

Social Network Analysis

Chen Avin

School of Electrical and Computer Engineering



In[398]:=

Unit 6

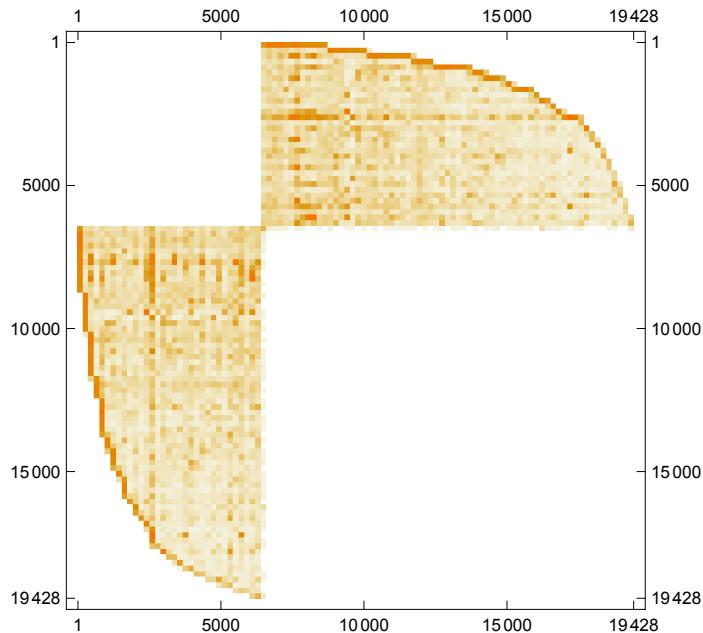
Core-Periphery

The Marvel Example

```
In[399]:= Marvel = ExampleData[{"NetworkGraph", "MarvelUniverseSocialGraph"}];
```

```
In[400]:= MatrixPlot[AdjacencyMatrix[Marvel]]
```

```
Out[400]=
```



```
In[401]:=
```

```
AM2 = MatrixPower[AdjacencyMatrix[Marvel], 2];
(* Take the part of the heros *)
Heros = AM2[[1 ;; 6486, 1 ;; 6486]];
(* remove values on the diagonal *)
SimpleHeros = Heros - Heros * IdentityMatrix[6486];
StrongEdges[Mat_, limit_] := Module[{rules},
  rules = ArrayRules[Mat];
  Select[rules, #[[2]] >= limit &] [[All, 1]]];
(* take strong edges *)
MarvelStrongEdges = StrongEdges[SimpleHeros, 50];
```

```
In[406]:=
```

```
Length[MarvelStrongEdges]
```

```
Out[406]=
```

```
1952
```

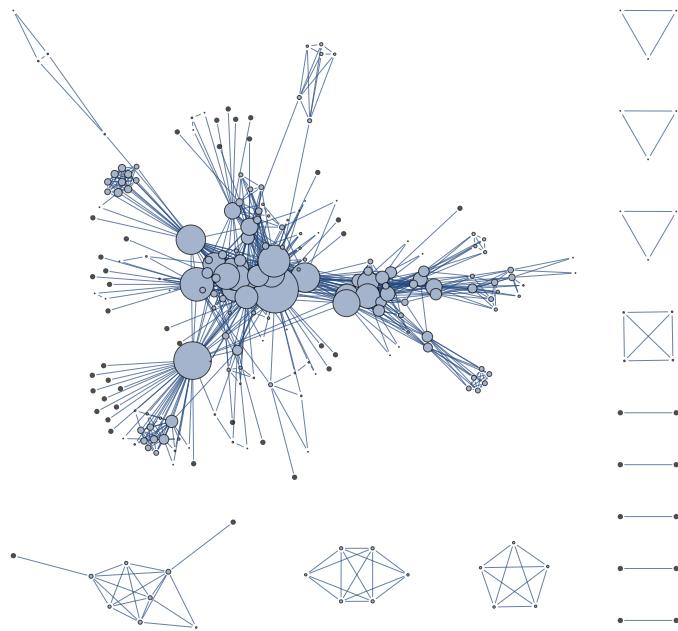
In[407]:=

```

HerosStrong = SimpleGraph[Graph[#, 1] & /@ MarvelStrongEdges]];
namelist = PropertyValue[{Marvel, #}, "Name"] & /@ VertexList[HerosStrong];
Graph[HerosStrong,
  VertexLabels → Table[VertexList[HerosStrong][i] → Placed[namelist[i], Tooltip],
  {i, 1, VertexCount[HerosStrong]}], VertexSize →
  Thread[VertexList[HerosStrong] → 9 * Rescale[VertexDegree[HerosStrong]]]]

