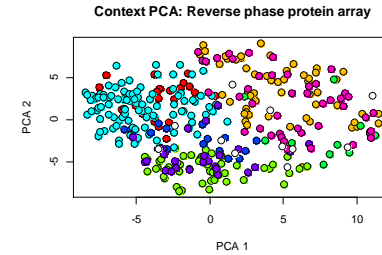
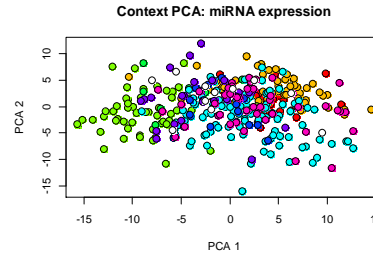
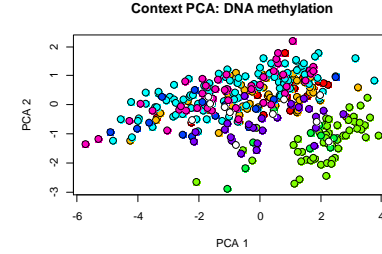
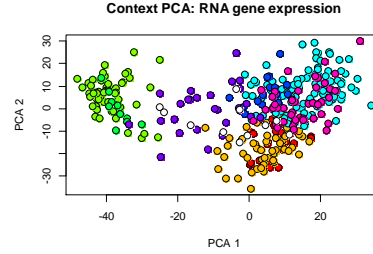


