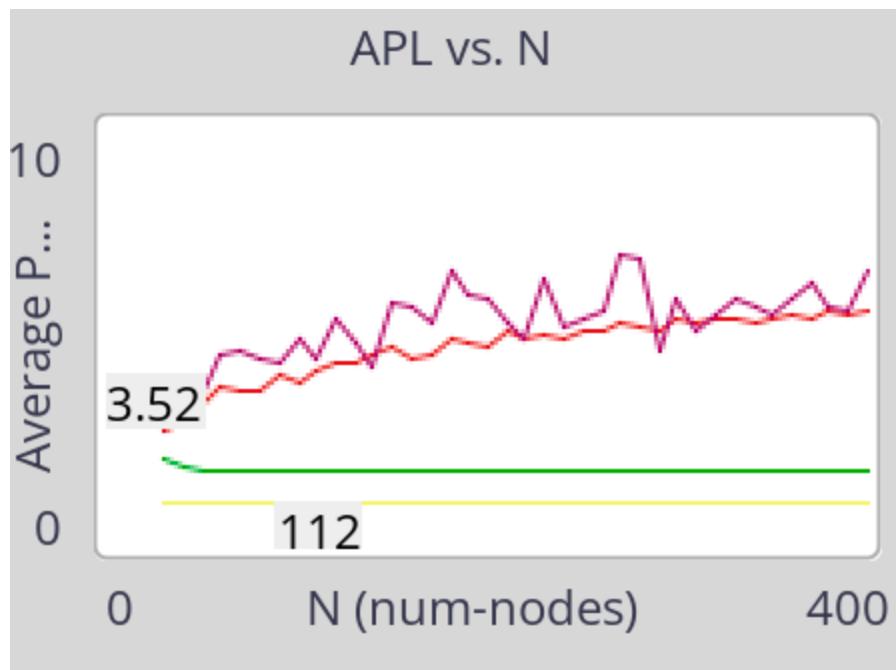


Challenge 3. Networks

APL Plot for different networks



Green color - Erdos-Renyi

Red color - small world

Yellow color - complete graph

Purple color - Barabasi-Albert

```
extensions[nw]

turtles-own [
  infected?
]

to setup
  clear-all
  ask patches [set pcolor white]
```

```

create-turtles num-nodes [
  set color blue
  set shape "circle"
]

reset-ticks
end

to random-graph
  refresh

  create-turtles num-nodes [
    set color blue
  ]
  ask turtles [
    if count other turtles with [link-neighbor? myself = false] > 0 [
      create-links-with other turtles with [link-neighbor? myself = false]
    ]
  ]
]

layout-circle turtles 15
tick
end

to erno-graph
  refresh
  nw:generate-random turtles links num-nodes p [ set color blue ]
  layout-circle turtles 15
  tick
end

to small-world-graph
  refresh
  nw:generate-watts-strogatz turtles links num-nodes 2 p [ fd 10 ]
  ask turtles [set color blue]

```

```

layout-circle turtles 15
tick

end

to barabasi-graph
  refresh
  nw:generate-preferential-attachment turtles links num-nodes 1 [ set color blue ]
end

layout-circle turtles 15
tick

end

to layoutspring
  repeat 30 [layout-spring turtles links 0.2 5 1]
end

to complete-graph
  refresh
  create-turtles num-nodes [
    set color blue
  ]
  ask turtles [
    if count other turtles with [link-neighbor? myself = false] > 0 [
      create-links-with other turtles with [link-neighbor? myself = false]
    ]
  ]
  layout-circle turtles 15
  tick

end

to launch-virus

```

```
ask turtles [
  set color blue
  set infected? false]
```

```
ask one-of turtles [
  set color red
  set infected? true
]
```

```
tick
end
```

to make-step-simple

```
ask turtles [
  if count link-neighbors with [color = red] > 0 [
    repeat count link-neighbors with [color = red] [
      if random-float 1 < transmission [
        set infected? true
      ]
    ]
  ]
]
```

```
ask turtles [
  if infected? = true [set color red]
]
update
tick
end
```

to make-step-complex

```
ask turtles [
  if count link-neighbors with [color = red] > 0 [
    if count link-neighbors with [color = red] / count link-neighbors > threshold [
      d [

```

```

    set color red
]
]
]

update
tick
end

to refresh
;clear-all
clear-turtles
clear-links
reset-ticks
ask patches [set pcolor white]

tick
end

to update
set-current-plot "Log-Log"
set-current-plot-pen "pen1"

let max-degree max [count link-neighbors] of turtles
let degree 1

while [degree <= max-degree] [
let matches turtles with [count link-neighbors = degree]
if any? matches [
plotxy (log (count matches) 10) (log degree 10)

]

set degree degree + 1
]