```

Out[409]=



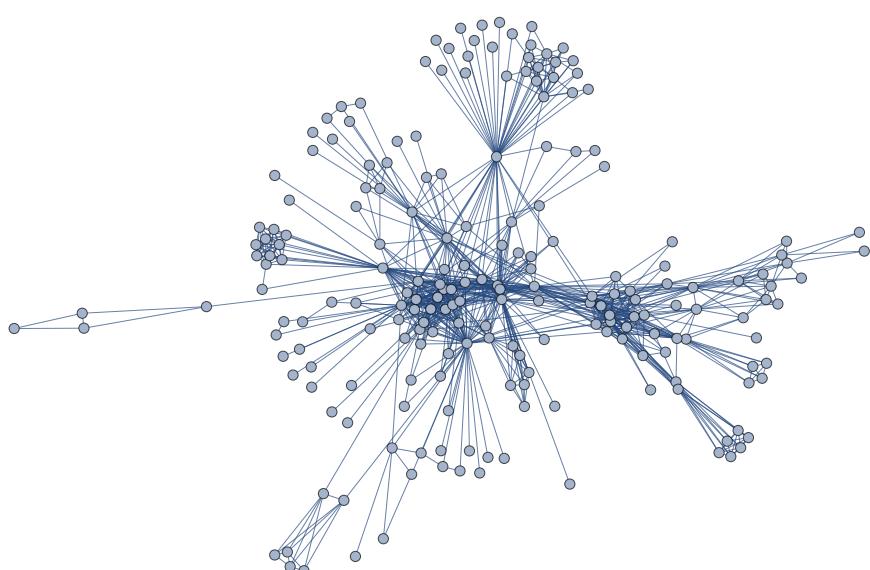
In[410]:=

```

MarvelExample =
Subgraph[HerosStrong, First@ConnectedComponents[HerosStrong], VertexLabels →
  Table[VertexList[HerosStrong][i] → Placed[namelist[i], Tooltip],
  {i, 1, VertexCount[HerosStrong]}]]

```

Out[410]=



```
In[411]:= VertexCount[MarvelExample]
Out[411]= 208
```

```
In[412]:= EdgeCount[MarvelExample]
Out[412]= 912
```

■ The Rich - Club

```
In[413]:= Rich[G_, k_] := VertexList[G][[Ordering[VertexDegree[G], k, Greater]]]
```

■ Symmetry Point Size

```
In[414]:= SymmetryPointSize[G_] := First@First@Position[Accumulate[VertexDegree[G]], Select[Accumulate[VertexDegree[G]], # ≥ Total[VertexDegree[G]] / 2 &, 1][[1]]]
```