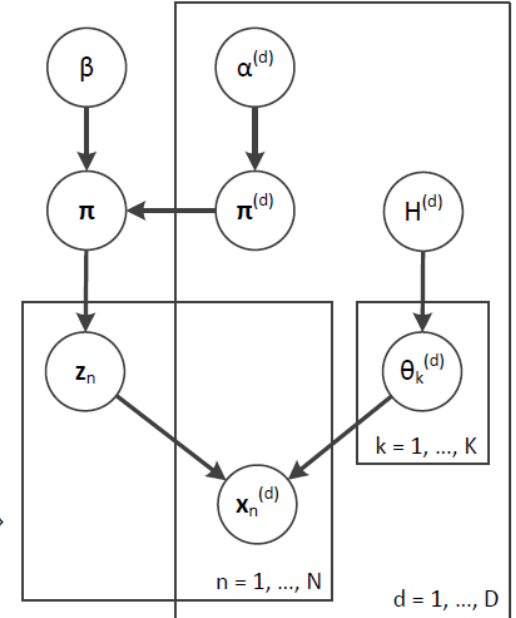
# Understanding social networks with F#

Evelina Gabasova  
@evelgab

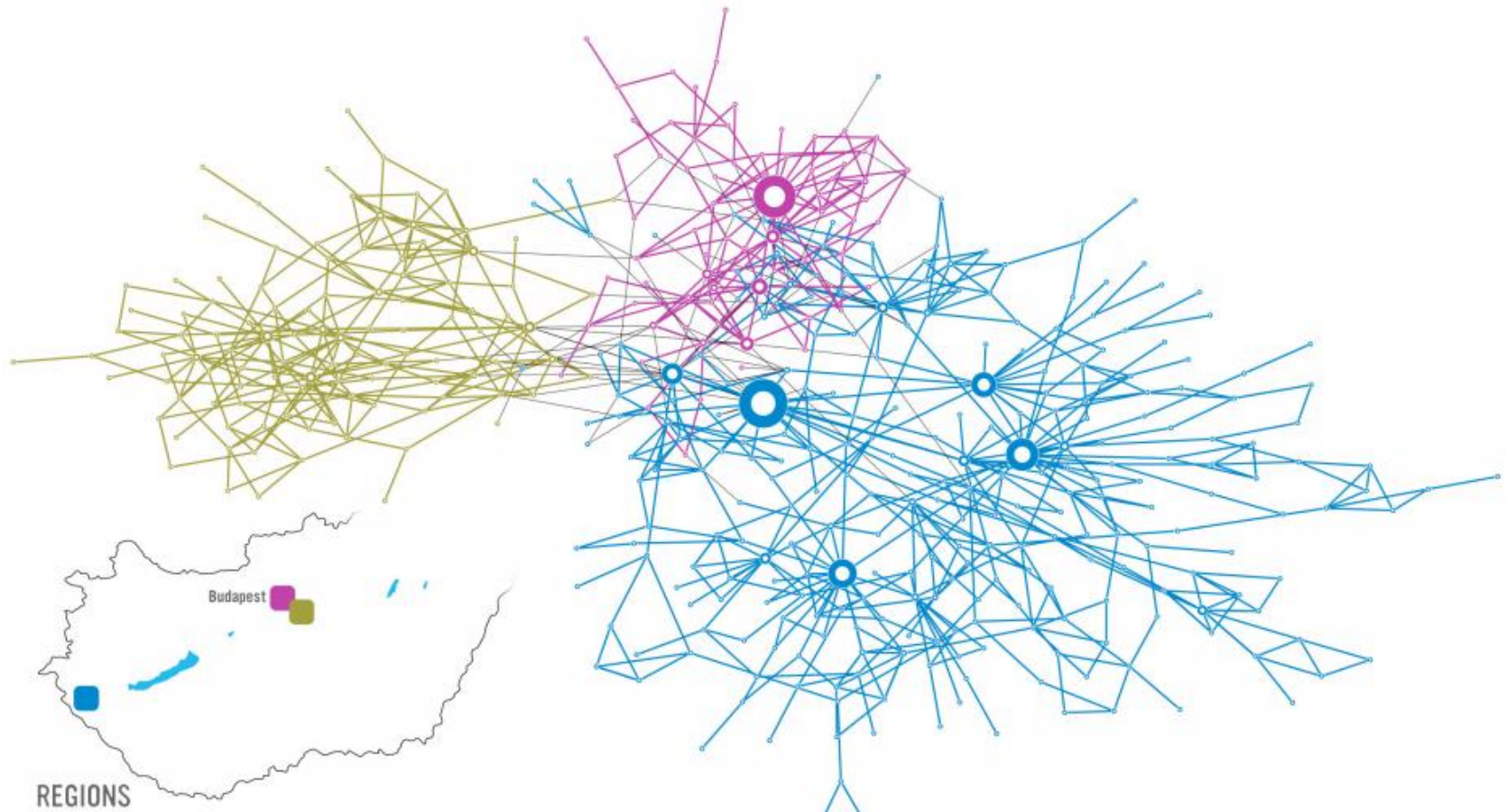
evelinag.com  
@evelgab



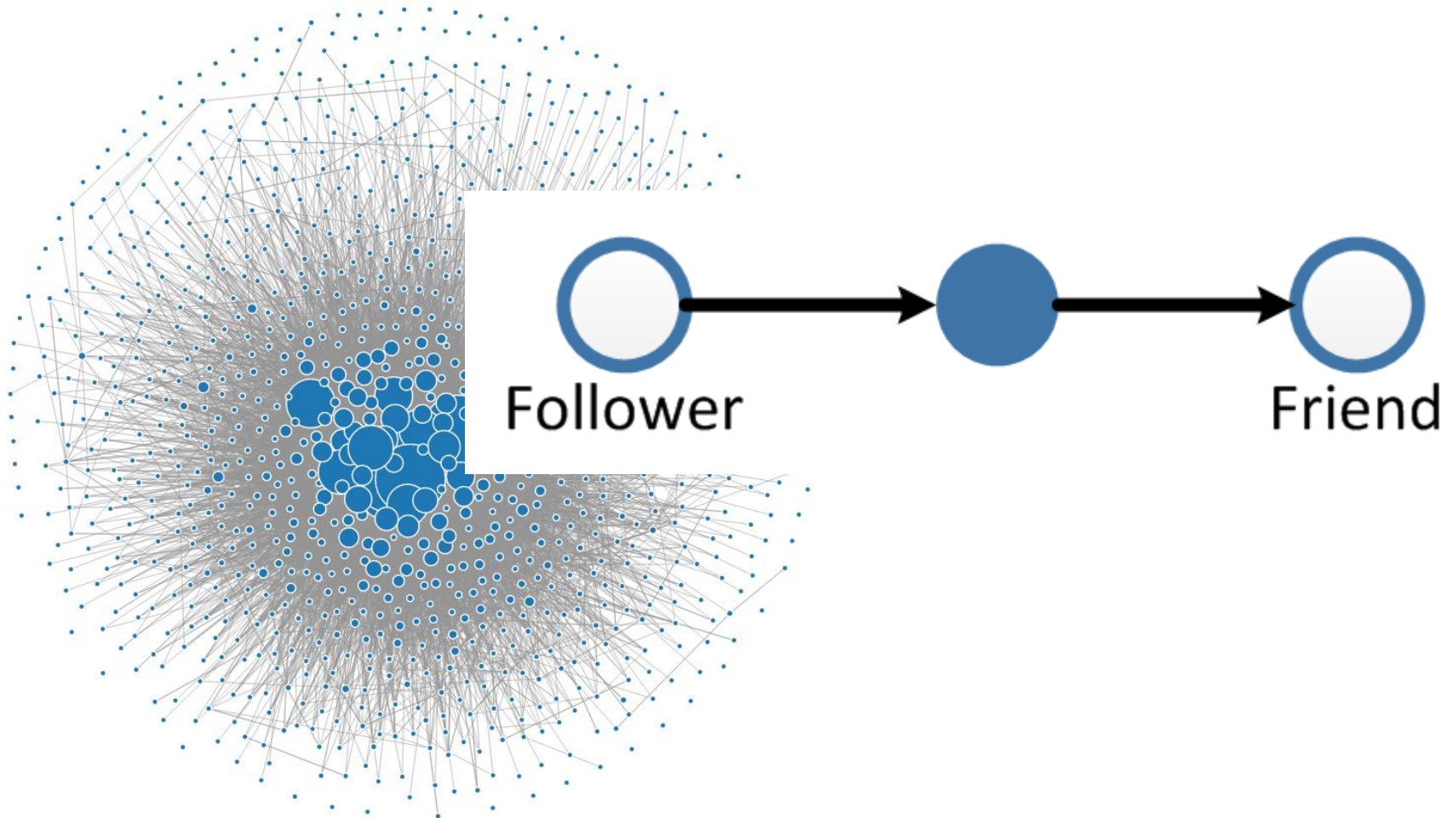
$$\begin{aligned}
 & p\left(\pi^{(d)} \mid \alpha_d, \beta, \pi\right) \\
 & \propto p\left(\pi^{(d)} \mid \alpha_d\right) p\left(\pi \mid \pi^{(1)}, \dots, \pi^{(D)}, \beta\right) \\
 & = \left\{ \frac{\Gamma\left(K^{(d)} \alpha_d\right)}{\left(\Gamma\left(\alpha_d\right)\right)^{K^{(d)}}} \prod_{k=1}^{K^{(d)}}\left(\pi_k^{(d)}\right)^{\alpha_d-1} \right\} \times \\
 & \quad \times \left\{ \frac{\Gamma\left(\sum_{i_1=1}^{K^{(1)}} \dots \sum_{i_D=1}^{K^{(D)}}\left[\beta \times \pi_{i_1}^{(1)} \times \dots \times \pi_{i_D}^{(D)}\right]\right)}{\prod_{i_1=1}^{K^{(1)}} \dots \prod_{i_D=1}^{K^{(D)}} \Gamma\left(\beta \times \pi_{i_1}^{(1)} \times \dots \times \pi_{i_D}^{(D)}\right)} \prod_{i_1=1}^{K^{(1)}} \dots \prod_{i_D=1}^{K^{(D)}}\left(\pi_{i_1, \dots, i_D}\right)^{\beta \pi_{i_1}^{(1)} \dots \pi_{i_D}^{(D)}-1} \right\} \\