```

```

end

to-report calculate-apl
  nw:set-context turtles links

  let apl nw:mean-path-length

  if apl = -1 [
    report -1
  ]

  report apl
end

to run-apl-experiment
  let min-n 30
  let max-n 400
  let step-n 10

  set-current-plot "APL vs. N"
  clear-plot

  let current-n min-n
  while [current-n <= max-n] [
    set num-nodes current-n

    set-current-plot-pen "Erdos-Renyi"
    ernes-graph
    let apl-er calculate-apl
  ]

```

```

if apl-er != -1 [
    plotxy current-n apl-er
]

set-current-plot-pen "Small World"
small-world-graph
let apl-sw calculate-apl
if apl-sw != -1 [
    plotxy current-n apl-sw
]

set-current-plot-pen "Barabasi-Albert"
barabasi-graph
let apl-ba calculate-apl
if apl-ba != -1 [
    plotxy current-n apl-ba
]

; --- 4. Test Complete Graph ---
set-current-plot-pen "Complete Graph"
complete-graph
let apl-cg calculate-apl
if apl-cg != -1 [
    plotxy current-n apl-cg
]

; --- Increment N for the next loop ---
set current-n (current-n + step-n)
]

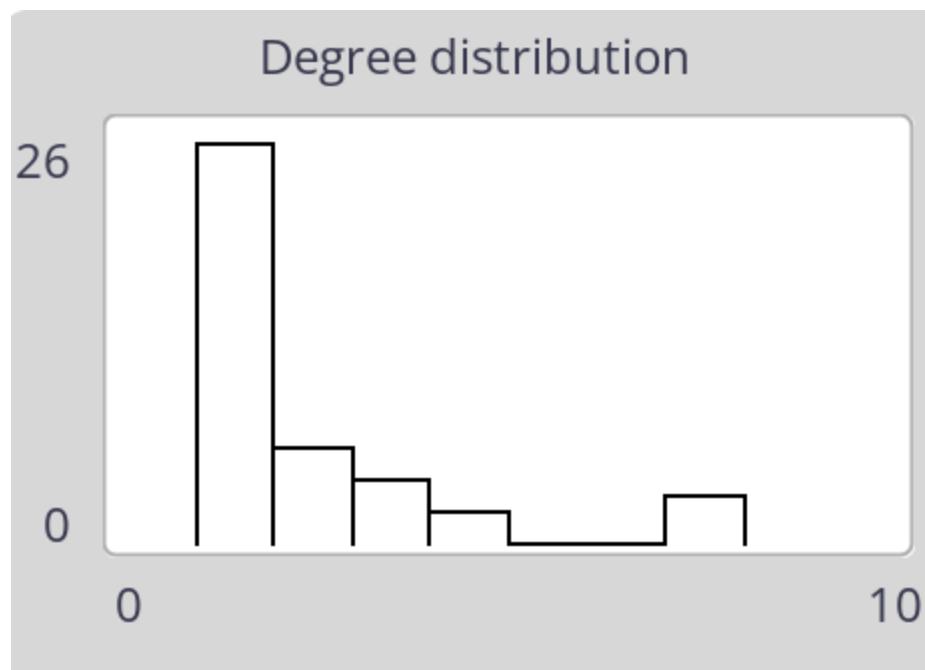
end