■ Rename a graph

```
In[415]:= RenameGraphN[g_, sl_] :=
Module[{G = g, VrDg = sl, VrLs, VrLst, SVrLst, NewName, edges, NewEdges, ed},
VrLs = VertexList[G];
VrLst = Table[{VrDg[[i]], VrLs[[i]]}, {i, Length[VrDg]}];
SVrLst = Sort[VrLst, #1[[1]] > #2[[1]] &];
NewName = Table[SVrLst[[i]][[2]], {i, Length[VrDg]}];
edges = EdgeList[G];
If[DirectedGraphQ[G],
NewEdges = Table[Position[NewName, edges[[i]][[1]]][[1]] ↔
Position[NewName, edges[[i]][[2]]][[1]][[1]], {i, Length[edges]}],
NewEdges = Table[Position[NewName, edges[[i]][[1]]][[1]][[1]] ↔
Position[NewName, edges[[i]][[2]]][[1]][[1]], {i, Length[edges]}]];
Graph[Range[Length[VrLst]], NewEdges]]
```

■ Take the core at the symmetry point

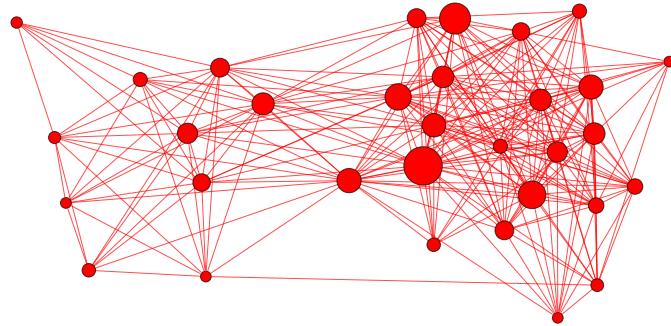
```
In[416]:= namelistcore = PropertyValue[{Marvel, #}, "Name"] & /@ VertexList[MarvelExample];
```

```
In[417]:= DegreeOrderGraph = RenameGraphN[MarvelExample, VertexDegree[MarvelExample]];
```

In[418]:=

```
Core = Subgraph[MarvelExample,
  Rich[MarvelExample, SymmetryPointSize[DegreeOrderGraph]], VertexLabels →
  Table[VertexList[MarvelExample][[i]] → Placed[namelistcore[[i]], Tooltip],
  {i, 1, VertexCount[MarvelExample]}], VertexSize →
  Thread[VertexList[MarvelExample] → Rescale[VertexDegree[MarvelExample]]],
  VertexStyle → Red, EdgeStyle → Red]
```

Out[418]=



In[419]:=



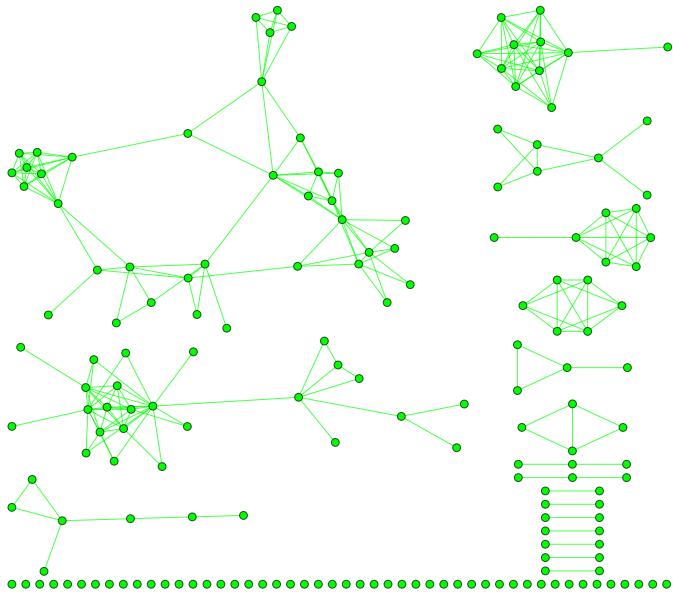
Out[419]=



In[420]:=

```
Periphery = Subgraph[MarvelExample,
  Complement[VertexList[MarvelExample], VertexList[Core]],
  VertexStyle → Green, EdgeStyle → Green]
```