 & = \frac{\Gamma\left(K^{(d)} \alpha_d\right)}{\left(\Gamma\left(\alpha_d\right)\right)^{K^{(d)}}} \times \frac{\Gamma(\beta)}{\prod_{i_1=1}^{K^{(1)}} \dots \prod_{i_D=1}^{K^{(D)}} \Gamma\left(\beta \times \pi_{i_1}^{(1)} \times \dots \times \pi_{i_D}^{(D)}\right)} \times \\
 & \quad \times \left\{ \prod_{k=1}^{K^{(d)}}\left(\pi_k^{(d)}\right)^{\alpha_d-1} \right\} \left\{ \prod_{i_1=1}^{K^{(1)}} \dots \prod_{i_D=1}^{K^{(D)}}\left(\pi_{i_1, \dots, i_D}\right)^{\beta \pi_{i_1}^{(1)} \dots \pi_{i_D}^{(D)}-1} \right\} \\
 & \propto \frac{\Gamma\left(K^{(d)} \alpha_d\right)}{\left(\Gamma\left(\alpha_d\right)\right)^{K^{(d)}}} \left\{ \prod_{k=1}^{K^{(d)}}\left(\pi_k^{(d)}\right)^{\alpha_d-1} \right\} \left\{ \prod_{i_1=1}^{K^{(1)}} \dots \prod_{i_D=1}^{K^{(D)}} \frac{1}{\Gamma\left(\beta \times \pi_{i_1}^{(1)} \times \dots \times \pi_{i_D}^{(D)}\right)}\left(\pi_{i_1, \dots, i_D}\right)^{\beta \pi_{i_1}^{(1)} \dots \pi_{i_D}^{(D)}-1} \right\}
 \end{aligned} \tag{18}$$



