

# Analytics en Redes sociales - Fase 3

Daniel Florez - Jeison Díaz - Juan Guerrero - Wilson Lopez

El código se puede encontrar en: <https://github.com/jddiaze/AnalyticsRedesSociales>

## Desarrollo

**Cargar las librerías** Importar las librerías a usar.

```
library(rjson)
library(tidyverse)
library(sna)
library(igraph)
library(jsonlite)
library(purrr)
library(data.table)
library(readxl)
library(dplyr)
```

Cargar los datos iniciales de arcos y nodos

```
# Cargando los datos de arcos y nodos
ubicacion_edges<-"Arcos.xlsx"
arcos<-read_excel(ubicacion_edges)

colnames(arcos)<-c("from", "to")

nodos_file <-"Nodos.xlsx"
nodos <- read_excel(nodos_file)

arcos <- arcos[!is.na(arcos$from), ]

nodos <- nodos %>%
  filter(Id %in% arcos$from)

arcos <- arcos %>%
  filter(to %in% nodos$Id & from %in% nodos$Id)

n_grafo <- graph_from_edgelist(as.matrix(arcos))

numero_nodos <- vcount(n_grafo)
numero_arcos <- ecount(n_grafo)

print(paste("Número de nodos:", numero_nodos))
```

```
## [1] "Número de nodos: 649"
```

```
print(paste("Número de arcos:", numero_arcos))
```

```
## [1] "Número de arcos: 829"
```

Como puede observar en la red, esta se compone de 649 nodos y 829 arcos.

```
#Escriba su código aquí
y<- as.matrix(as_adjacency_matrix(n_grafo))
```

```

grado_total <- degree(n_grafo, mode = "all")

grado_total <- (grado_total-min(grado_total))/(max(grado_total)-min(grado_total))

x1 <- matrix(NA, nrow = nrow(y), ncol = nrow(y))

# Asignar nombres o índices de los nodos a las filas y columnas de x1
rownames(x1) <- rownames(y)
colnames(x1) <- colnames(y)

# Llenar la matriz x1 con la similitud entre los grados normalizados de los nodos
for (i in 1:nrow(y)) {
  for (j in 1:nrow(y)) {
    x1[i, j] <- 1 - abs(grado_total[i] - grado_total[j])
  }
}

bet <- betweenness(n_grafo)

bet <- (bet-min(bet))/(max(bet)-min(bet))

x2 <- matrix(NA, nrow = nrow(y), ncol = nrow(y))

# Asignar nombres o índices de los nodos a las filas y columnas de x1
rownames(x2) <- rownames(y)
colnames(x2) <- colnames(y)

# Llenar la matriz x1 con la similitud entre los grados normalizados de los nodos
for (i in 1:nrow(y)) {
  for (j in 1:nrow(y)) {
    x2[i, j] <- 1 - abs(bet[i] - bet[j])
  }
}

pr <- page_rank(n_grafo)$vector

pr <- (pr-min(pr))/(max(pr)-min(pr))

x3 <- matrix(NA, nrow = nrow(y), ncol = nrow(y))

# Asignar nombres o índices de los nodos a las filas y columnas de x1
rownames(x3) <- rownames(y)
colnames(x3) <- colnames(y)

# Llenar la matriz x1 con la similitud entre los grados normalizados de los nodos
for (i in 1:nrow(y)) {
  for (j in 1:nrow(y)) {
    x3[i, j] <- 1 - abs(pr[i] - pr[j])
  }
}

```

**Modelo QAP** Finalmente, estimemos la regresión QAP con estas dos matrices.

```
# Estimar la regresión QAP con la función netlogit. Debido a que son varios nodos, reduciremos la canti
qap_fit <-netlogit(y,list(x1,x2,x3),rep = 10)
```

```
# Ver los resultados esperados al finalizar las matrices calculadas
summary(qap_fit)
```

```
##
## Network Logit Model
##
## Coefficients:
##           Estimate   Exp(b)      Pr(<=b) Pr(>=b) Pr(>=|b|)
## (intercept) -1.1997025 0.30128383 0         1         0
## x1          -2.7199788 0.06587615 0         1         0
## x2          -0.8671444 0.42014963 0         1         0
## x3          -1.9554165 0.14150553 0         1         0
##
## Goodness of Fit Statistics:
##
## Null deviance: 583008.9 on 420552 degrees of freedom
## Residual deviance: 11571.37 on 420548 degrees of freedom
## Chi-Squared test of fit improvement:
##   571437.5 on 4 degrees of freedom, p-value 0
## AIC: 11579.37    BIC: 11623.16
## Pseudo-R^2 Measures:
## (Dn-Dr)/(Dn-Dr+dfn): 0.576052
## (Dn-Dr)/Dn: 0.9801523
## Contingency Table (predicted (rows) x actual (cols)):
##
##           0         1
## 0    419728      824
## 1           0         0
##
## Total Fraction Correct: 0.9980407
## Fraction Predicted 1s Correct: NaN
## Fraction Predicted 0s Correct: 0.9980407
## False Negative Rate: 1
## False Positive Rate: 0
##
## Test Diagnostics:
##
## Null Hypothesis: qap
## Replications: 10
## Distribution Summary:
##
##           (intercept)      x1      x2      x3
## Min      -2.957522 -2.068751 -1.822771 -2.003862
## 1stQ      -1.265542 -1.371149 -0.949268 -0.883549
## Median    -0.805444 -0.088125  0.062737  0.017815
## Mean      -0.425374  0.009024 -0.047678 -0.189708
## 3rdQ       0.031205  1.293408  0.641997  0.453887
## Max       2.390544  2.359323  2.256387  1.122889
```

Se puede evidenciar que los atributos betweeness y Page Rank son significativos.

**Conclusiones** En primer lugar, la significancia de “betweenness” indica que la centralidad de los nodos en la red, medida por la cantidad de intermediación que realizan, tiene un impacto estadísticamente significativo en el número de inversiones entre las organizaciones. Esto sugiere que identificar y apoyar organizaciones con una posición central en la red puede influir positivamente en la formación de conexiones e inversiones.

En segundo lugar, la significancia de “Page Rank” indica que la importancia relativa de los nodos en función de su influencia global también es un factor relevante. Aquellas organizaciones con un alto “Page Rank” están más conectadas y tienen un mayor impacto en la red, lo que se traduce en una mayor probabilidad de recibir inversiones.

En resumen, recomendamos a los fondos inversionistas considerar no solo la cantidad de conexiones de una organización, sino también su posición estratégica en términos de intermediación y su impacto global en la red. Invertir en organizaciones con una alta centralidad y una influencia significativa puede aumentar las posibilidades de éxito y maximizar el impacto de las inversiones en el contexto de esta red específica.