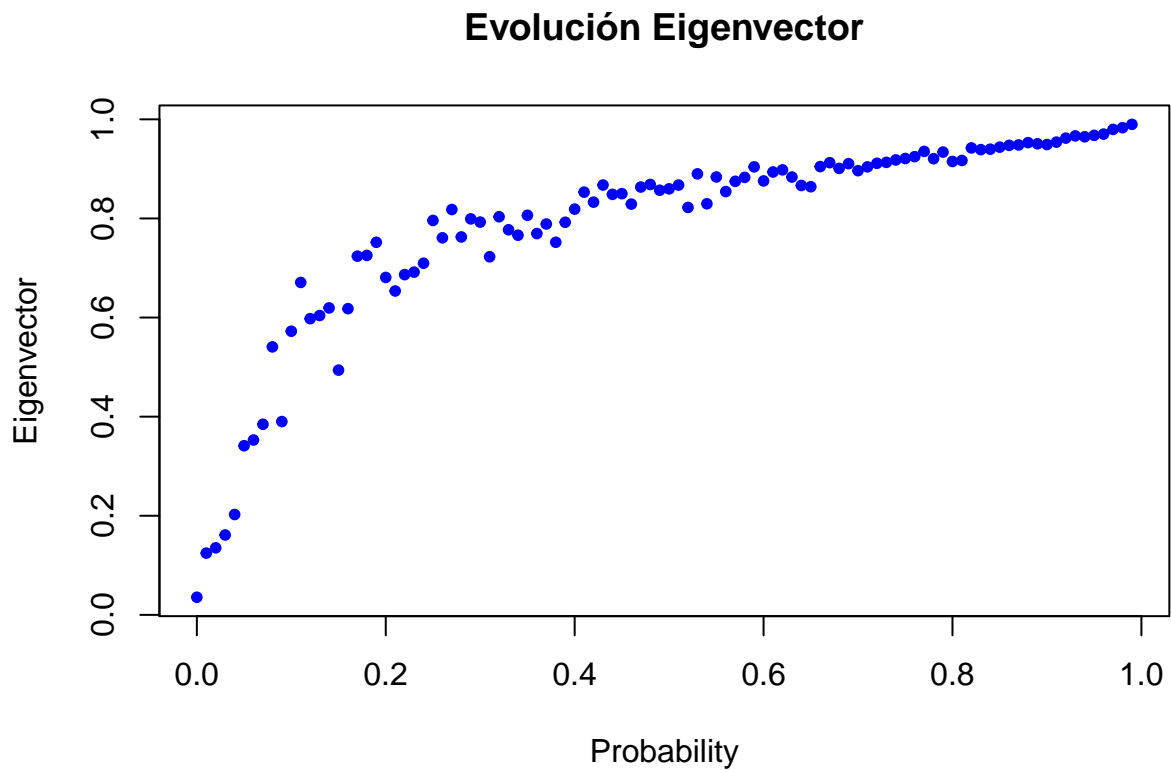


En este caso observamos que a medida que el aumenta la probabilidad de añadir aristas, betweenness decrece, ya que al haber más nodos conectados habrá mas caminos en los que un nodo no intervenga. Se observa también que para probabilidades más bajas  $[0, 0.2]$  decrece más rápido que para probabilidad mayor que 0.2.

### Eigenvector

Mide la influencia de un nodo en una red. Los nodos que poseen un valor alto de esta medida de centralidad están conectados a muchos nodos que a su vez están bien conectados (son buenos candidatos para difundir información, divulgar rumores o enfermedades, etc)

```
plot(ex1$p, ex1$Average.Eigenvector,  
      xlab="Probability", ylab="Eigenvector",  
      main="Evolución Eigenvector",  
      col="blue",  
      pch=20)
```

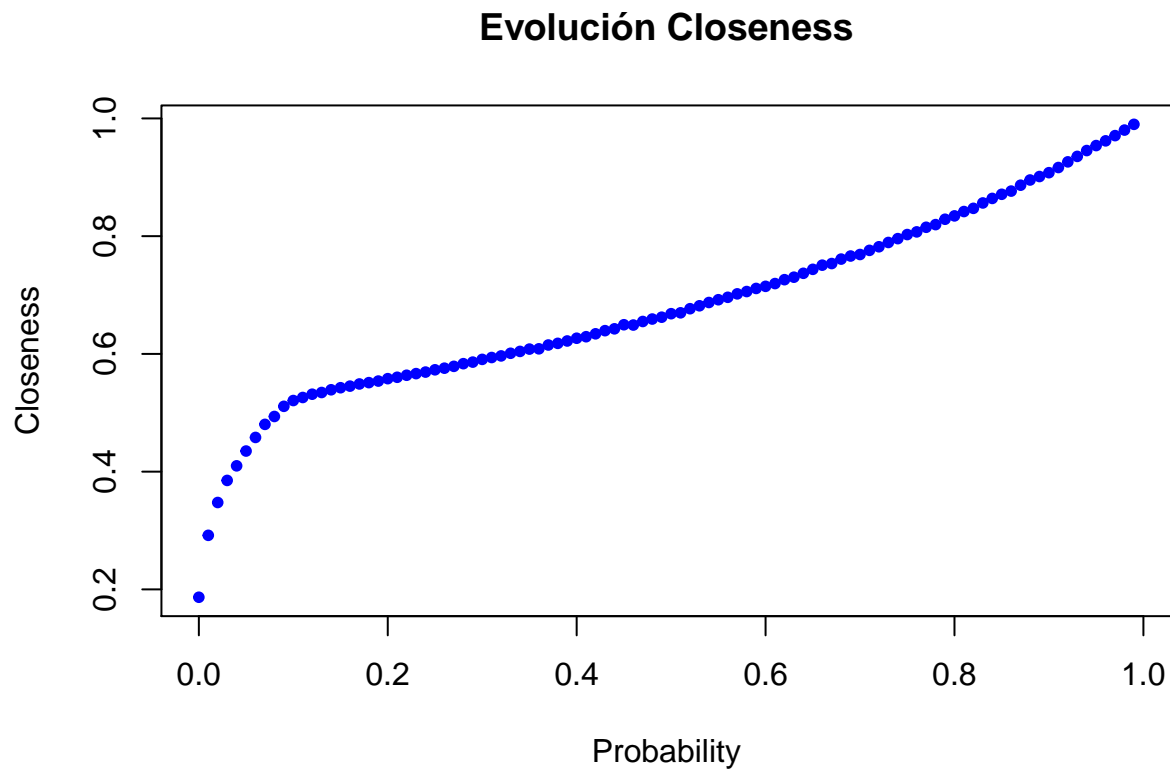


Eigenvector aumenta con la probabilidad hasta valores de probabilidad en torno a 0.4, para valores de  $p$  mayores, Eigenvector aumenta ligeramente, se podría considerar prácticamente constante.

### Closeness

Promedio de las distancias más cortas desde un nodo hacia todos los demás

```
plot(ex1$p, ex1$Average.Closeness,
     xlab="Probability", ylab="Closeness",
     main="Evolución Closeness",
     col="blue",
     pch=20)
```



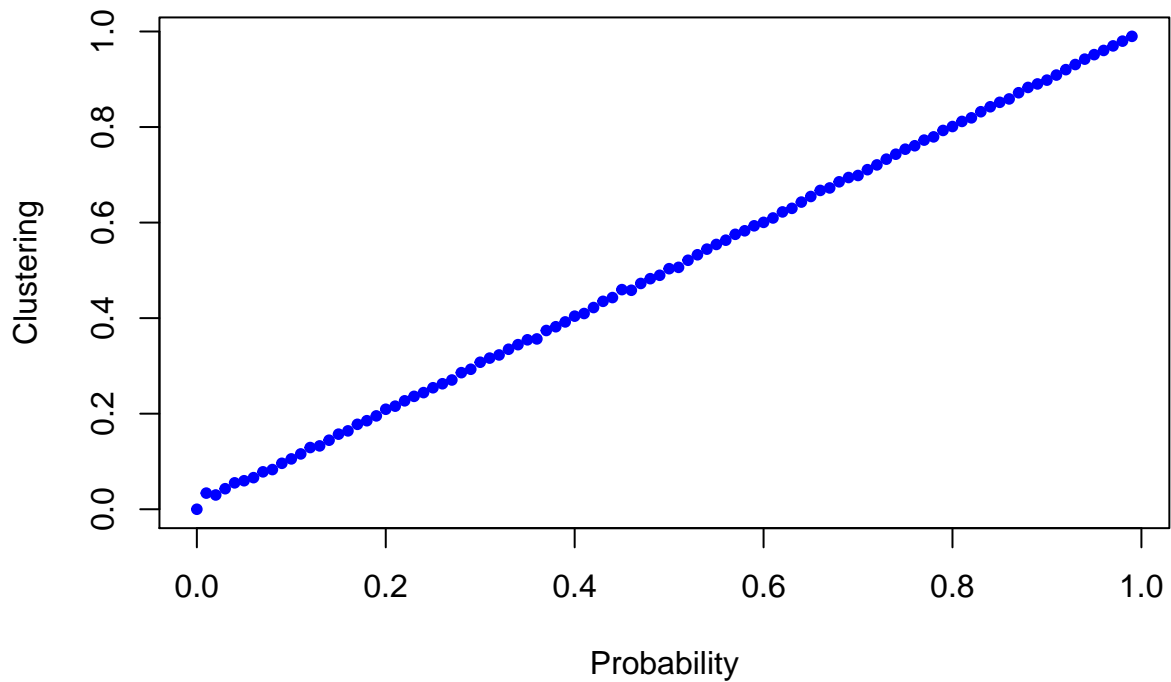
Closeness también aumenta con  $p$ , ya que al aumentar la probabilidad de añadir una arista también es más probable que aumente la longitud del camino entre 2 nodos. Se observa que el crecimiento es mucho más rápido para probabilidades bajas (hasta 0.3)

### Clustering

Probabilidad de que dos vecinos de un nodo de la red escogido aleatoriamente estén conectados entre sí

```
plot(ex1$p, ex1$Average.Clustering,  
     xlab="Probability", ylab="Clustering",  
     main="Evolución Clustering",  
     col="blue",  
     pch=20)
```