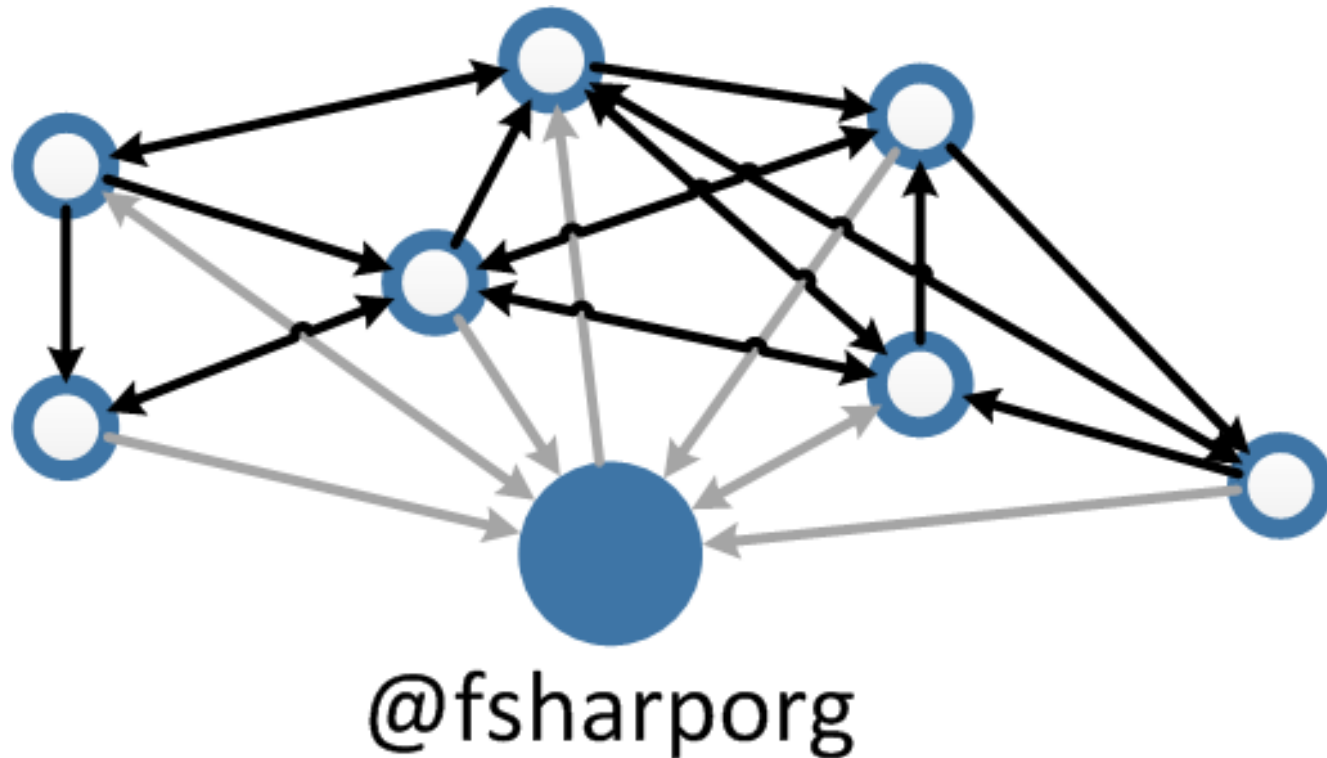
# Insights from social networks



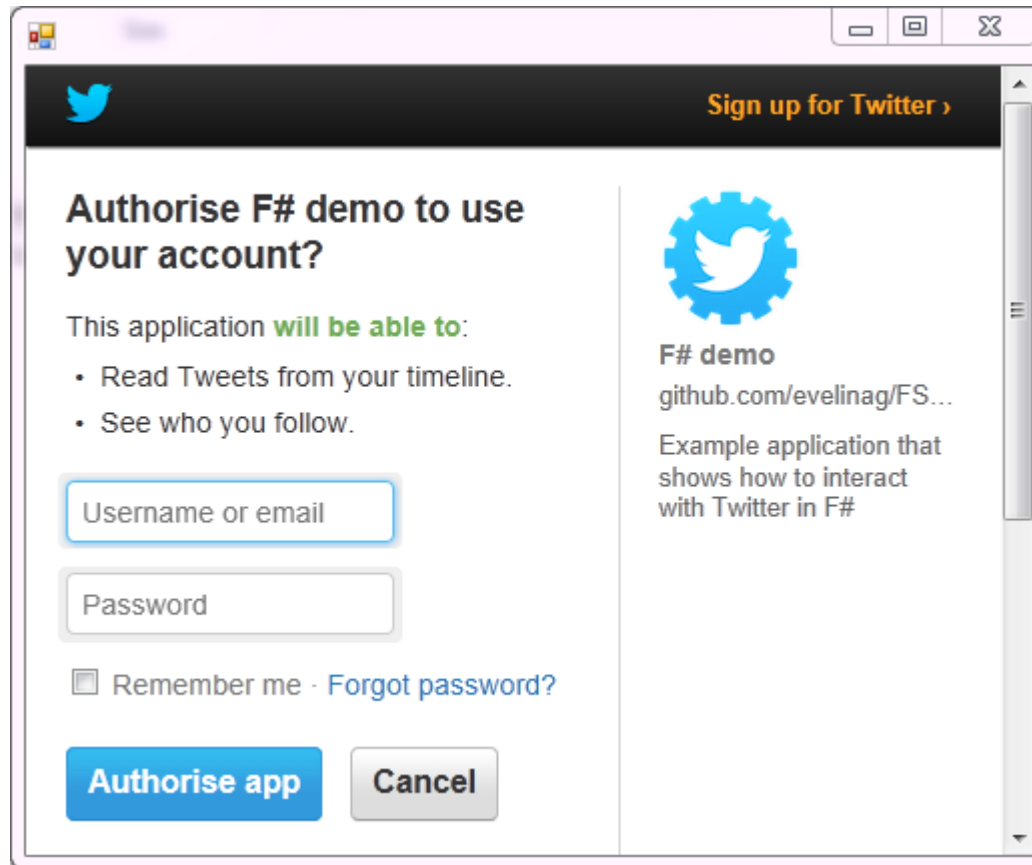
# Twitter network



# How large is your ego?



# Connecting to twitter



A screenshot of a web browser window showing the Twitter authorization page for an application named 'F# demo'. The window has a standard macOS-style title bar with minimize, maximize, and close buttons. The Twitter logo is in the top left, and a 'Sign up for Twitter' link is in the top right. The main heading asks 'Authorise F# demo to use your account?'. Below this, it states 'This application will be able to:' followed by a list of permissions: 'Read Tweets from your timeline.' and 'See who you follow.'. There are input fields for 'Username or email' and 'Password'. A checkbox for 'Remember me' is present, along with a link for 'Forgot password?'. At the bottom are two buttons: 'Authorise app' (blue) and 'Cancel' (grey). On the right side, there is a blue gear icon with a white Twitter bird inside, followed by the text 'F# demo', the GitHub URL 'github.com/evelinag/FS...', and a description: 'Example application that shows how to interact with Twitter in F#'. A vertical scrollbar is visible on the right edge of the content area.