```

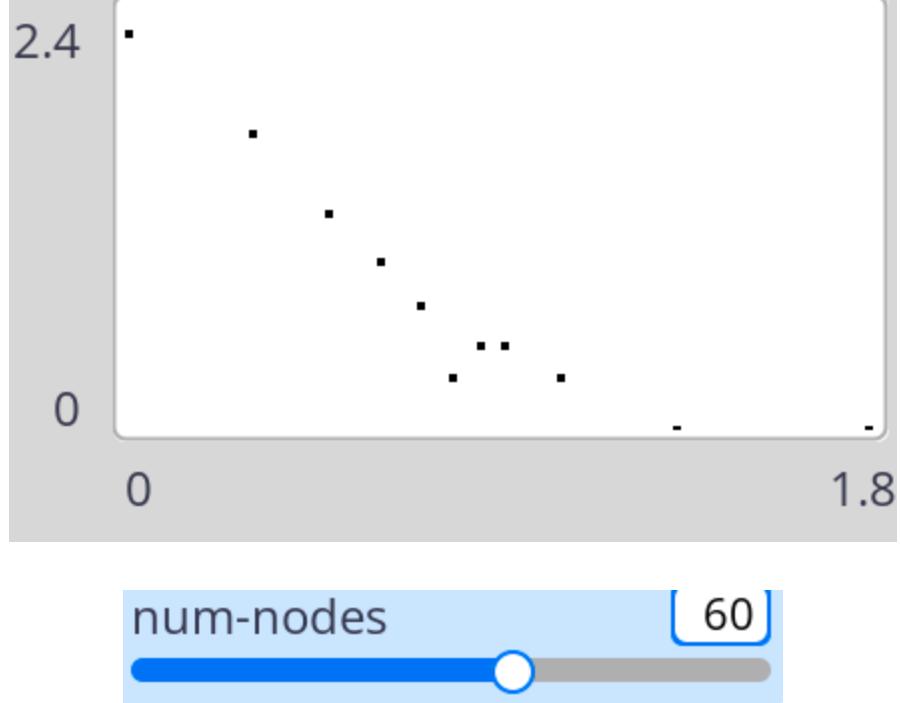
NetworksPractice.nlogox

Verify Exponential distribution for Barabasi model

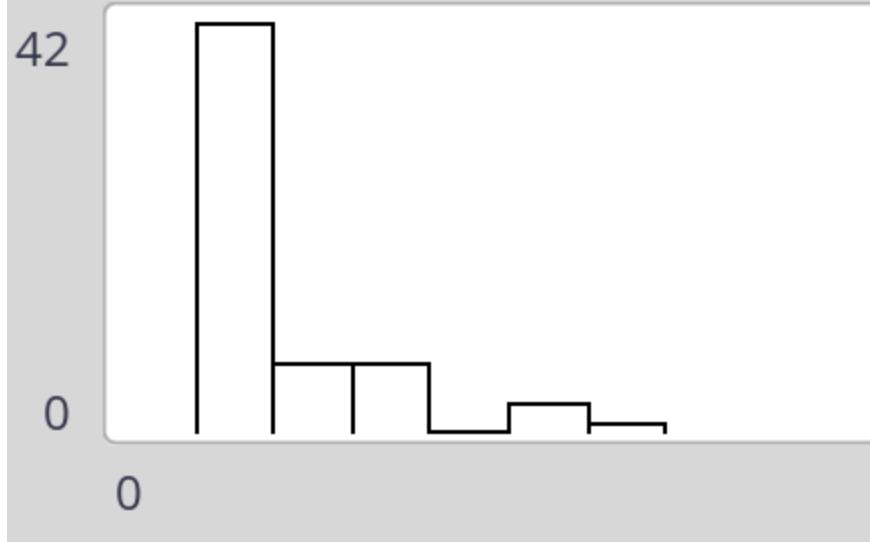
num-nodes

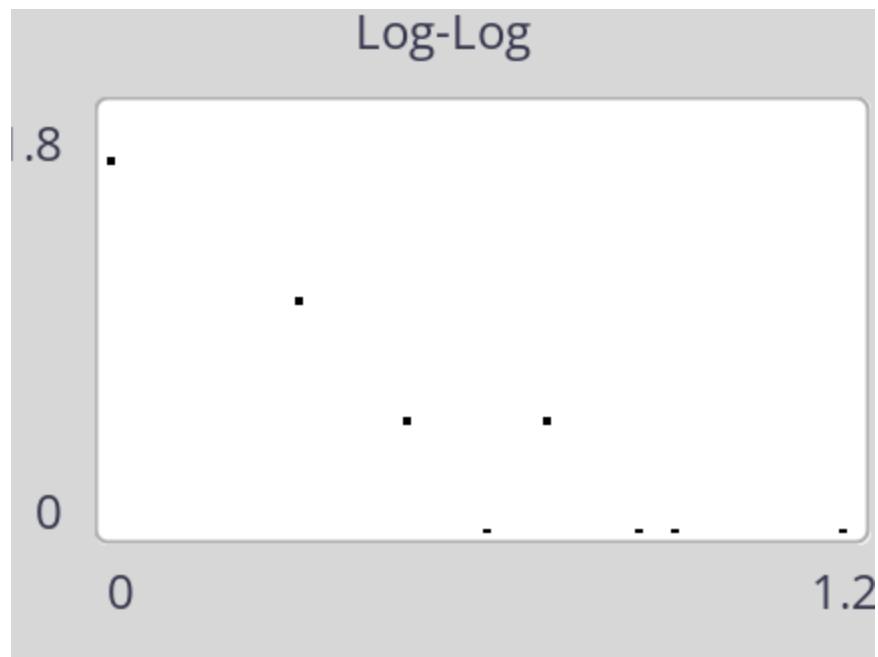