## Evolución Clustering

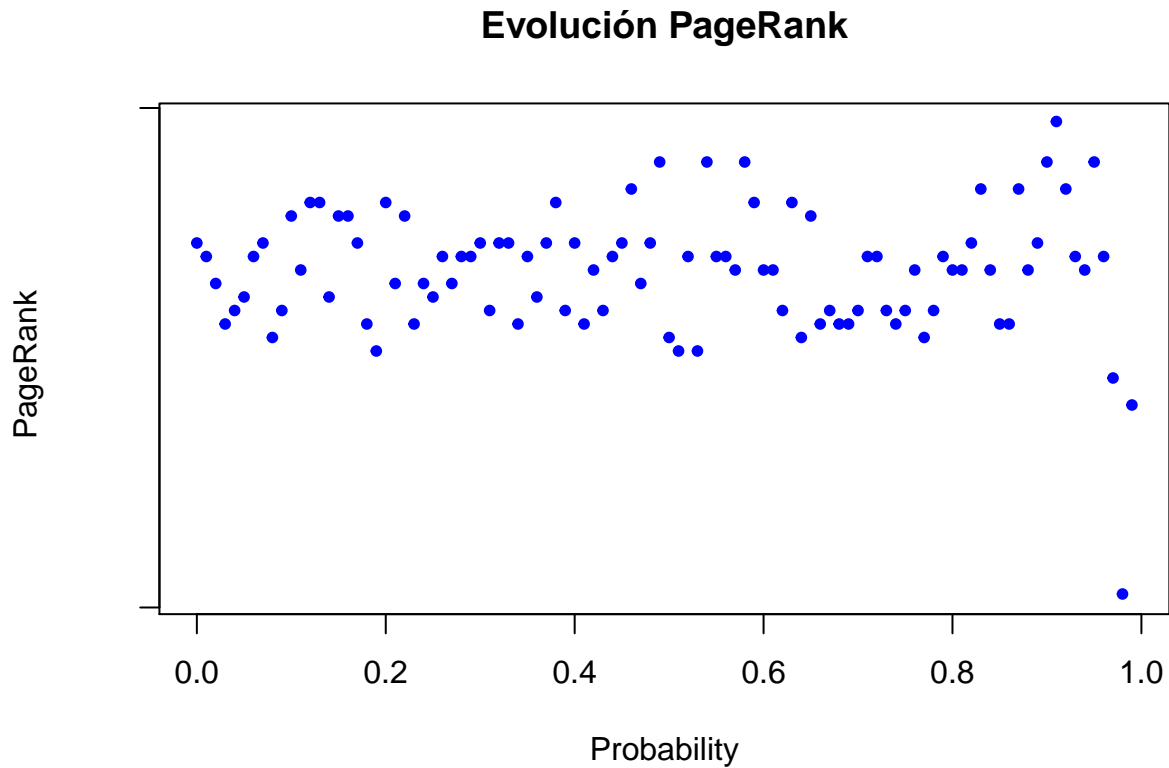


La medida de clustering es directamente proporcional con la probabilidad de añadir aristas. Cuantas más aristas se añadan más probable será que dos nodos aleatoriamente seleccionados estén también conectados.

## PageRank

```
plot(ex1$p, ex1$Average.PageRank,  
     xlab="Probability", ylab="PageRank",  
     main="Evolución PageRank",  
     col="blue",  
     pch=20)
```

```
## Warning in plot.window(...): amplitud relativa de valores = 20 * EPS, es  
## pequeño (eixo 2)
```



Se mantiene prácticamente constante aunque se observan fluctuaciones.

## Ejercicio 2

*Analiza cómo varían el diámetro y longitud de camino medio según los parámetros de los diferentes modelos que ofrece la herramienta (por ejemplo, ¿cómo varían estos valores según el número de nodos de un grafo aleatorio para una probabilidad fija?)*

### 2.1 Modelo Erdos-Rényi

#### 2.1.1 Variando numero de nodos

##### Script NetLogo

```
;; Modelo Erdos-Rényi en funcion del número de nodos
to ex2_er-rn_nodes
  clear
  no-display
  print ("Diameter Average-Path-Length probability nodes")
  foreach (range 2 1000 1)
  [ [n] ->
    clear
    no-display
    ER-RN n .47
    print (list diameter Average-Path-Length .47 n)
```



```
]
end
```

### Carga de datos

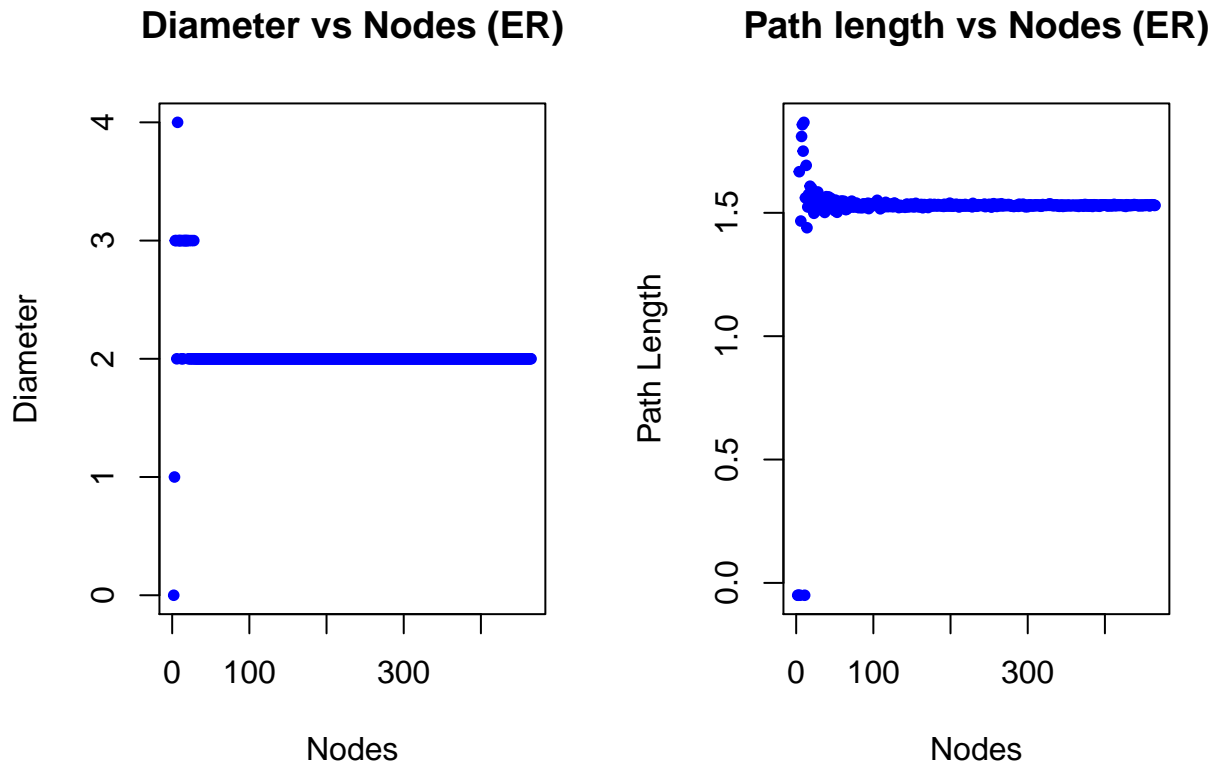
```
ex2_er_rn_nodes<- read.table("ex2_er-rn_nodes.txt", header=TRUE)
summary(ex2_er_rn_nodes)
```

```
##      Diameter      Average.Path.Length  probability      nodes
##  Min.   :0.000   Min.   : -0.050      Min.   :0.47   Min.   :  2.0
## 1st Qu.:2.000   1st Qu.: 1.528      1st Qu.:0.47   1st Qu.:117.8
## Median :2.000   Median : 1.530      Median :0.47   Median :233.5
## Mean   :2.032   Mean   : 1.520      Mean   :0.47   Mean   :233.5
## 3rd Qu.:2.000   3rd Qu.: 1.532      3rd Qu.:0.47   3rd Qu.:349.2
## Max.   :4.000   Max.   : 1.867      Max.   :0.47   Max.   :465.0
```

### Representación gráfica

```
par(mfrow =c(1,2))
plot(ex2_er_rn_nodes$nodes, ex2_er_rn_nodes$Diameter,
     xlab="Nodes", ylab="Diameter",
     main="Diameter vs Nodes (ER)",
     col="blue",
     pch=20)

plot(ex2_er_rn_nodes$nodes, ex2_er_rn_nodes$Average.Path.Length,
     xlab="Nodes", ylab="Path Length",
     main="Path length vs Nodes (ER)",
     col="blue",
     pch=20)
```



En el modelo de Erdos-Rényi se observa que el diámetro de la red se mantiene constante aunque aumente el número de nodos. Y que el camino medio decrece al aumentar el número de nodos al principio, pero que al alcanzar los 50 nodos empieza a dejar de tener influencia y mantenerse constante.

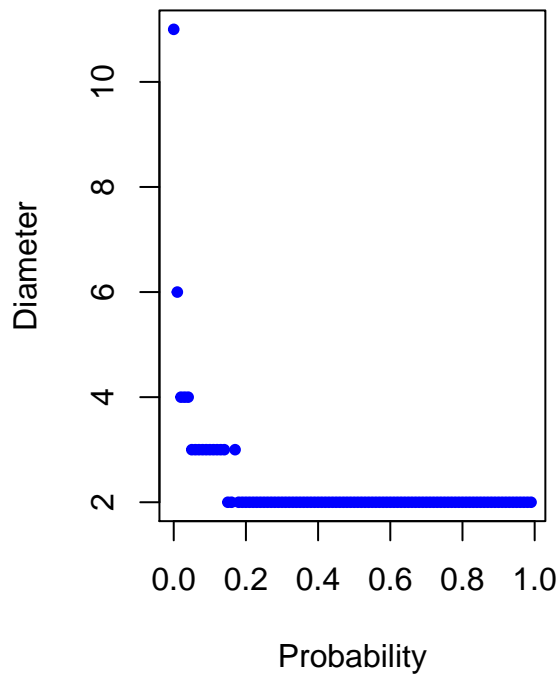
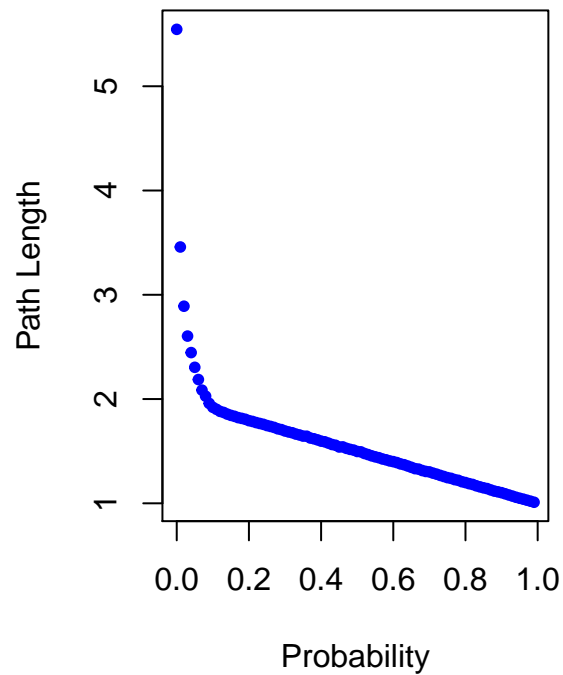
*Nota: en los los datos anteriores se ha reemplazado los valores false del camino medio que devolver NetLogo cuando no es posible calcular el camino medio por el valor -0.05. Vemos que esto sucede cuando se encuentran nodos inconexos o aislados.*