Sign up for Twitter ›

## Authorise F# demo to use your account?

This application **will be able to:**


- Read Tweets from your timeline.
- See who you follow.

Username or email

Password

☐ Remember me · [Forgot password?](#)

**Authorise app** Cancel



**F# demo**  
github.com/evelinag/FS...  
Example application that shows how to interact with Twitter in F#



# Interaction rates

F# 86.1 %

Erlang 85.9 %

Clojure 82.7 %

Haskell 75.7 %

Scala 75.6 %

# Interaction rates

F# 86.1 %

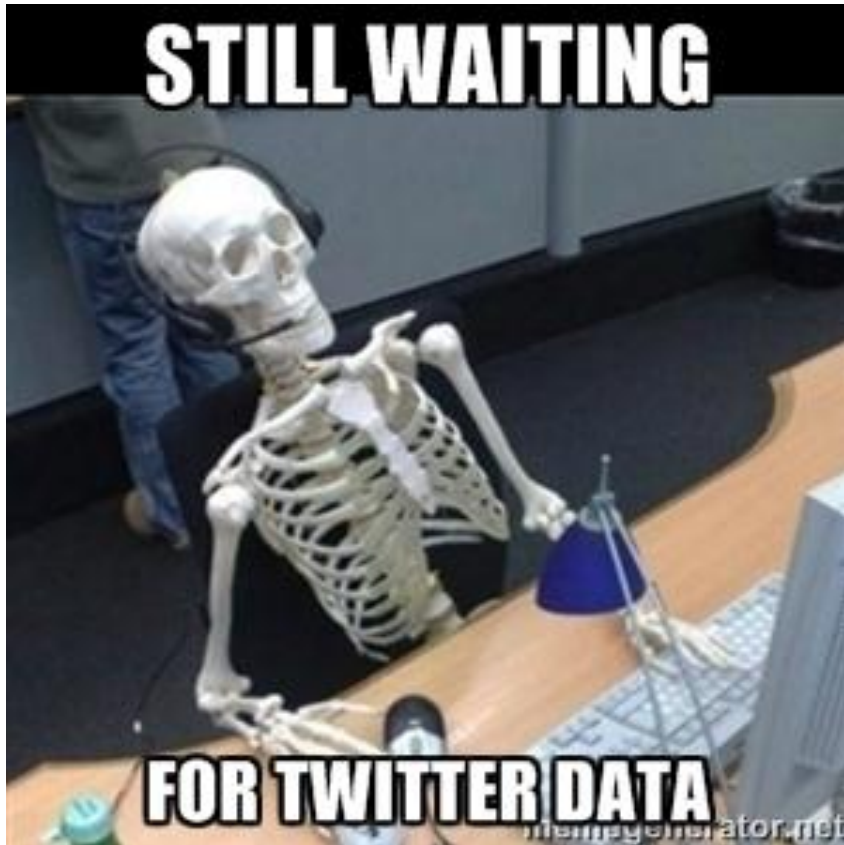
Erlang 85.8 %

Tweet functional  
programming

Scala 75.6 %



# Downloading data

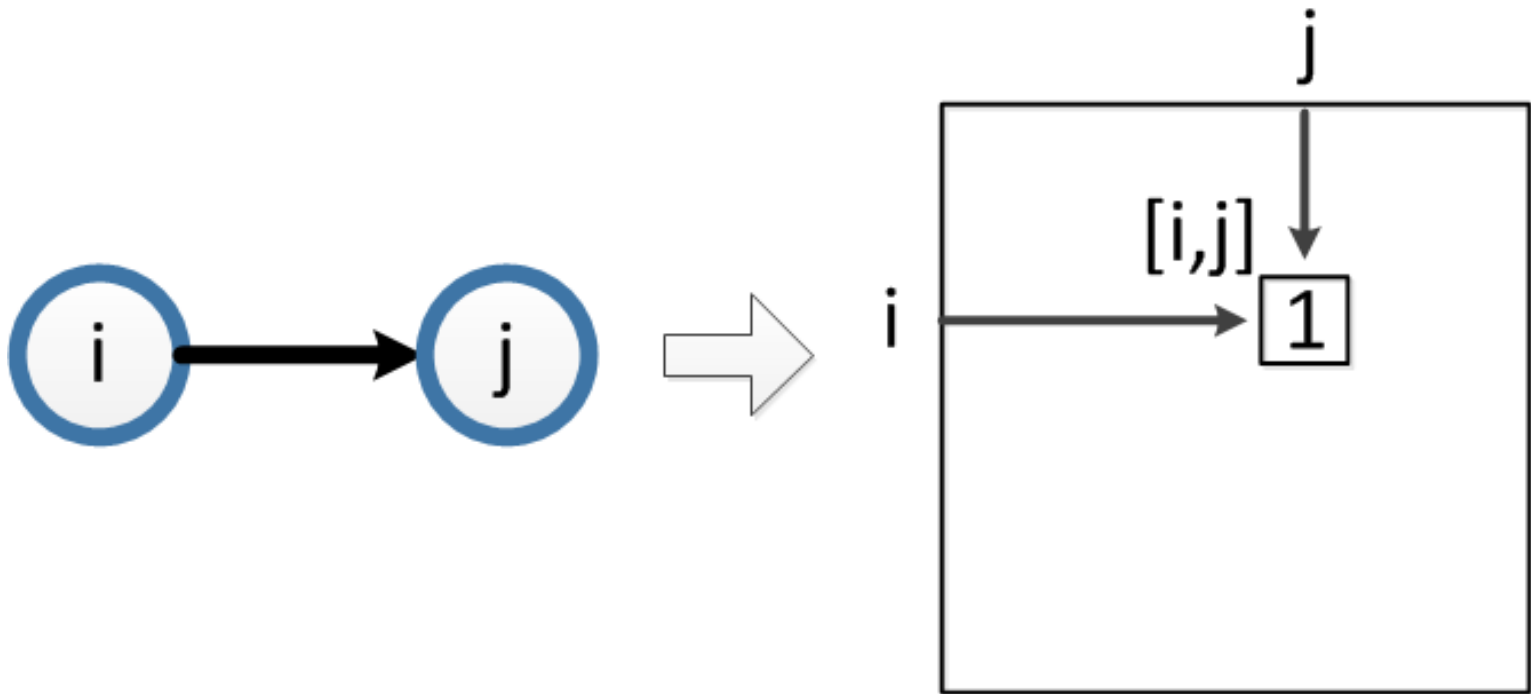


- 1) List of nodes
- 2) Connections between nodes

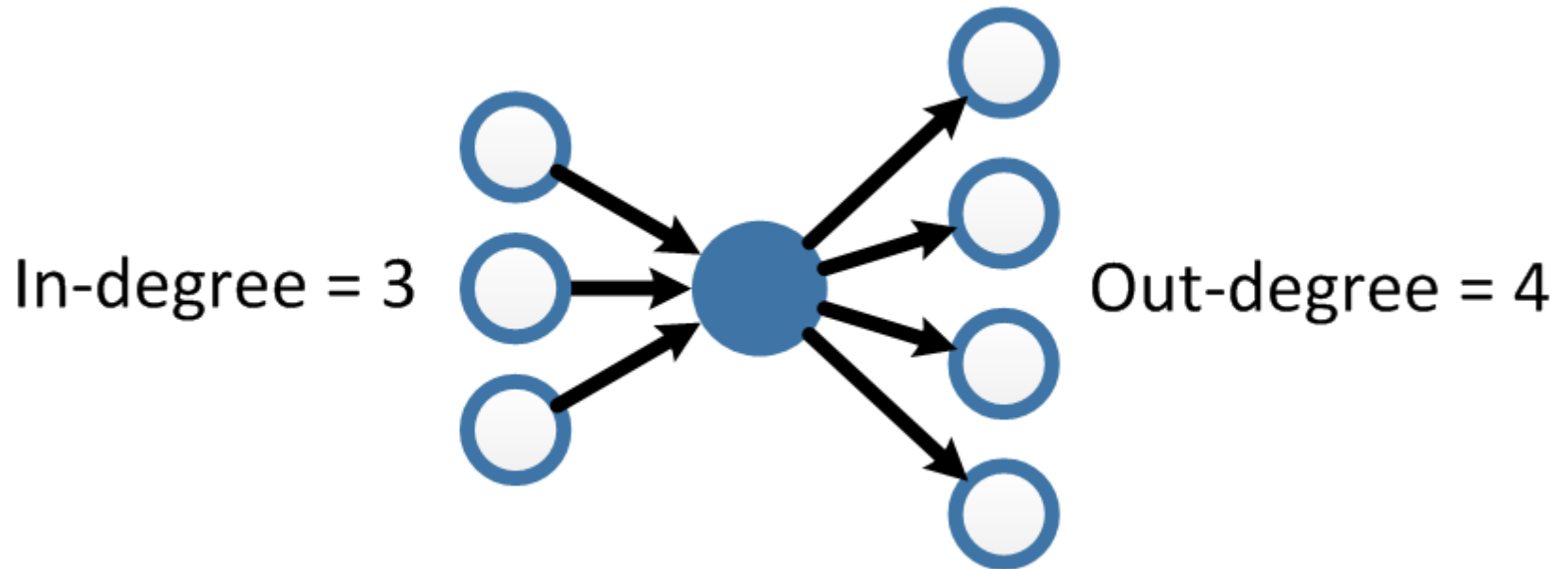
Twitter API allows only 15 requests every 15 minutes to list connections.