Log-Log



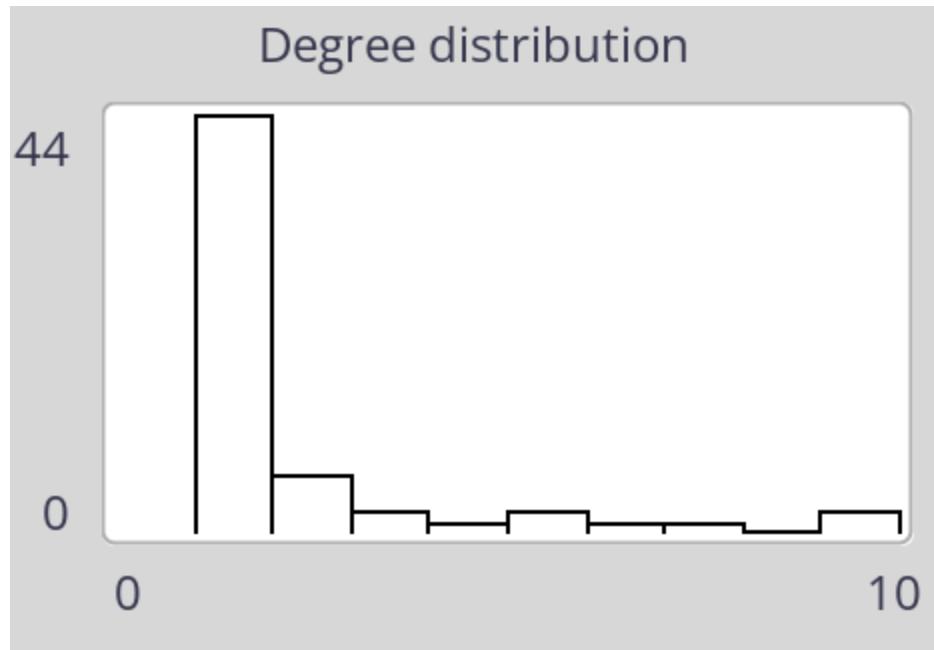
Degree distribution

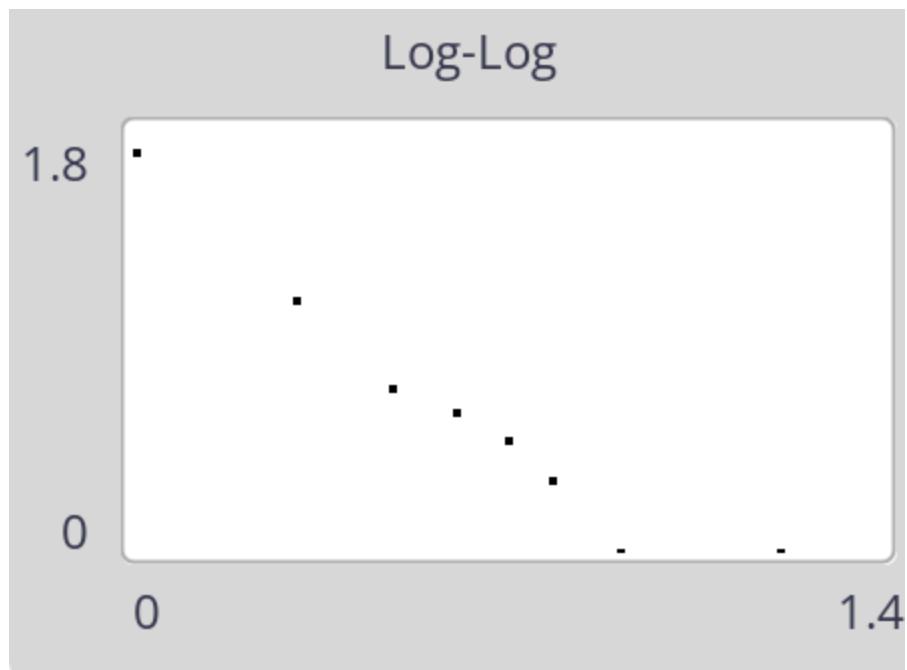




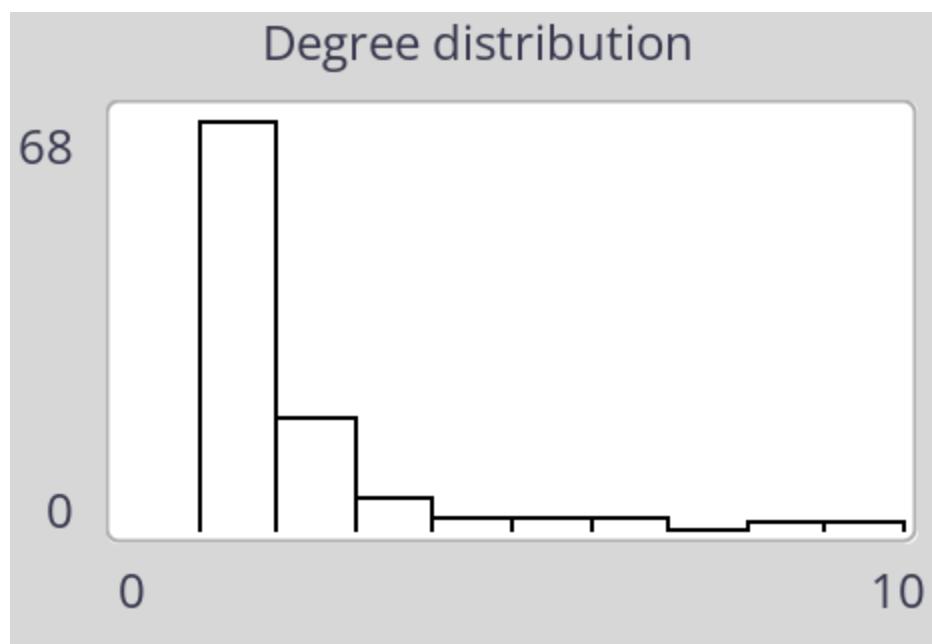
num-nodes

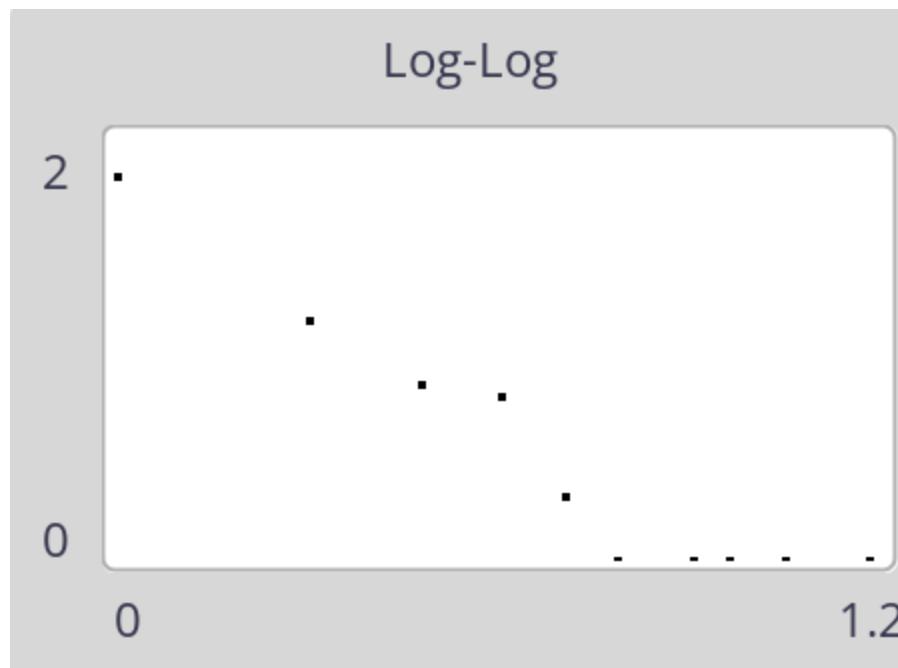
80