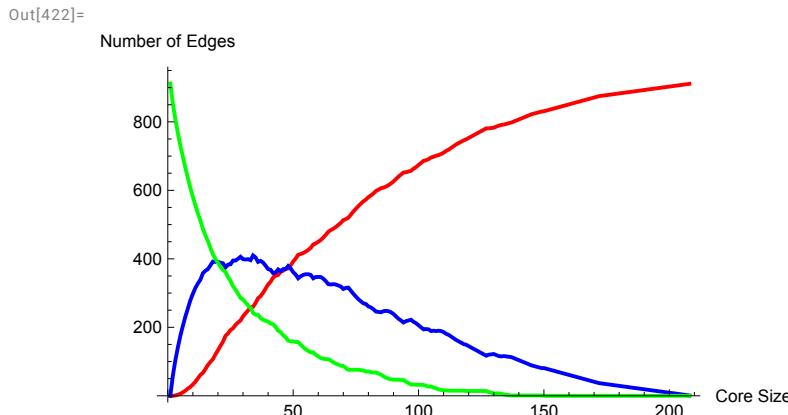
Out[420]=



■ The Symmetry Point Graph

```
In[421]:= SymmetryPoint[G_] := Module[{s, EE = {}, EP = {},
PE = {}, PP = {}, ee = 0, ep = 0, pe = 0, pp = 0, tc, bc, tr, br},
s = Length[VertexList[G]];
ee = AdjacencyMatrix[G][[1]][1];
pe = Total[Total[Take[AdjacencyMatrix[G], {2, s}, {1, 1}]]];
ep = Total[Total[Take[AdjacencyMatrix[G], {1, 1}, {2, s}]]];
pp = Total[Total[Take[AdjacencyMatrix[G], {2, s}, {2, s}]]];
EE = {0, ee / 2};
EP = {0, ep / 2};
PE = {0, pe / 2};
PP = {EdgeCount[G], pp / 2};
For[i = 1, i < s - 1, i++,
tc = Total[Total[Take[AdjacencyMatrix[G], {1, i - 1}, {i, i}]]];
bc = Total[Total[Take[AdjacencyMatrix[G], {i + 1, s}, {i, i}]]];
tr = Total[Total[Take[AdjacencyMatrix[G], {i, i}, {1, i - 1}]]];
br = Total[Total[Take[AdjacencyMatrix[G], {i, i}, {i + 1, s}]]];
ee = ee + tc + tr + AdjacencyMatrix[G][[i]][i];
ep = ep + br - tc;
pe = pe + bc - tr;
pp = pp - bc - br - AdjacencyMatrix[G][[i]][i];
EE = Join[EE, {ee / 2}];
EP = Join[EP, {ep / 2}];
PE = Join[PE, {pe / 2}];
PP = Join[PP, {pp / 2}];];
{EE, EP + PE, PP}]
```

```
In[422]:= ListLinePlot[SymmetryPoint[DegreeOrderGraph],
PlotLegends -> {"I(ε,ε)", "I(ρ,ε)", "I(ρ,ρ)"}, PlotRange -> All,
AxesLabel -> {"Core Size", "Number of Edges"},
PlotStyle -> {{Red, Thick}, {Blue, Thick}, {Green, Thick}}]
```



```
In[423]:= EdgeCount[Core]
```

```
Out[423]=
```

```
In[424]:= EdgeCount[Periphery]
```

```
Out[424]= 257
```

```
In[425]:= VertexCount[Core]
```

```
Out[425]= 32
```