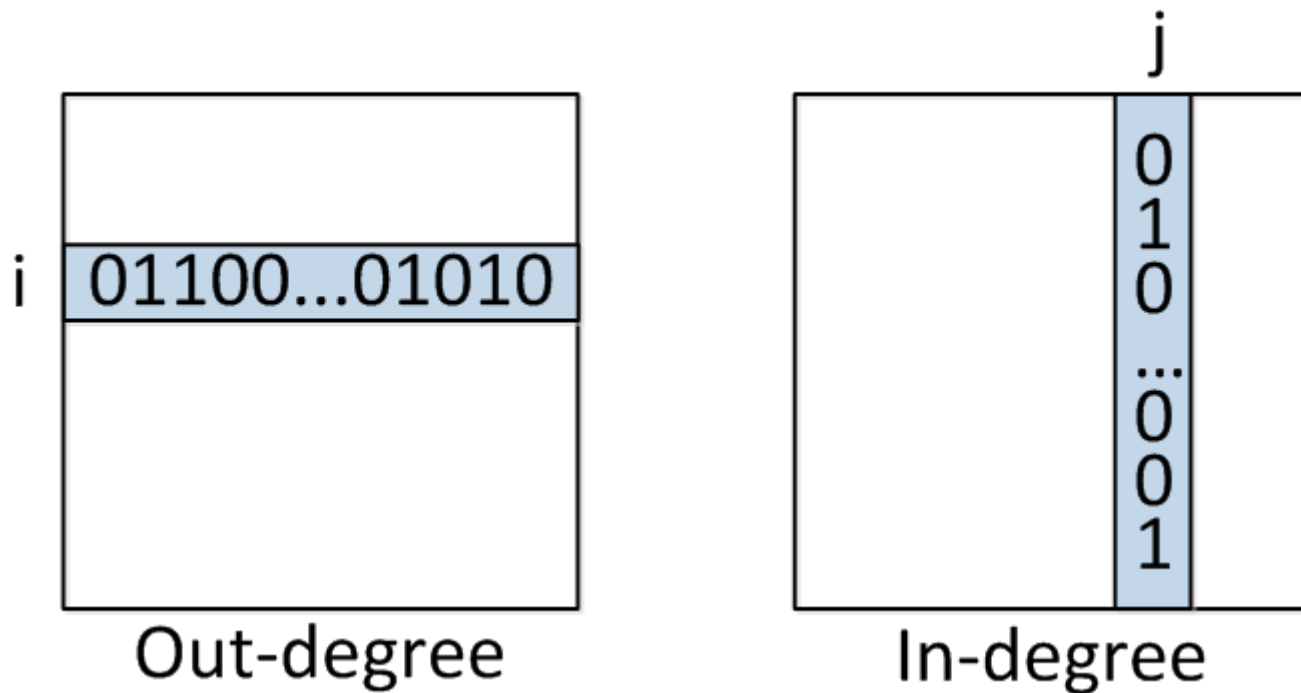
# Adjacency matrix



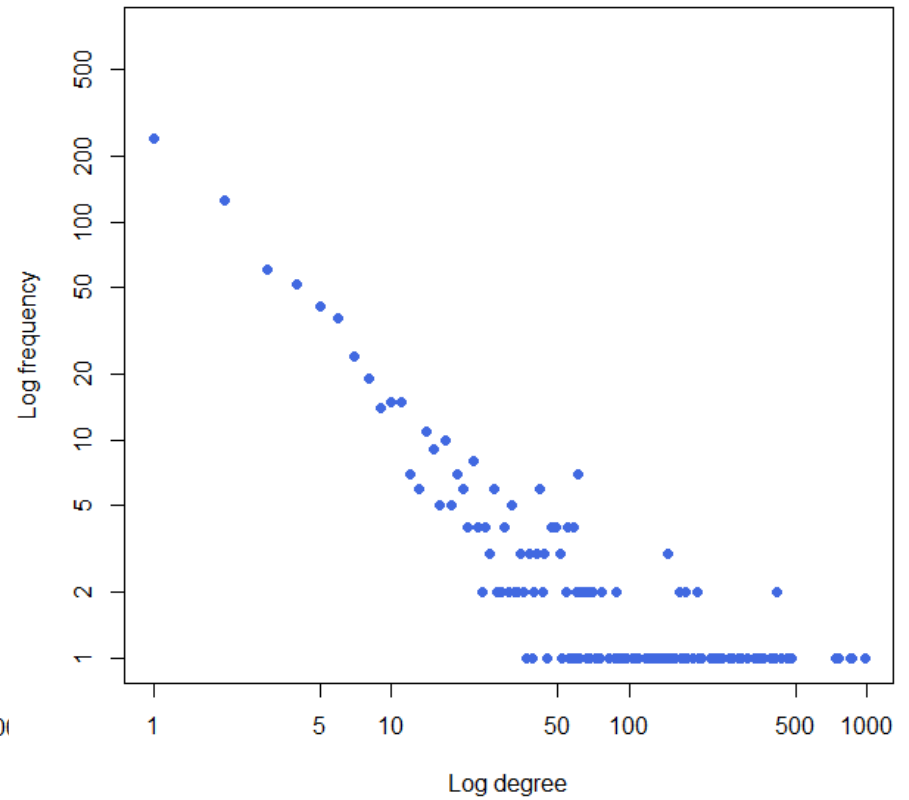
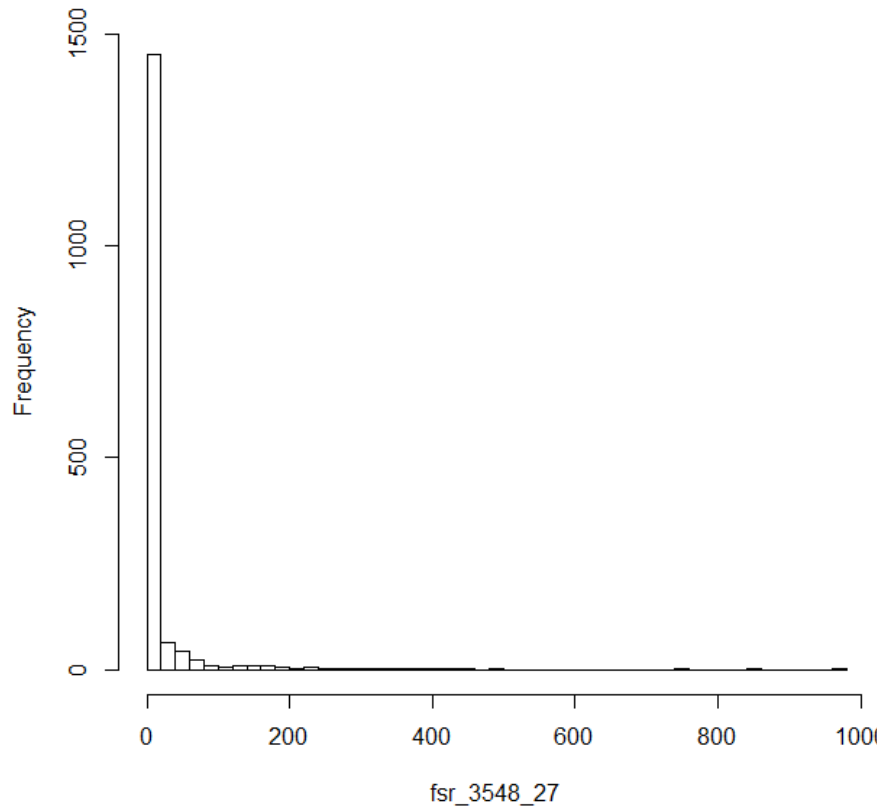
# Degrees