num-nodes



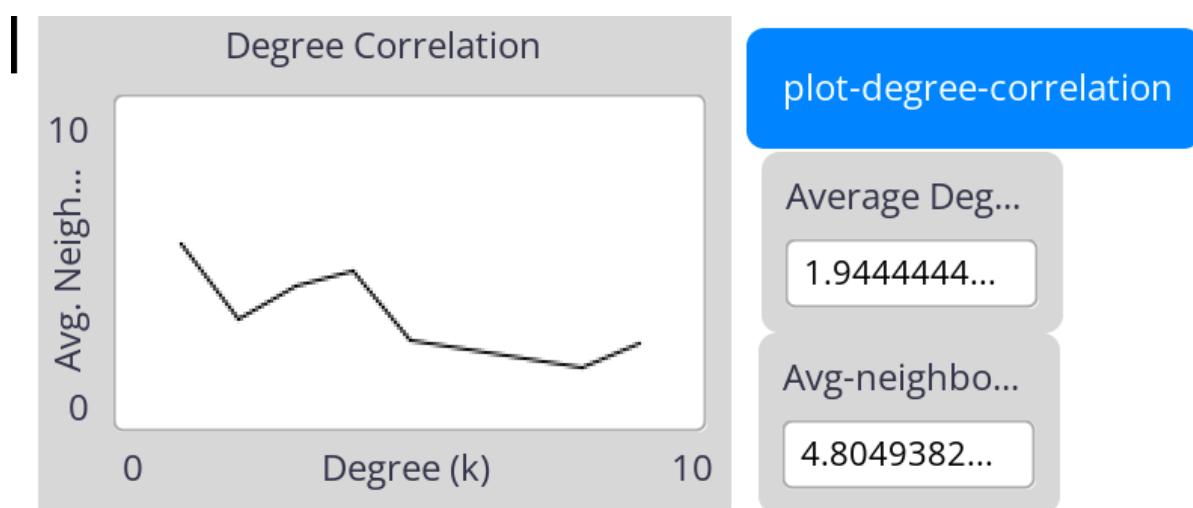


Indeed, as the number of nodes grows, we see that there is group of nodes to which almost all other nodes connected. Distribution also has bigger bump at the beginning.