### 2.1.2 Variando probabilidad de añadir aristas

Se utilizan los datos obtenidos con la simulación del script del ejercicio 1.

```
par(mfrow =c(1,2))
plot(ex1$p, ex1$Diameter,
     xlab="Probability", ylab="Diameter",
     main="Diameter vs Probability (ER)",
     col="blue",
     pch=20)

plot(ex1$p, ex1$Average.Path.Length,
     xlab="Probability", ylab="Path Length",
     main="Path length vs Probability (ER)",
     col="blue",
     pch=20)
```

**Diameter vs Probability (ER)****Path length vs Probability (ER)**

No influye aumentar la probabilidad de añadir aristas en el diámetro de la red, esta medida se mantiene constante. Sin embargo la relación con la longitud del camino es inversamente proporcional, si aumenta  $p$  disminuye la longitud de camino medio.

## 2.2 Modelo Barabasi & Albert

### 2.2.1 Variando Número de nodos

#### Script NetLogo

```
to ex2_ba-pa_nodes
  clear
  no-display
  print ("Diameter Average-Path-Length nodes initial links")
  foreach (range 1 1000 1)
  [ [n] ->
    clear
    no-display
    BA-PA n 2 1
    print (list diameter Average-Path-Length n 2 1)
  ]
end
```

#### Carga de datos

```
ex2_ba_pa_nodes<- read.table("04.ex2_ba-pa_nodes.txt", header=TRUE)
summary(ex2_ba_pa_nodes)
```

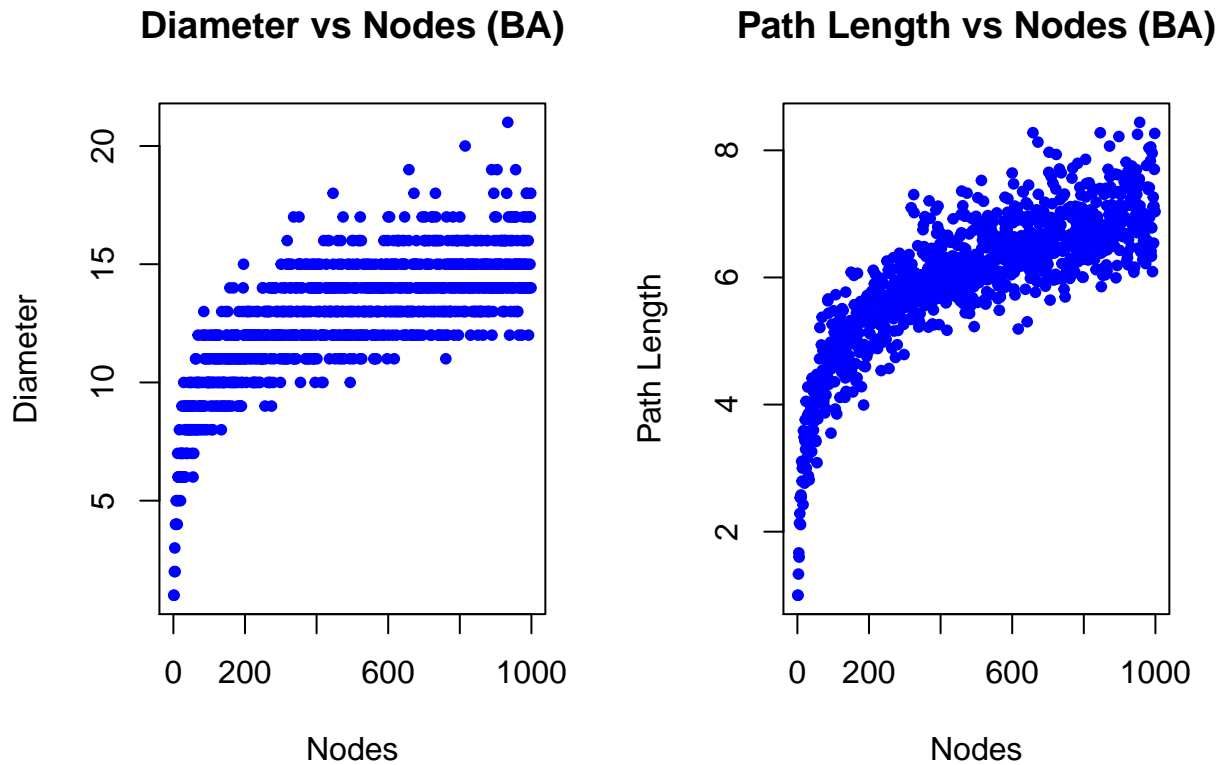
```
##      Diameter      Average.Path.Length      nodes      initial
## Min.       : 1.00    Min.       :1.000      Min.       : 1.0    Min.       :2
## 1st Qu.:12.00    1st Qu.:5.524      1st Qu.:250.5    1st Qu.:2
## Median :13.00    Median :6.164      Median :500.0    Median :2
## Mean      :12.81    Mean      :6.000      Mean      :500.0    Mean      :2
## 3rd Qu.:14.00    3rd Qu.:6.697      3rd Qu.:749.5    3rd Qu.:2
## Max.       :21.00    Max.       :8.440      Max.       :999.0    Max.       :2
##      links
## Min.       :1
## 1st Qu.:1
## Median :1
## Mean      :1
## 3rd Qu.:1
## Max.       :1
```

### Representación gráfica

```
par(mfrow =c(1,2))

plot(ex2_ba_pa_nodes$nodes, ex2_ba_pa_nodes$Diameter,
     xlab="Nodes", ylab="Diameter",
     main="Diameter vs Nodes (BA)",
     col="blue",
     pch=20)

plot(ex2_ba_pa_nodes$nodes, ex2_ba_pa_nodes$Average.Path.Length,
     xlab="Nodes", ylab="Path Length",
     main="Path Length vs Nodes (BA)",
     col="blue",
     pch=20)
```



Se observa que para el modelo de Barabasi & Albert si el diámetro y el camino medio aumentan si aumenta el número de nodos. Sin embargo alcanzados los 200 nodos aproximadamente el aumento es mucho menos pronunciado, la variación al aumentar el numero de nodos llegado este rango ya es muy pequeño.

### 2.2.2 Variando el grafo inicial completo

#### Script NetLogo

```
to ex2_ba-pa_initial
  clear
  no-display
  print ("Diameter Average-Path-Length nodes initial links")
  foreach (range 2 700 1)
  [ [m0] ->
    clear
    no-display
    BA-PA 700 m0 1
    print (list diameter Average-Path-Length 700 m0 1)
  ]
end
```

#### Carga de datos

```
ex2_ba_pa_initial<- read.table("05.ex2_ba-pa_initial.txt", header=TRUE)
summary(ex2_ba_pa_initial)
```

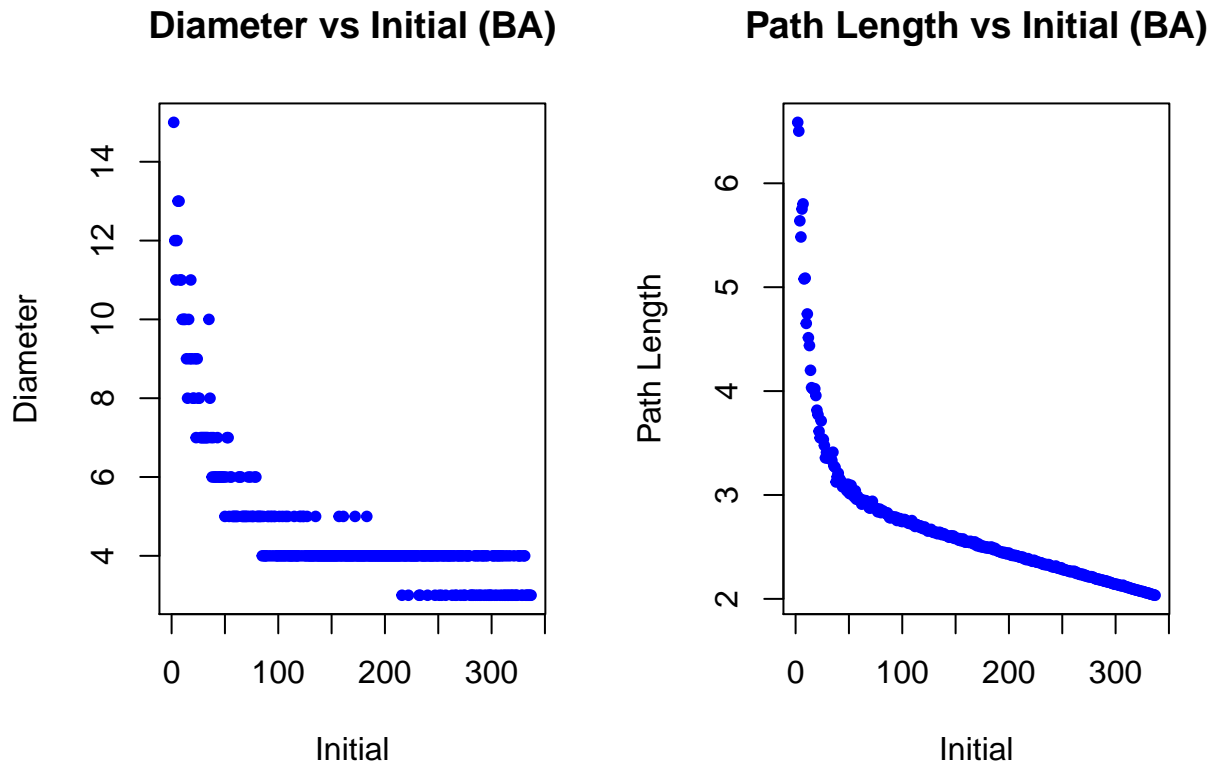
```
##      Diameter      Average.Path.Length      nodes      initial
```

```
## Min.    : 3.000    Min.    :2.035        Min.    :700    Min.    : 2.00
## 1st Qu.: 4.000    1st Qu.:2.276        1st Qu.:700    1st Qu.: 85.75
## Median : 4.000    Median :2.529        Median :700    Median :169.50
## Mean   : 4.696    Mean   :2.678        Mean   :700    Mean   :169.50
## 3rd Qu.: 5.000    3rd Qu.:2.821        3rd Qu.:700    3rd Qu.:253.25
## Max.   :15.000    Max.   :6.587        Max.   :700    Max.   :337.00
##      links
## Min.    :1
## 1st Qu.:1
## Median :1
## Mean    :1
## 3rd Qu.:1
## Max.    :1
```