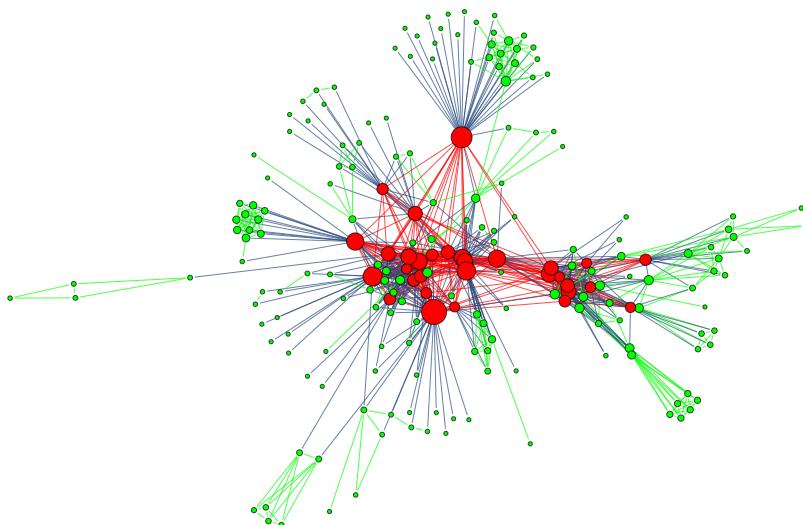
```
In[426]:= VertexCount[Periphery]
```

```
Out[426]= 176
```

■ Putting it Together

```
In[427]:= HighlightGraph[MarvelExample, {Style[VertexList[Core], Red],
Style[VertexList[Periphery], Green], Style[EdgeList[Core], Red],
Style[EdgeList[Periphery], Green]}, VertexSize → Thread[
VertexList[MarvelExample] → 1 + 5 * Rescale[VertexDegree[MarvelExample]]]]
```

```
Out[427]=
```



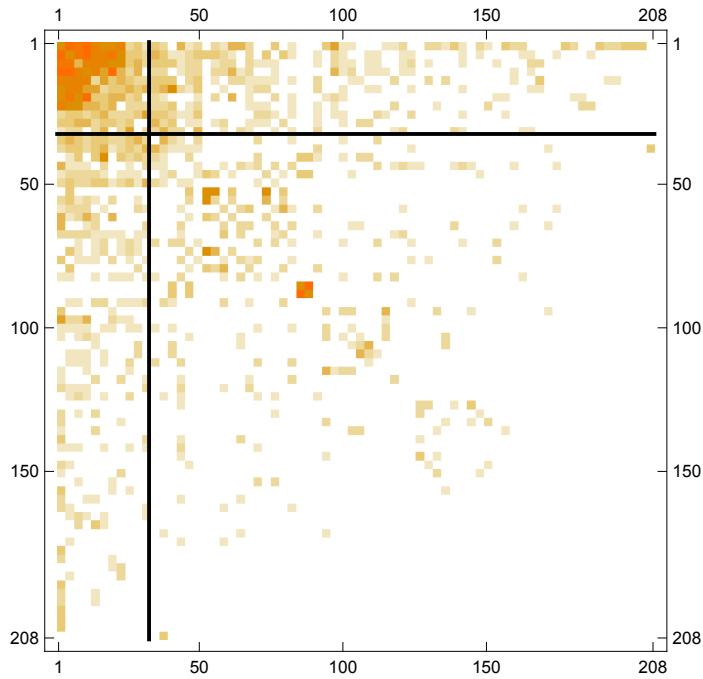
```
In[428]:=
```

```
CoreRatio = SymmetryPointSize[DegreeOrderGraph] / VertexCount[DegreeOrderGraph];
```

In[429]:=

```
Show[MatrixPlot[AdjacencyMatrix[DegreeOrderGraph]],  
Graphics[{{Thick, Line[{{CoreRatio * 70, 0}, {CoreRatio * 70, 70}}]},  
, {Thick, Line[{{0, 70 - CoreRatio * 70}, {70, 70 - CoreRatio * 70}}]}]]]
```