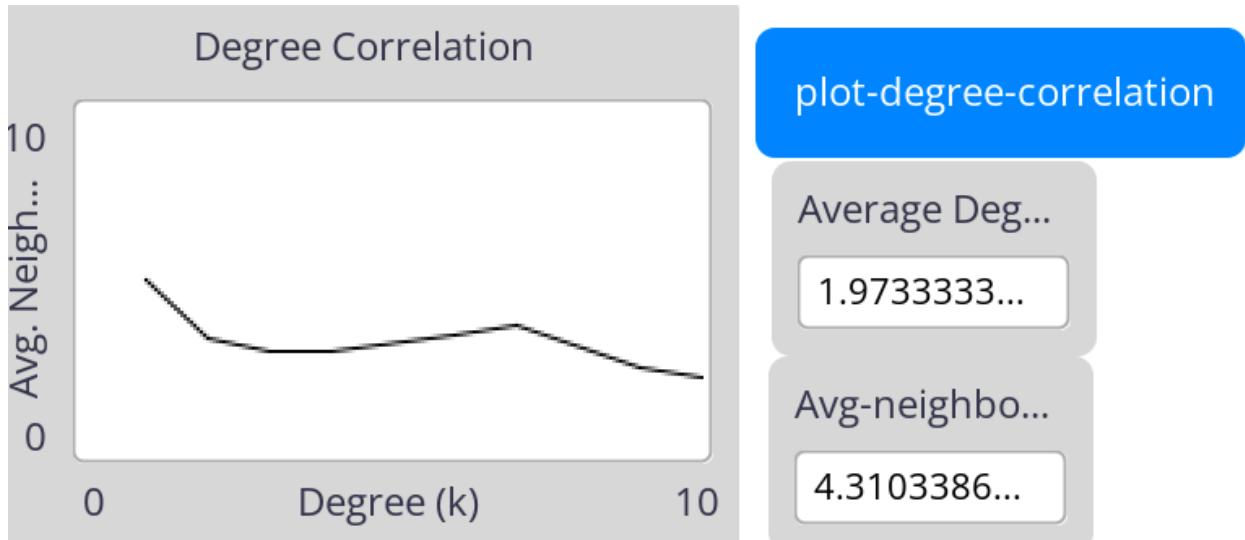
Friendship Paradox

Barabasi-Albert

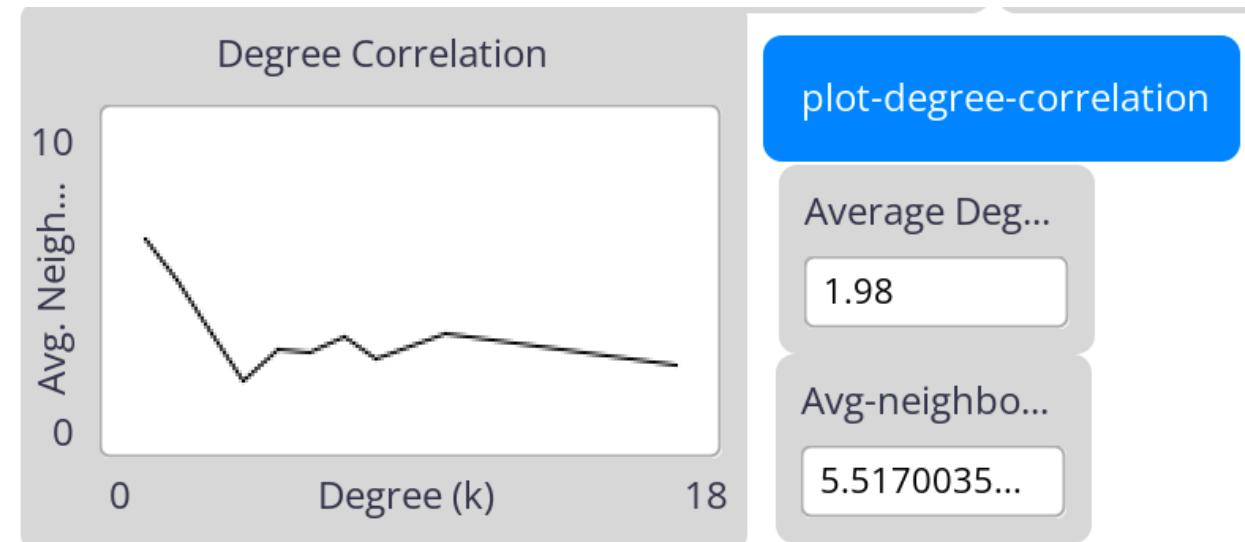
40 nodes



75 nodes

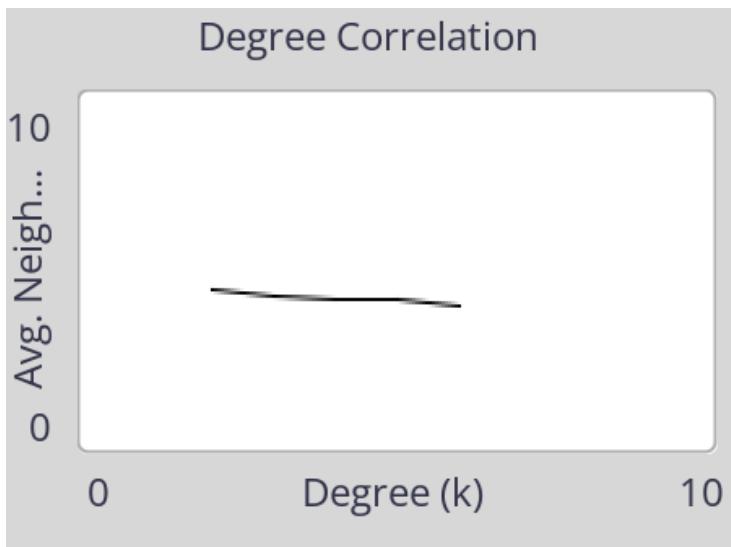


100 nodes



Small world