### Representación gráfica

```
par(mfrow =c(1,2))
plot(ex2_ba_pa_initial$initial, ex2_ba_pa_initial$Diameter,
     xlab="Initial", ylab="Diameter",
     main="Diameter vs Initial (BA)",
     col="blue",
     pch=20)

plot(ex2_ba_pa_initial$initial, ex2_ba_pa_initial$Average.Path.Length,
     xlab="Initial", ylab="Path Length",
     main="Path Length vs Initial (BA)",
     col="blue",
     pch=20)
```



Ambas medidas, diámetro y longitud media del camino decrecen si aumenta el grafo inicial.

### 2.2.3 Variando el número de links

#### Script NetLogo

```
to ex2_ba-pa_links
  clear
  no-display
  print ("Diameter Average-Path-Length nodes initial links")
  foreach (range 1 500 1)
  [ [m] ->
    clear
    no-display
    BA-PA 500 2 m
    print (list diameter Average-Path-Length 500 2 m)
  ]
end
```

#### Carga de datos

```
ex2_ba_pa_links<- read.table("06.ex2_ba-pa_links.txt", header=TRUE)
summary(ex2_ba_pa_links)
```

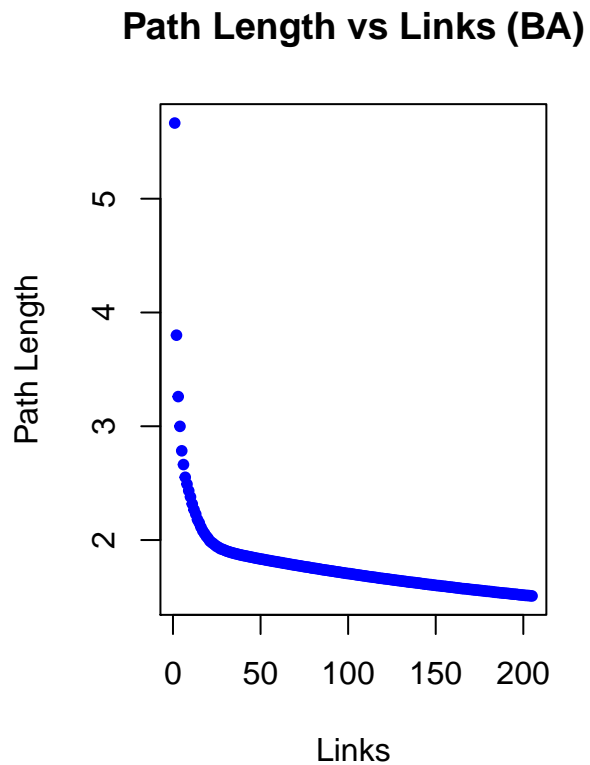
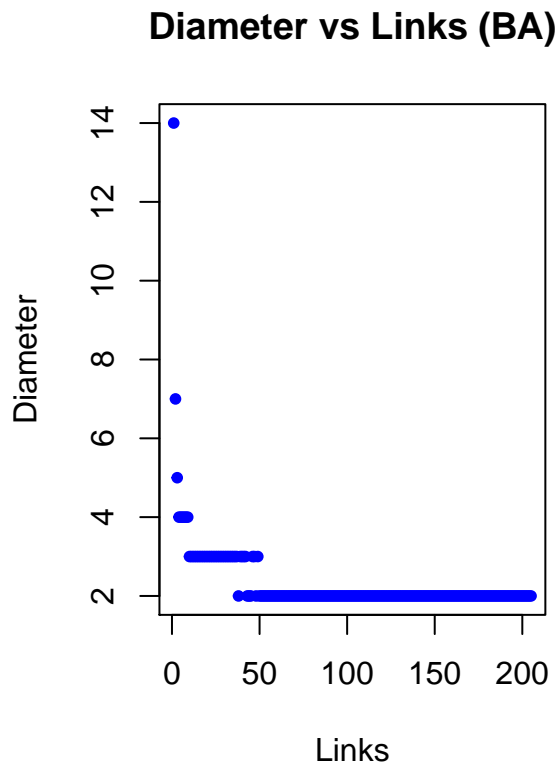
```
##      Diameter      Average.Path.Length      nodes      initial
##  Min.   : 2.000    Min.   :1.509      Min.   :500    Min.   :2
## 1st Qu.: 2.000    1st Qu.:1.594      1st Qu.:500    1st Qu.:2
```

```
## Median : 2.000   Median :1.699       Median :500   Median :2
## Mean   : 2.327   Mean   :1.783       Mean   :500   Mean   :2
## 3rd Qu.: 2.000   3rd Qu.:1.828       3rd Qu.:500   3rd Qu.:2
## Max.   :14.000   Max.   :5.665       Max.   :500   Max.   :2
##      links
## Min.    : 1
## 1st Qu.: 52
## Median :103
## Mean    :103
## 3rd Qu.:154
## Max.    :205
```

### Representación gráfica

```
par(mfrow =c(1,2))
plot(ex2_ba_pa_links$links, ex2_ba_pa_links$Diameter,
     xlab="Links", ylab="Diameter",
     main="Diameter vs Links (BA)",
     col="blue",
     pch=20)

plot(ex2_ba_pa_links$links, ex2_ba_pa_links$Average.Path.Length,
     xlab="Links", ylab="Path Length",
     main="Path Length vs Links (BA)",
     col="blue",
     pch=20)
```





El diametro decrece ligeramente si aumenta el número de enlaces, pero a partir de cierta cantidad de enlaces (50 aproximadamente) el diámetro se mantiene constante. La longitud media del camino sí que decrece a medida que aumenta el numero de enlaces.

## 2.3 Modelo Watts & Strogatz

### 2.3.1 Variando el número de nodos

#### Script NetLogo

```
to ex2_ws_nodes
  clear
  no-display
  print ("Diameter Average-Path-Length nodes initial rewiring")
  foreach (range 2 1000 1)
  [ [n] ->
    clear
    no-display
    WS n 2 .6
    print (list diameter Average-Path-Length n 2 .6)
  ]
end
```

#### Carga de datos

```
ex2_ws_nodes<- read.table("07.ex2_ws_nodes.txt", header=TRUE)
summary(ex2_ws_nodes)
```

```
##      Diameter      Average.Path.Length      nodes      initial
## Min.   : 1.00    Min.   :-1.000      Min.   : 2.0    Min.   :2
## 1st Qu.: 38.00    1st Qu.:-1.000      1st Qu.:251.2    1st Qu.:2
## Median : 54.00    Median :-1.000      Median :500.5    Median :2
## Mean   : 54.47    Mean   : 2.012      Mean   :500.5    Mean   :2
## 3rd Qu.: 70.75    3rd Qu.:-1.000      3rd Qu.:749.8    3rd Qu.:2
## Max.   :135.00    Max.   :47.533      Max.   :999.0    Max.   :2
##      rewiring
## Min.   :0.6
## 1st Qu.:0.6
## Median :0.6
## Mean   :0.6
## 3rd Qu.:0.6
## Max.   :0.6
```

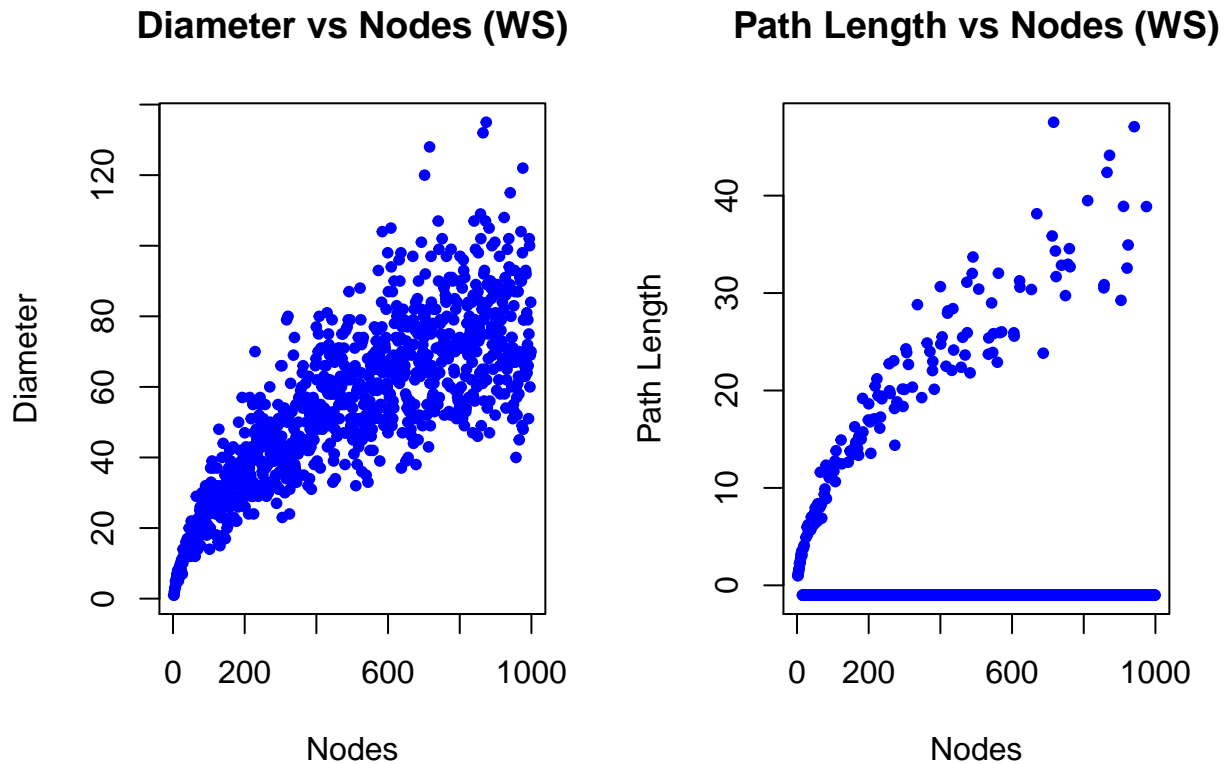
#### Representación gráfica

```
par(mfrow =c(1,2))

plot(ex2_ws_nodes$nodes, ex2_ws_nodes$Diameter,
     xlab="Nodes", ylab="Diameter",
     main="Diameter vs Nodes (WS)",
     col="blue",
     pch=20)

plot(ex2_ws_nodes$nodes, ex2_ws_nodes$Average.Path.Length,
     xlab="Nodes", ylab="Path Length",
     main="Path Length vs Nodes (WS)",
```

```
col="blue",
pch=20)
```