Out[429]=



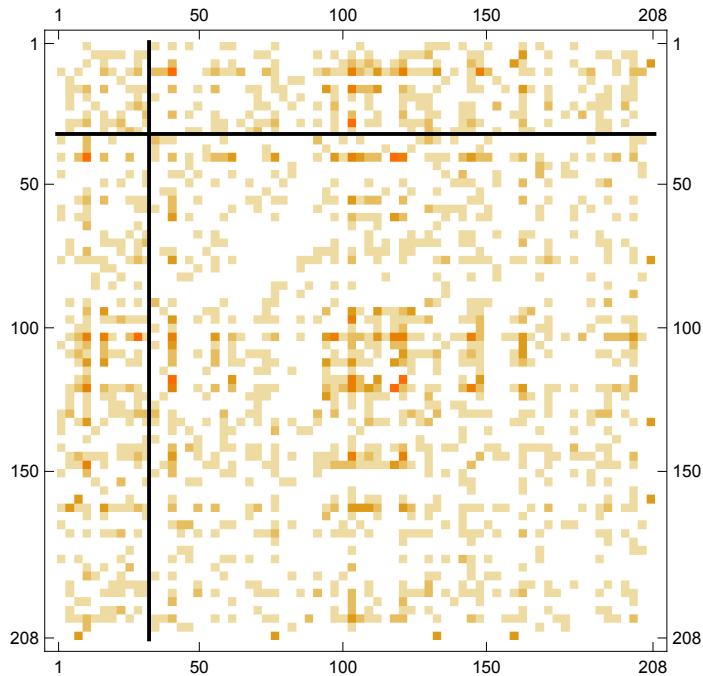
In[430]:=

```
RandomOrderGraph = RenameGraphN[MarvelExample,  
RandomSample[VertexList[MarvelExample], VertexCount[MarvelExample]]];
```

In[431]:=

```
Show[MatrixPlot[AdjacencyMatrix[RandomOrderGraph]],
Graphics[{{Thick, Line[{{CoreRatio * 70, 0}, {CoreRatio * 70, 70}}]}, {Thick, Line[{{0, 70 - CoreRatio * 70}, {70, 70 - CoreRatio * 70}}]}]]]
```

Out[431]=



In[432]:=

```
Sqrt[EdgeCount[MarvelExample]] // N
```

Out[432]=

```
30.1993
```

In[433]:=

```
CoreRatio // N
```

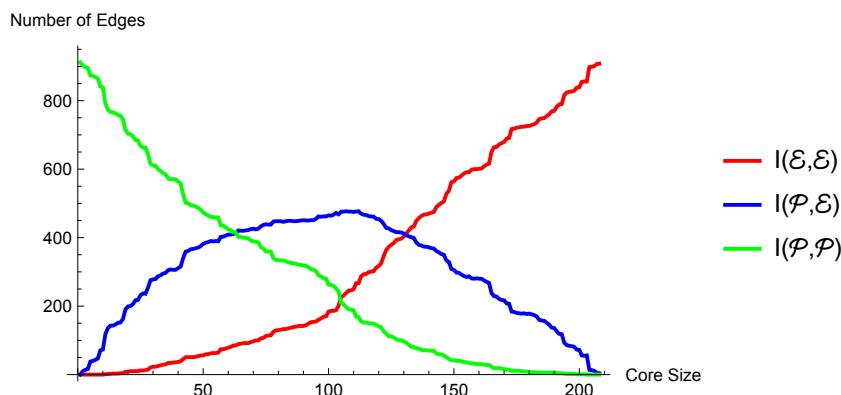
Out[433]=

```
0.153846
```

In[434]:=

```
ListLinePlot[SymmetryPoint[RandomOrderGraph],
PlotLegends → {"I(ε,ε)", "I(ρ,ε)", "I(ρ,ρ)"}, PlotRange → All,
AxesLabel → {"Core Size", "Number of Edges"},
PlotStyle → {{Red, Thick}, {Blue, Thick}, {Green, Thick}}]
```