40 nodes



plot-degree-correlation

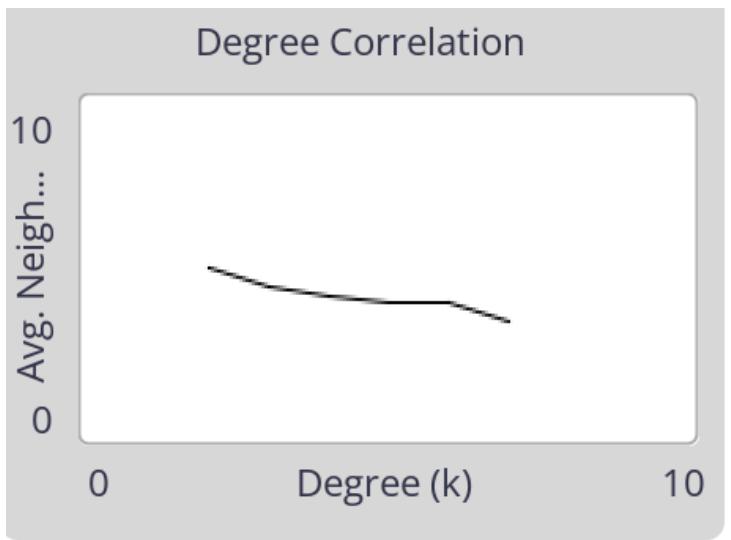
Average Deg...

4

Avg-neighbo...

4.2166666...

75 nodes



plot-degree-correlation

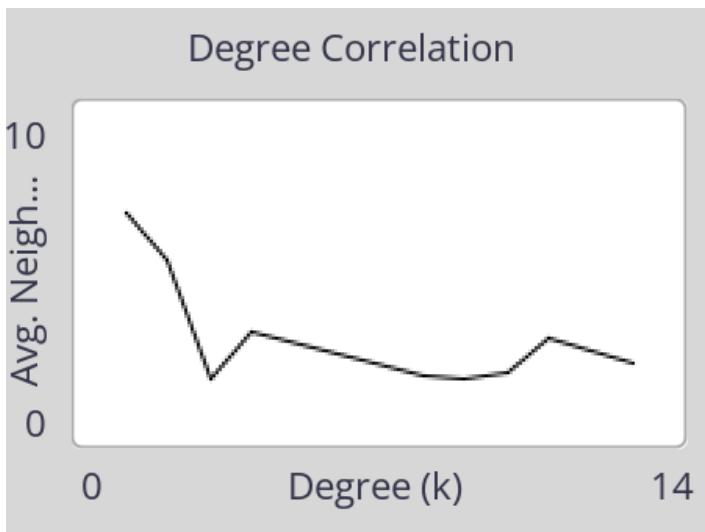
Average Deg...