En el modelo Watts & Strogatz vemos que el aumento del número de nodos hace que aumente tanto el diámetro como la longitud del camino medio de la red.

- *Nota: se ha sustituido por -1 los valores false obtenidos en el cálculo del camino.*

### 2.3.2 Variando la probabilidad de recableado

#### Script NetLogo

```
to ex2_ws_rewiring
  clear
  no-display
  print ("Diameter Average-Path-Length nodes initial rewiring")
  foreach (range 0 1 .001)
  [ [p] ->
    clear
    no-display
    BA-PA 500 2 1
    WS 0 1 p
    print (list diameter Average-Path-Length 0 1 p)
  ]
end
```

Carga de datos

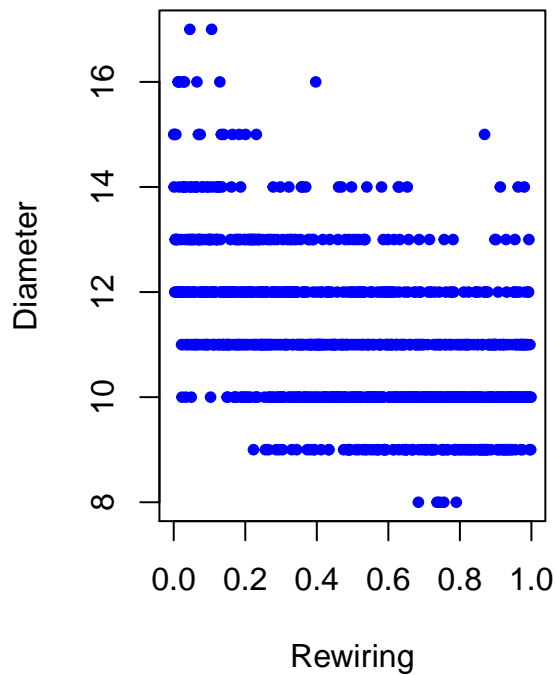
```
ex2_ws_rewiring<- read.table("08.ex2_ws_rewiring.txt", header=TRUE)
summary(ex2_ws_rewiring)
```

```
##      Diameter      Average.Path.Length      nodes      initial
## Min.       : 8.00    Min.       :-1.0000    Min.       :0    Min.       :1
## 1st Qu.:10.00    1st Qu.: -1.0000    1st Qu.:0    1st Qu.:1
## Median :11.00    Median :-1.0000    Median :0    Median :1
## Mean   :11.06    Mean   :-0.9703    Mean   :0    Mean   :1
## 3rd Qu.:12.00    3rd Qu.: -1.0000    3rd Qu.:0    3rd Qu.:1
## Max.    :17.00    Max.    : 7.0577    Max.    :0    Max.    :1
##      rewiring
## Min.       :0.0000
## 1st Qu.:0.2497
## Median :0.4995
## Mean   :0.4995
## 3rd Qu.:0.7492
## Max.    :0.9990
```

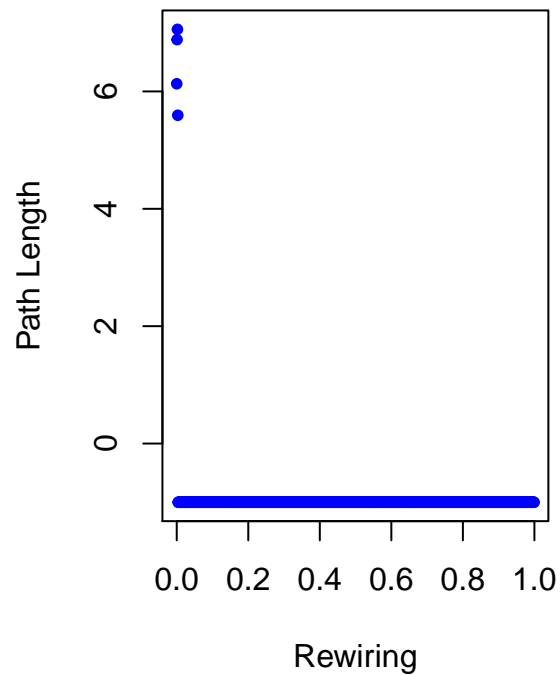
```
par(mfrow =c(1,2))
plot(ex2_ws_rewiring$rewiring, ex2_ws_rewiring$Diameter,
     xlab="Rewiring", ylab="Diameter",
     main="Diameter vs Rewiring (WS)",
     col="blue",
     pch=20)

plot(ex2_ws_rewiring$rewiring, ex2_ws_rewiring$Average.Path.Length,
     xlab="Rewiring", ylab="Path Length",
     main="Path Lenght vs Rewiring (WS)",
     col="blue",
     pch=20)
```

### Diameter vs Rewiring (WS)



### Path Length vs Rewiring (WS)



Ninguna de las dos medidas, longitud media del camino y diámetro parecen tener relación directa con la probabilidad de recableado.

### 2.3.3 Variando el grado inicial

#### Script NetLogo

```
to ex2_ws_initial
  clear
  no-display
  print ("Diameter Average-Path-Length nodes initial rewiring")
  foreach (range 1 200 1)
  [ [i] ->
    clear
    no-display
    WS 700 i .6
    print (list diameter Average-Path-Length 700 i .6)
  ]
end
```

#### Carga de datos

```
ex2_ws_initial<- read.table("09.ex2_ws_initial.txt", header=TRUE)
summary(ex2_ws_initial)
```

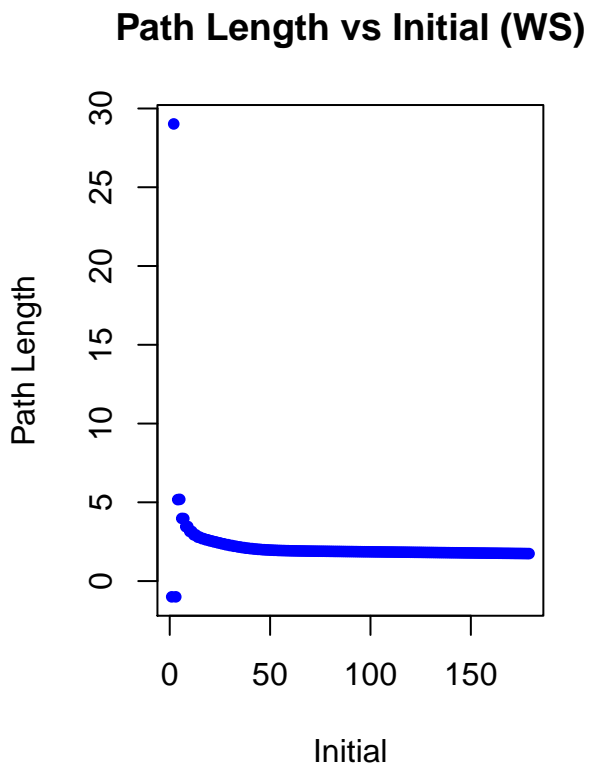
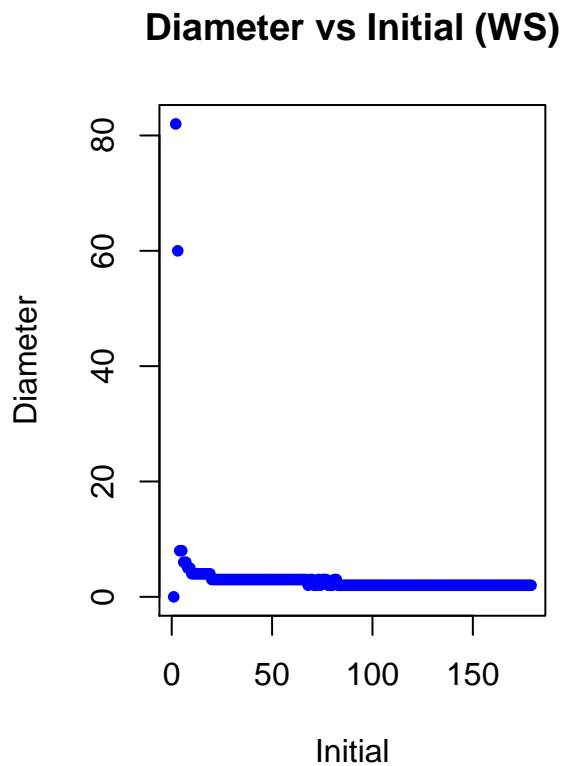
```
##      Diameter      Average.Path.Length      nodes      initial
## Min.       : 0.00   Min.       :-1.000   Min.       :700   Min.       : 1.0
```

```
## 1st Qu.: 2.00    1st Qu.: 1.805      1st Qu.:700    1st Qu.: 45.5
## Median : 2.00    Median : 1.868      Median :700    Median : 90.0
## Mean   : 3.33    Mean   : 2.156      Mean   :700    Mean   : 90.0
## 3rd Qu.: 3.00    3rd Qu.: 1.986      3rd Qu.:700    3rd Qu.:134.5
## Max.   :82.00    Max.   :29.019      Max.   :700    Max.   :179.0
## rewiring
## Min.    :0.6
## 1st Qu.:0.6
## Median :0.6
## Mean    :0.6
## 3rd Qu.:0.6
## Max.    :0.6
```

### Representación gráfica

```
par(mfrow =c(1,2))
plot(ex2_ws_initial$initial, ex2_ws_initial$Diameter,
     xlab="Initial", ylab="Diameter",
     main="Diameter vs Initial (WS)",
     col="blue",
     pch=20)

plot(ex2_ws_initial$initial, ex2_ws_initial$Average.Path.Length,
     xlab="Initial", ylab="Path Length",
     main="Path Length vs Initial (WS)",
     col="blue",
     pch=20)
```



El grado inicial no parece tener mucha influencia sobre ambas medidas, porque prácticamente se mantienen constantes aunque variemos el grado inicial.

