**Project – Pseudo-Code**

1. Predefine parameters:

* p: Part of missing nodes (error level) = p in P [???]
* q: Fraction of immunized nodes = 10%
* n: Number of nodes = 1000
* t: Transitivity = t in T [0.3; 0.155; 0.01] (🡪 High, Medium, Low)
* Outbreak parameters:
  + Beta: 0.95
  + Gamma: 1.00
* c: centrality measure (functions)
  + Degree
  + Betweenness
  + Eigenvector
  + PageRank

1. Generate true networks G

~~For each transitivity t in T~~

Generate 5000 synthetic networks G using configuration model with

* Properties of Colorado Springs Network
* Undirected
* Unweighted
* n nodes

**Robustness of the immunization strategy:**

1. Compute robustness of immunization strategy

For each centrality measure c

For each true network G

For each error level p (observed G’)

~~Generate 2000 observed networks G’ by removing p% of the nodes at random~~

For each observed network G’

Immunize q% of the nodes at random to obtain treated network G’’(random)

Simulate outbreak with parameters beta and gamma for G’’(random) to obtain outbreak size

For each node in G’

Compute centrality measure c

Immunize q% of the nodes with max c in G’ to obtain treated network G’’(c)

Simulate outbreak with parameters beta and gamma for G’’(c) to obtain outbreak size

Compute difference between outbreak sizes of G’(random) and G’(c)

Compute mean of differences (over all observed networks G’)

Compute variance of differences (over all observed networks G’)

Compute mean of differences (over all true networks G)

Compute variance of differences (over all true networks G)

**Robustness of the centrality measure:**

1. Compute robustness of the centrality measure

For each centrality measure c

For each error level p

For each true network G

For each node in G

Compute centrality measure c

Sort nodes in G decreasingly wrt. c and store in vector v1

Generate 2000 observed networks G’ by removing p% of the nodes at random

For each observed network G’

For each node in G’

Compute centrality measure c

Sort nodes in G’ decreasingly wrt. c and store in vector v2

Compute rank correlation (Kendall’s tau b) of v1 and v2

Compute mean of rank correlation (over all observed networks G’)

Compute variance of rank correlation (over all observed networks G’)

Compute mean of rank correlation (over all true networks G)

Compute variance of rank correlation (over all true networks G)

1. Plot results

Generate 4 plots (for each centrality measure c)

* X-axis: error level p
* Y-axis:
  + Mean and variance of differences over all true networks G
  + Mean and variance of rank correlation over all true networks G