4

Avg-neighbo...

4.2492698...

100 nodes



plot-degree-correlation

Average Deg...

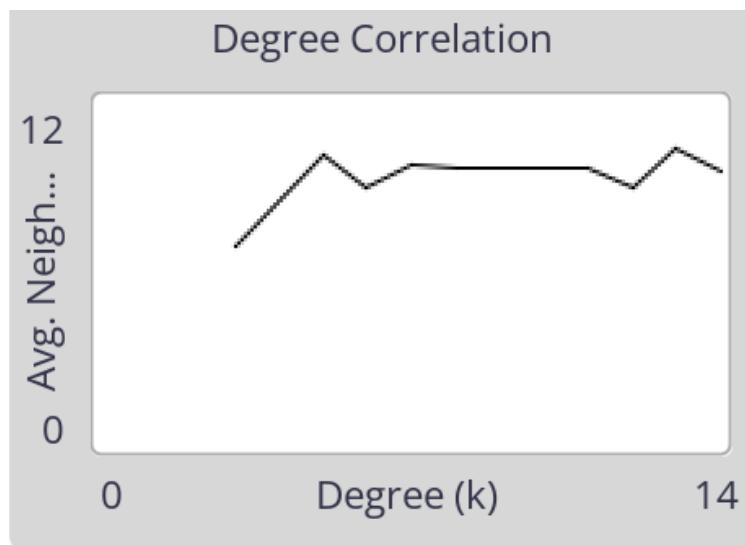
1.98

Avg-neighbo...

5.9696804...

Erdos-Renyi

40 nodes



plot-degree-correlation

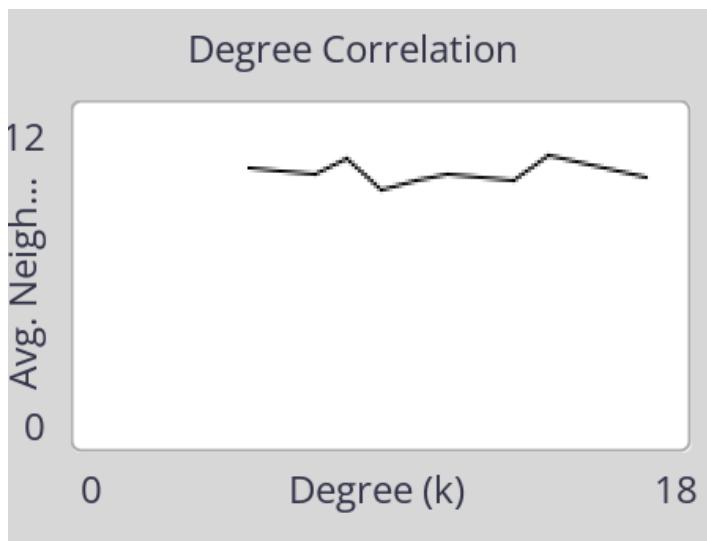
Average Deg...

8.95

Avg-neighbo...

9.6928565...

75 nodes



plot-degree-correlation

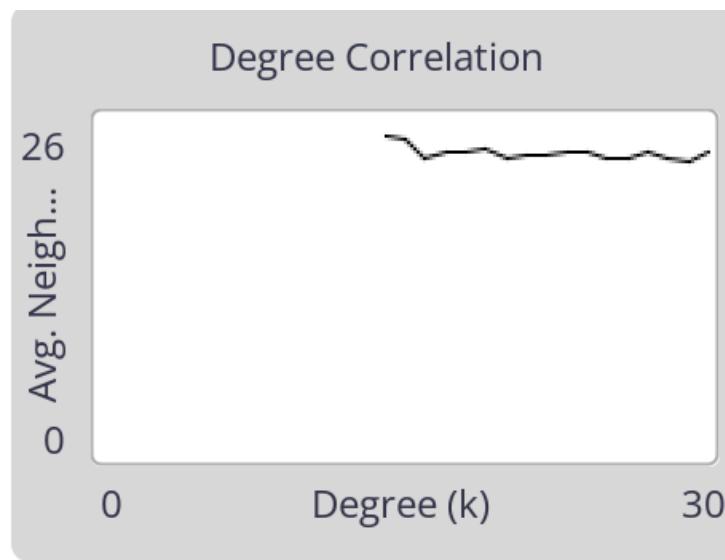
Average Deg...

8.8

Avg-neighbo...

9.6815096...

100 nodes



plot-degree-correlation

Average Deg...

22.66

Avg-neighbo...

23.311241...

I plotted avg neighbour degree, for each degree of the node itself. I also added monitors for average degree of a node and average degree of neighbours. I used code provided in the lab pdf.

[NetworksPractice.nlogo](#)