## Ejercicio 3

*Haz experimentos de infección de un virus haciendo uso de Spread para una familia de redes parametrizada, elegida por ti, e intenta concluir para qué valores de los parámetros se obtiene redes con mayor resistencia y redes con mayor permeabilidad*

### 3.1 En funcion del número de nodos

#### Script NetLogo

Se usa una red de Erdos-Rényi de 300 nodos y probabilidad 0.01 de añadir aristas, y voy variando el numero de nodos inicialmente infectados y el numero de iteraciones

```
to infeccion_nodos
  print ("NoInfectados Infectados Inmunes Nodos Pinfeccion Precuperacion Pinmunidad Iteraciones Model")
  foreach (range 1 100 5)
  [ [i] ->
    foreach (range 0 150 5)
    [ [n] ->
      clear
      ER-RN 300 .01
      layout "spring"
      Spread n 0.3 0.3 0 i
      print (spread-summary-param n 0.3 0.3 0 i "ER")
    ]
  ]
end
```

```
to-report spread-summary-param [Nodos Pinfeccion Precuperacion Pinmunidad Iteraciones Model]
  let s count nodes with [infected = 0]
  let i count nodes with [infected = 1]
  let r count nodes with [infected = 2]
  report (list s i r Nodos Pinfeccion Precuperacion Pinmunidad Iteraciones Model)
end
```

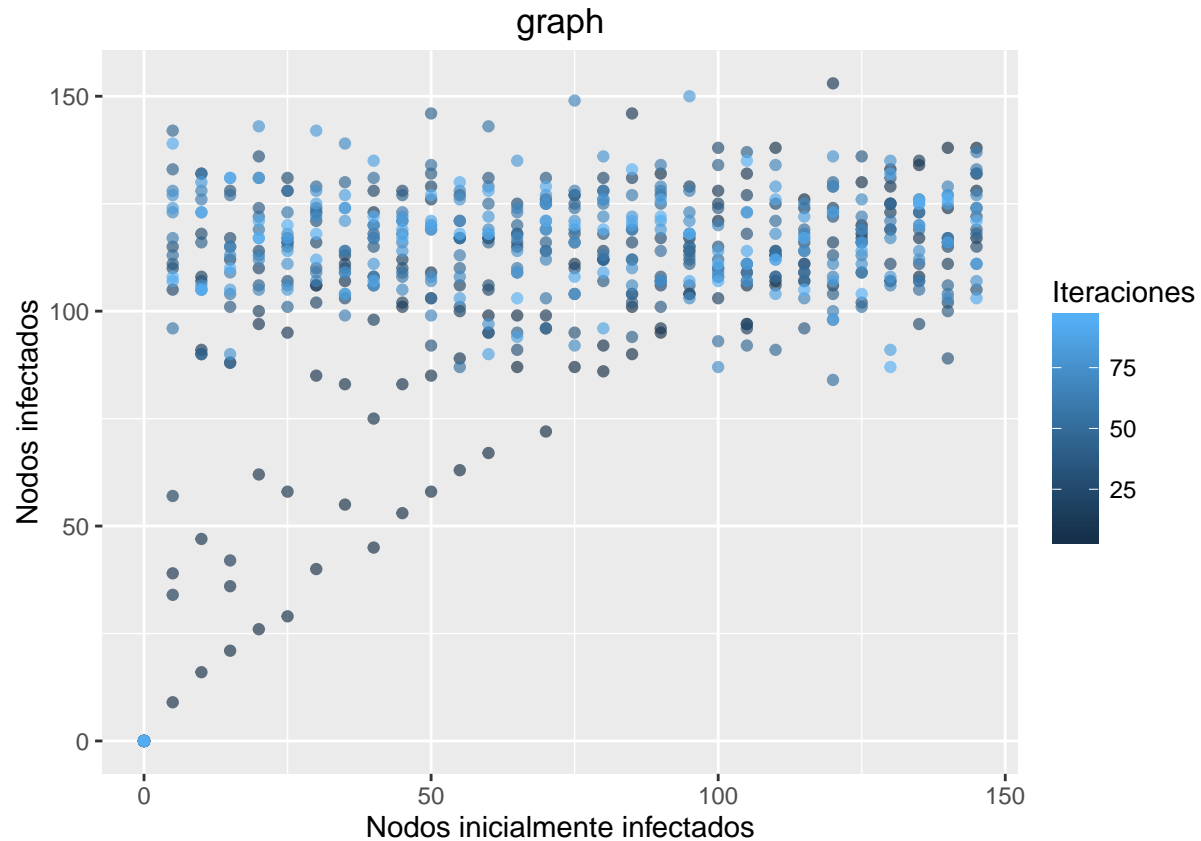
```
infeccion_nodos<- read.table("infeccion_nodos.txt", header=TRUE)
summary(infeccion_nodos)
```

```
##   NoInfectados   Infectados   Inmunes   Nodos   Pinfeccion
##   Min.    :147.0   Min.    : 0.0   Min.    :0   Min.    : 0.0   Min.    :0.3
##   1st Qu.:177.0   1st Qu.:105.0   1st Qu.:0   1st Qu.: 35.0   1st Qu.:0.3
##   Median :186.0   Median :114.0   Median :0   Median : 72.5   Median :0.3
##   Mean   :191.2   Mean   :108.8   Mean    :0   Mean   : 72.5   Mean   :0.3
##   3rd Qu.:195.0   3rd Qu.:123.0   3rd Qu.:0   3rd Qu.:110.0   3rd Qu.:0.3
##   Max.    :300.0   Max.    :153.0   Max.    :0   Max.    :145.0   Max.    :0.3
##   Precuperacion   Pinmunidad   Iteraciones
##   Min.    :0.3   Min.    :0   Min.    : 1.00
##   1st Qu.:0.3   1st Qu.:0   1st Qu.:24.75
##   Median :0.3   Median :0   Median :48.50
```

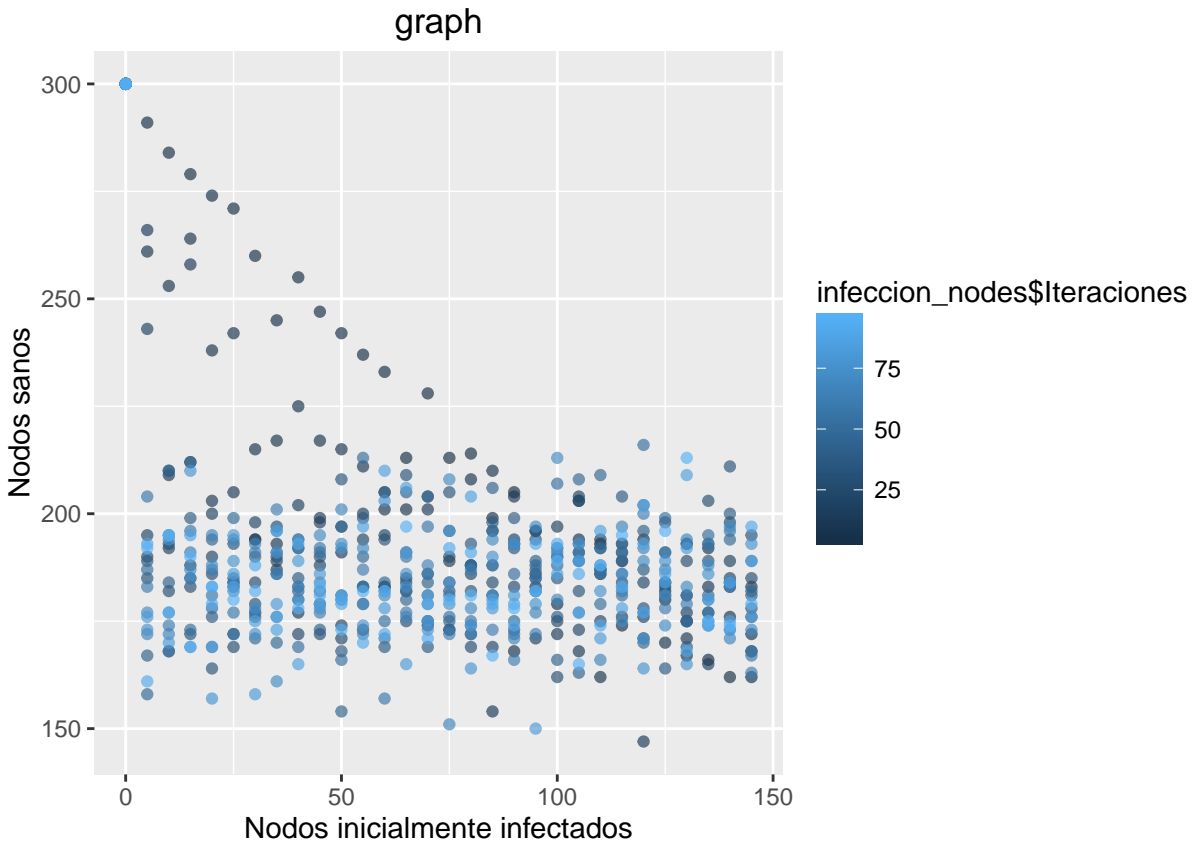
```
## Mean :0.3 Mean :0 Mean :48.50
## 3rd Qu.:0.3 3rd Qu.:0 3rd Qu.:72.25
## Max. :0.3 Max. :0 Max. :96.00
```

```
par(mfrow = c(1,2))
```

```
qplot(infeccion_nodos$Nodos, infeccion_nodos$Infectados, data = infeccion_nodos,
      xlab="Nodos inicialmente infectados",
      ylab="Nodos infectados",
      colour = Iteraciones,
      alpha = I(0.65), main = "graph")
```



```
qplot(infeccion_nodos$Nodos, infeccion_nodos$NoInfectados, data = infeccion_nodos,
      xlab="Nodos inicialmente infectados",
      ylab="Nodos sanos",
      colour = infeccion_nodos$Iteraciones,
      alpha = I(0.65), main = "graph")
```



Se observa que el número de nodos inicialmente infectados no tiene mucho efecto a la hora de propagar una infección o mensaje en las redes Erdos-Rényi, ya que tanto el numero de nodos infectados como el numero de nodos no infectados se mantienen dentro de un rango independientemente de si se aumenta o no el numero inicial de nodos infectados.