# Degrees



# Degree distribution

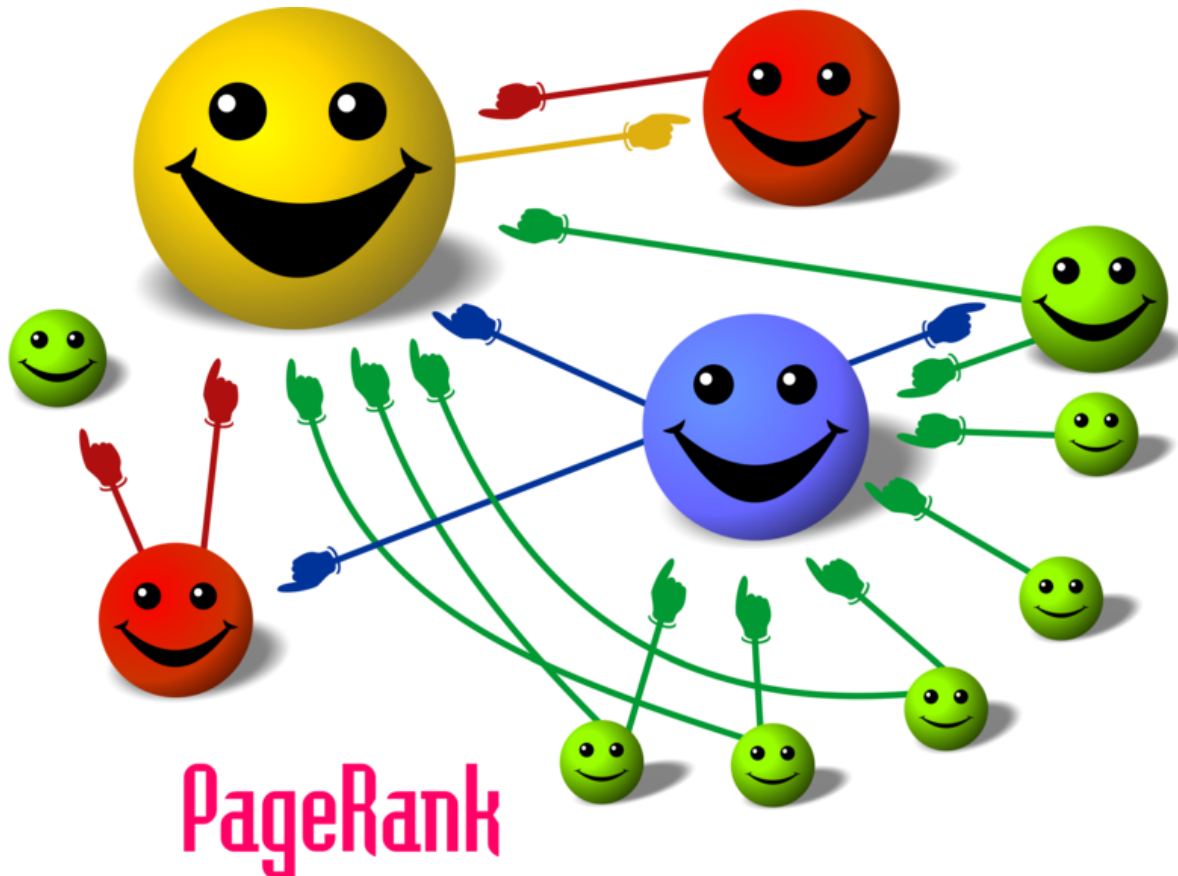


# Hubs in networks

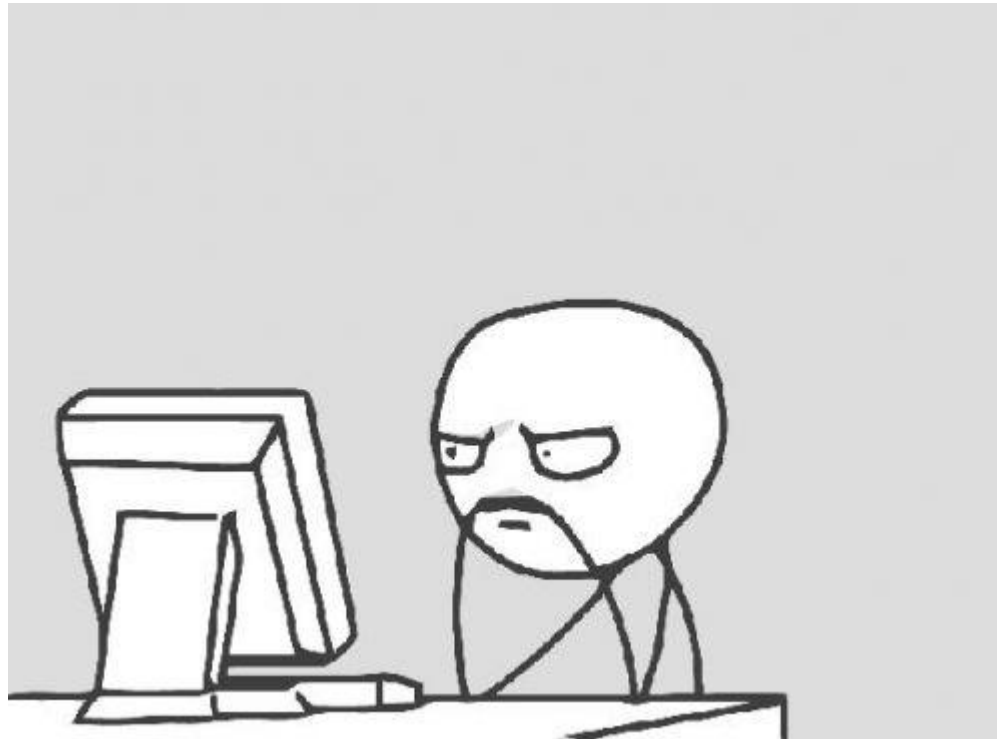
How to identify most  
**important** nodes in a network?

# Centrality with PageRank

Your followers are not created equal.