### 3.2 En función de la probabilidad de infección ( $p_s$ )

#### Script NetLogo

```
to infeccion_ps
  print ("NoInfectados Infectados Inmunes Nodos Pinfeccion Precuperacion Pinmunidad Iteraciones Model")
  ;; Erdos Renyi
  foreach (range 0 1 0.05)
  [ [ps] ->
    clear
    ER-RN 300 .01
    layout "spring"
    Spread 1 ps 0.3 0 10
    print (spread-summary-param 1 ps 0.3 0 10 "ER")
  ]

  ;; Barabasi & Albert
  foreach (range 0 1 0.05)
  [ [ps] ->
    clear
    BA-PA 300 2 1
```



```

    layout "spring"
    Spread 1 ps 0.3 0 10
    print (spread-summary-param 1 ps 0.3 0 10 "BA")
  ]

;; Watts & Strogatz
foreach (range 0 1 0.05)
[ [ps] ->
  clear
  WS 300 2 .01
  layout "spring"
  Spread 1 ps 0.3 0 10
  print (spread-summary-param 1 ps 0.3 0 10 "WS")
]

end

```

### Carga de los datos

```

infeccion_ps<- read.table("infeccion_ps.txt", header=TRUE)
summary(infeccion_ps)

```

```

##      NoInfectados      Infectados      Inmunes      Nodos      Pinfeccion
## Min.       : 98.0    Min.       : 0.00    Min.      :0    Min.      :1    Min.      :0.0000
## 1st Qu.:254.0    1st Qu.: 0.00    1st Qu.:0    1st Qu.:1    1st Qu.:0.2375
## Median :296.0    Median : 4.00    Median :0    Median :1    Median :0.4750
## Mean   :262.6    Mean   : 37.35    Mean      :0    Mean      :1    Mean      :0.4750
## 3rd Qu.:300.0    3rd Qu.: 46.00    3rd Qu.:0    3rd Qu.:1    3rd Qu.:0.7125
## Max.   :300.0    Max.   :202.00    Max.      :0    Max.      :1    Max.      :0.9500
## Precuperacion  Pinmunidad  Iteraciones Model
## Min.       :0.3    Min.      :0    Min.      :10    BA:20
## 1st Qu.:0.3    1st Qu.:0    1st Qu.:10    ER:20
## Median :0.3    Median :0    Median :10    WS:20
## Mean   :0.3    Mean      :0    Mean      :10
## 3rd Qu.:0.3    3rd Qu.:0    3rd Qu.:10
## Max.   :0.3    Max.      :0    Max.      :10

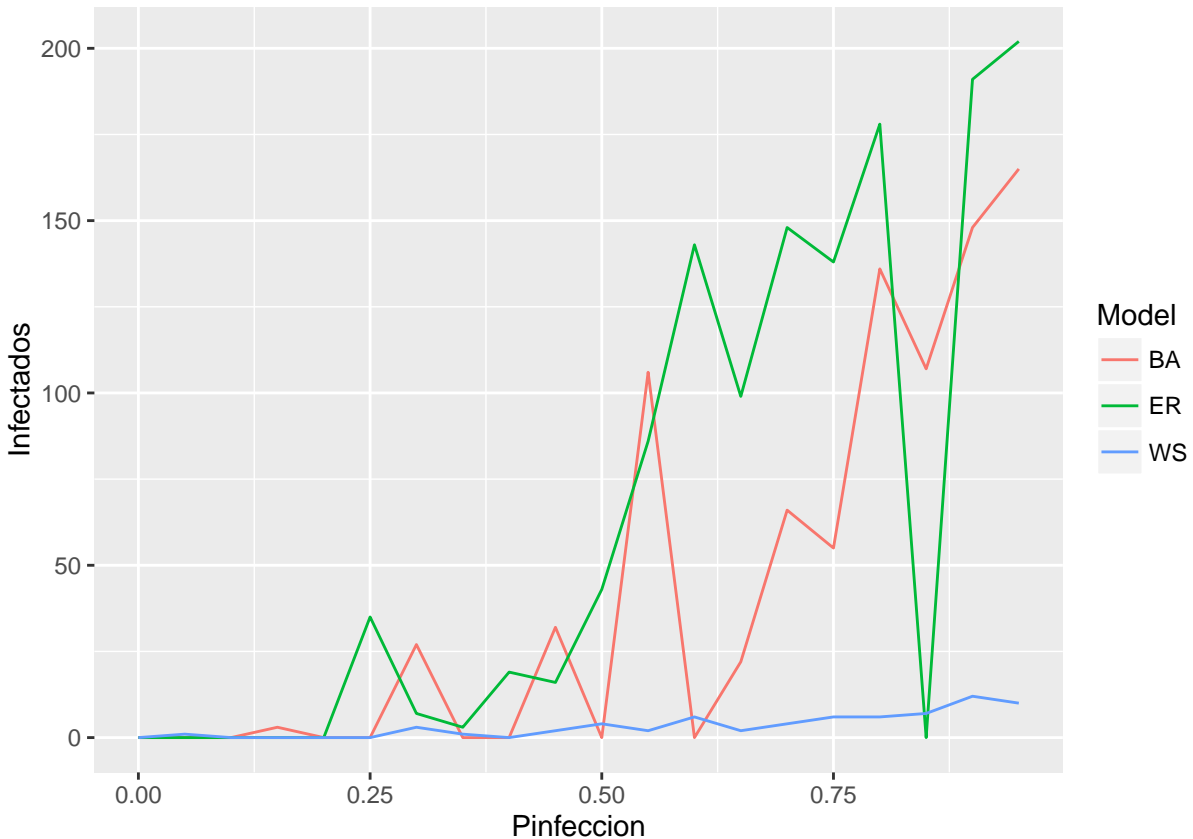
```

### Representación gráfica

```

ggplot(data = infeccion_ps, aes(x = Pinfeccion, y = Infectados, color = Model)) +
geom_line(aes(group = Model))

```



Se observa que cuando la probabilidad de infección supera la de recuperación, en este caso (0,3) el número de nodos infectados comienza a aumentar rápidamente en las redes Erdos Renyi y Barabasi. Sin embargo las redes Watts & Strogatz presentan mayor resistencia, porque apenas aumenta el número de nodos infectados con el aumento de la probabilidad de infección.

### 3.3 En función de la probabilidad de Recuperación (pr)

#### Script NetLogo

```
to infeccion_pr
  print ("NoInfectados Infectados Inmunes Nodos Pinfeccion Precuperacion Pinmunidad Iteraciones Model")

  ;; Erdos Renyi
  foreach (range 0 1 0.05)
  [ [pr] ->
    clear
    ER-RN 300 .01
    layout "spring"
    Spread 1 0.3 pr 0 10
    print (spread-summary-param 1 0.3 pr 0 10 "ER")
  ]

  ;; Barabasi & Albert
  foreach (range 0 1 0.05)
  [ [pr] ->
    clear
```

```

BA-PA 300 2 1
layout "spring"
Spread 1 0.3 pr 0 10
print (spread-summary-param 1 0.3 pr 0 10 "BA")
]

;; Watts & Strogatz
foreach (range 0 1 0.05)
[ [pr] ->
  clear
  WS 300 2 .01
  layout "spring"
  Spread 1 0.3 pr 0 10
  print (spread-summary-param 1 0.3 pr 0 10 "WS")
]
end