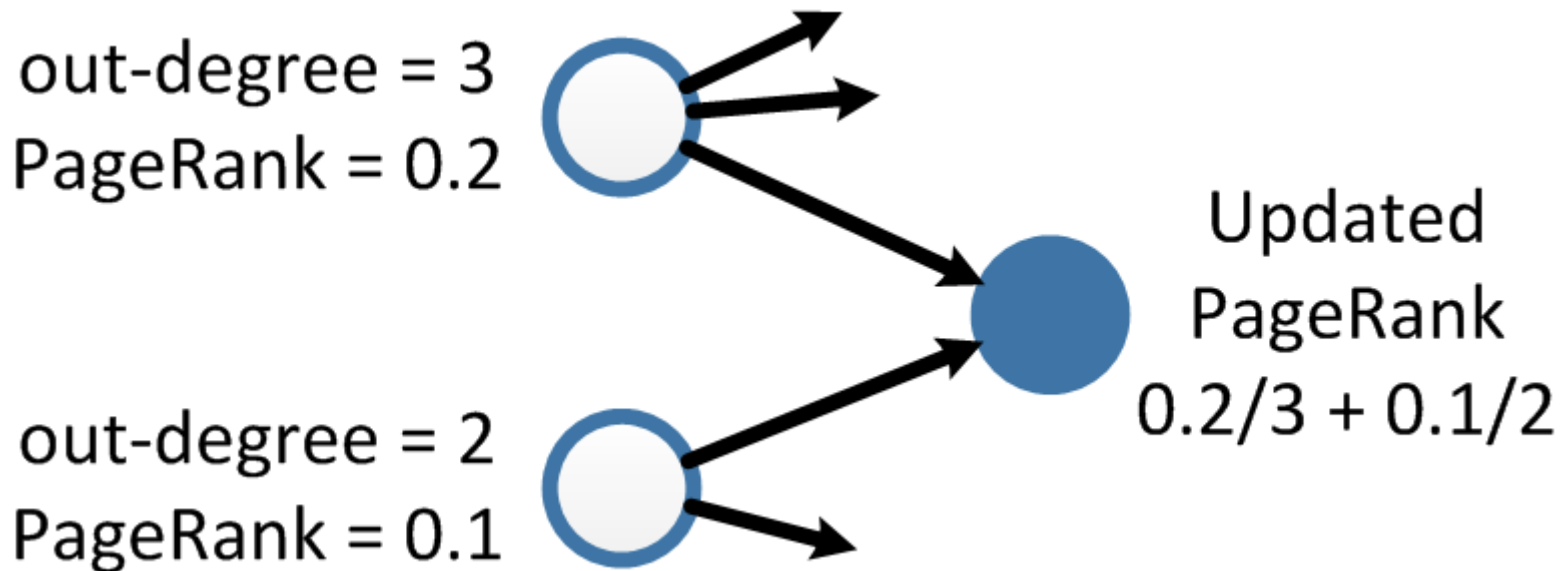


# Random surfer model

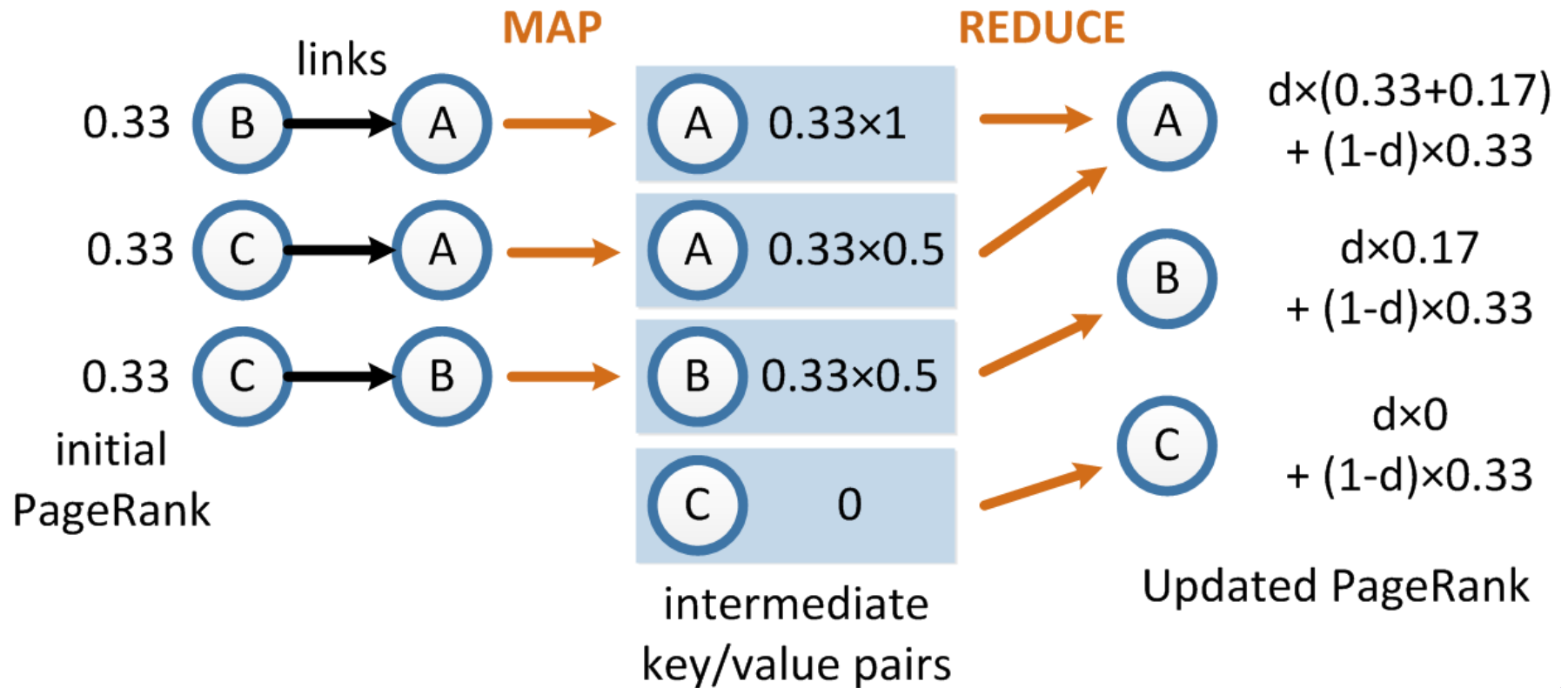
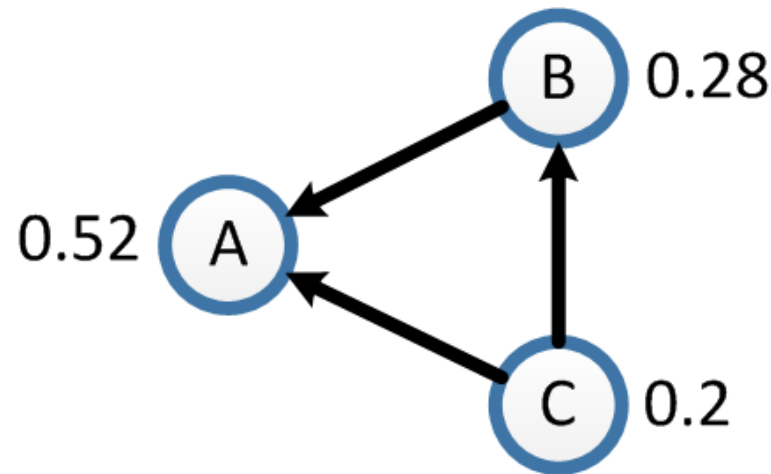




# Centrality with PageRank



+ random jumps



# PageRank changes

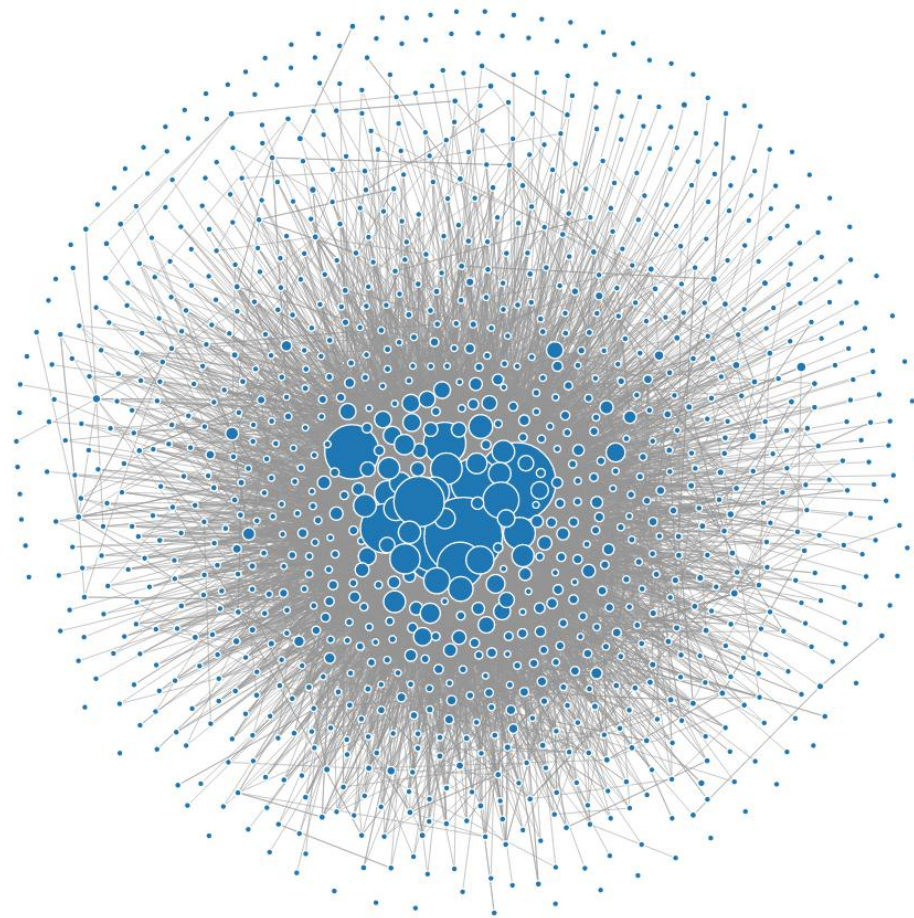
## February

1. migueldeicaza (0.033130)
2. dsyme (0.032783)
3. tomaspetricek (0.027756)
4. LincolnAtkinson (0.021993)
5. VisualFSharp (0.020233)
6. c4fsharp (0.019720)
7. rickasaurus (0.019189)
8. ptrelford (0.018099)
9. 1tgr (0.016525)
10. sforkmann (0.014970)

## September

1. dsyme (0.028640)
2. migueldeicaza (0.024808)
3. VisualFSharp (0.024479)
4. tomaspetricek (0.021066)
5. c4fsharp (0.019612)
6. rickasaurus (0.014272)
7. sforkmann (0.013471)
8. 1tgr (0.012768)
9. ptrelford (0.012669)
10. FSPowerTools (0.012113)

# Visualization with D3.js



Go play  
with data!

# Thank you

@evelgab

evelina@evelinag.com

<https://github.com/evelinag>

**fsharp.org**

**F# eXchange 2015**

17 April, London