```

### Carga de los datos

```

infeccion_pr<- read.table("infeccion_pr.txt", header=TRUE)
summary(infeccion_pr)

```

```

##      NoInfectados      Infectados      Inmunes      Nodos      Pinfeccion
## Min.       :113.0    Min.       : 0.00    Min.       :0    Min.       :1    Min.       :0.3
## 1st Qu.:297.8    1st Qu.: 0.00    1st Qu.:0    1st Qu.:1    1st Qu.:0.3
## Median :300.0    Median : 0.00    Median :0    Median :1    Median :0.3
## Mean   :289.1    Mean   : 10.88    Mean   :0    Mean   :1    Mean   :0.3
## 3rd Qu.:300.0    3rd Qu.: 2.25    3rd Qu.:0    3rd Qu.:1    3rd Qu.:0.3
## Max.   :300.0    Max.   :187.00    Max.   :0    Max.   :1    Max.   :0.3
## Precuperacion      Pinmunidad Iteraciones Model
## Min.       :0.0000    Min.       :0    Min.       :10    BA:20
## 1st Qu.:0.2375    1st Qu.:0    1st Qu.:10    ER:20
## Median :0.4750    Median :0    Median :10    WS:20
## Mean   :0.4750    Mean   :0    Mean   :10
## 3rd Qu.:0.7125    3rd Qu.:0    3rd Qu.:10
## Max.   :0.9500    Max.   :0    Max.   :10

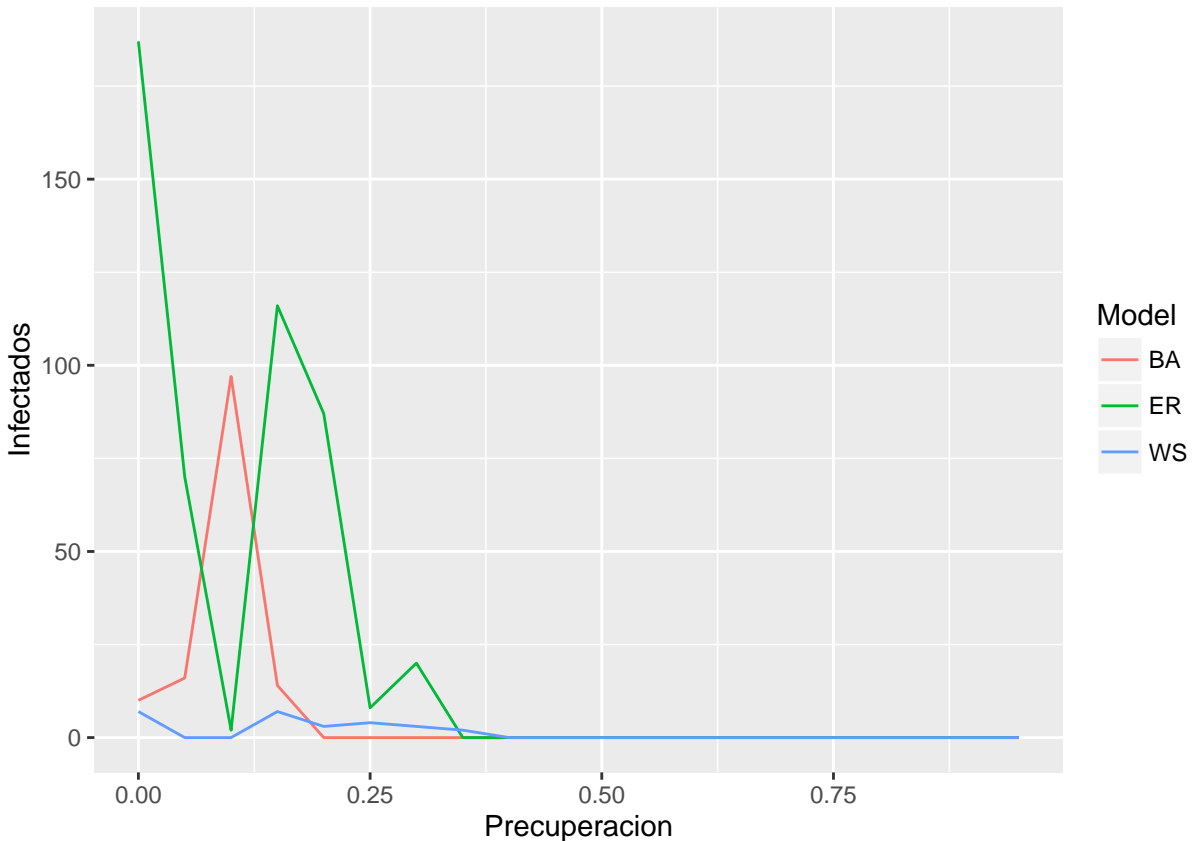
```

### Representación gráfica

```

ggplot(data = infeccion_pr, aes(x = Precuperacion, y = Infectados, color = Model)) +
geom_line(aes(group = Model))

```



En la gráfica anterior se muestra la influencia de la probabilidad de recuperación de una red frente al número de nodos infectados. De la gráfica se puede extraer al igual que en el experimento anterior (nodos infectados vs probabilidad de infección) que las redes Watts & Strogatz son mucho más resistentes, apenas existe variación o influencia frente a los nodos afectados, cuando la probabilidad de recuperación  $>$  probabilidad de infección el número de nodos infectados es 0. Sin embargo, en las redes Barabasi y Erdos Renyi si que existe una bajada brusca del número de nodos infectados con el aumento de la probabilidad de recuperación, especialmente en Erdos Renyi. Tanto en Erdos Renyi como en Barabasi, aunque son redes permeables que se contagian rápidamente, si se consigue una probabilidad de recuperación  $>$  probabilidad de infección el número de nodos infectados rápidamente será prácticamente 0.

### 3.4 En función de la probabilidad de Inmunidad tras recuperación (pin)

#### Script NetLogo

```
to infeccion_pin
  print ("NoInfectados Infectados Inmunes Nodos Pinfeccion Precuperacion Pinmunidad Iteraciones Model")

  ;; Erdos Renyi
  foreach (range 0 1 0.05)
  [ [pin] ->
    clear
    ER-RN 300 .01
    layout "spring"
    Spread 1 0.3 0.3 pin 10
    print (spread-summary-param 1 0.3 0.3 pin 10 "ER")
  ]
end
```

```

;; Barabasi & Albert
foreach (range 0 1 0.05)
[ [pin] ->
  clear
  BA-PA 300 2 1
  layout "spring"
  Spread 1 0.3 0.3 pin 10
  print (spread-summary-param 1 0.3 0.3 pin 10 "BA")
]

;; Watts & Strogatz
foreach (range 0 1 0.05)
[ [pin] ->
  clear
  WS 300 2 .01
  layout "spring"
  Spread 1 0.3 0.3 pin 10
  print (spread-summary-param 1 0.3 0.3 pin 10 "WS")
]
end

```

### Carga de los datos

```

infeccion_pin<- read.table("infeccion_pin.txt", header=TRUE)
summary(infeccion_pin)

```

```

##      NoInfectados      Infectados      Inmunes      Nodos
## Min.      :245.0    Min.      : 0.00    Min.      : 0.000    Min.      :1
## 1st Qu.:296.0    1st Qu.: 0.00    1st Qu.: 0.000    1st Qu.:1
## Median :298.0    Median : 0.00    Median : 1.000    Median :1
## Mean    :294.8    Mean    : 1.75    Mean    : 3.417    Mean     :1
## 3rd Qu.:299.0    3rd Qu.: 1.00    3rd Qu.: 3.000    3rd Qu.:1
## Max.    :300.0    Max.    :24.00    Max.    :31.000    Max.     :1
##      Pinfeccion  Precuperacion  Pinmunidad      Iteraciones Model
## Min.      :0.3    Min.      :0.3    Min.      :0.0000    Min.      :10    BA:20
## 1st Qu.:0.3    1st Qu.:0.3    1st Qu.:0.2375    1st Qu.:10    ER:20
## Median :0.3    Median :0.3    Median :0.4750    Median :10    WS:20
## Mean     :0.3    Mean     :0.3    Mean     :0.4750    Mean      :10
## 3rd Qu.:0.3    3rd Qu.:0.3    3rd Qu.:0.7125    3rd Qu.:10
## Max.     :0.3    Max.     :0.3    Max.     :0.9500    Max.      :10

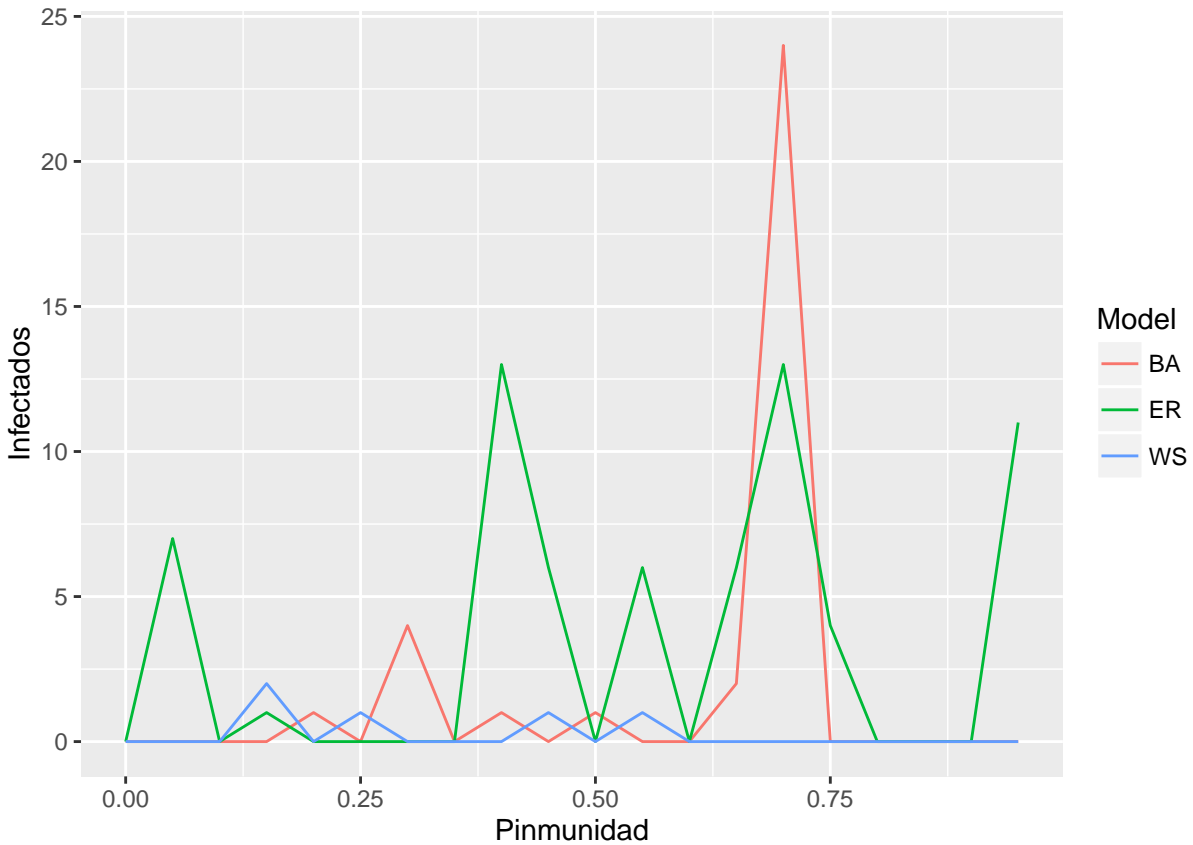
```

### Representación gráfica

```

ggplot(data = infeccion_pin, aes(x = Pinmunidad, y = Infectados, color = Model)) +
  geom_line(aes(group = Model))

```



En este caso, la probabilidad de que un nodo se convierta en inmune tras recuperarse es distinta de cero, esto hace que los nodos infectados no crezcan tan rápido y que no lleguen a valores como los vistos en las gráficas anteriores (alrededor de 150). En este caso el máximo de nodos infectados se tiene con las redes de Barabasi y no supera los 25.

En conclusión vemos que el modelo WS es muy resistente frente a contagios para cualquier. Sin embargo los modelos Barabasi y Erdos presentan una alta permeabilidad y cuando la probabilidad de recuperación es menor que la probabilidad de contagio el número de nodos infectados crece rápidamente. Se puede mejorar la situación si la probabilidad de un nodo de convertirse en inmune tras recuperarse es distinta de cero.