Out[434]=

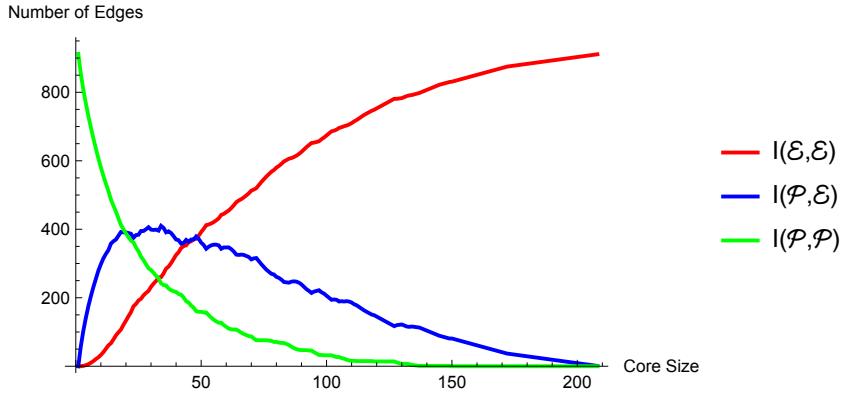


In[435]:=

In[436]:=

```
ListLinePlot[SymmetryPoint[DegreeOrderGraph],
PlotLegends → {"I(ε,ε)", "I(ρ,ε)", "I(ρ,ρ)"}, PlotRange → All,
AxesLabel → {"Core Size", "Number of Edges"},
PlotStyle → {{Red, Thick}, {Blue, Thick}, {Green, Thick}}]
```

Out[436]=



In[437]:=

```
GraphDensity[Core] // N
```

Out[437]=

```
0.522177
```

In[438]:=

```
GraphDensity[Periphery] // N
```

Out[438]=

```
0.0166883
```

In[439]:=

```
Density[G_] := Log[EdgeCount[G] + VertexCount[G]] / Log[VertexCount[G]] // N
```

In[440]:=

```
Density[Core]
```

Out[440]=

```
1.63698
```

In[441]:=

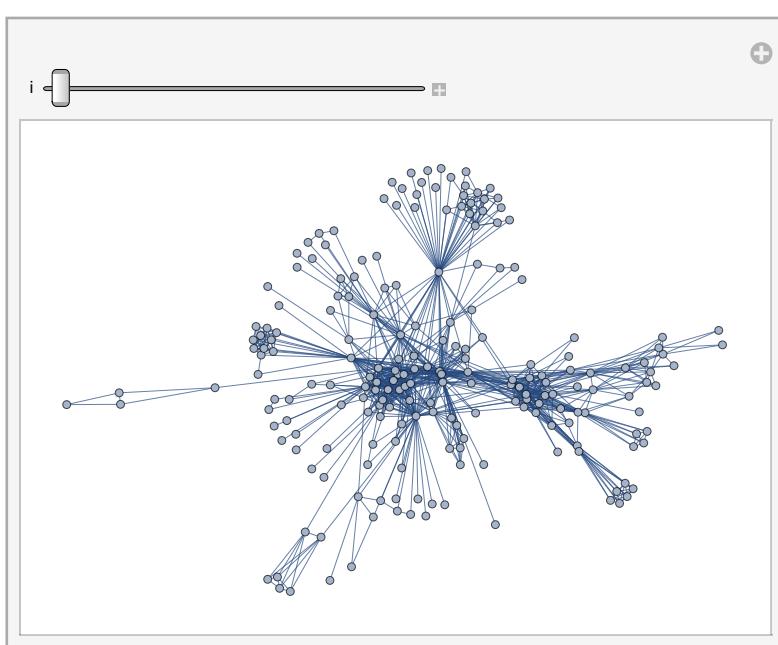
```
Density[Periphery]
```

Out[441]=

```
1.17411
```

```
In[442]:= KCoreComponents[MarvelExample, 3]
Out[442]= {1383, 6306, 5467, 4726, 2397, 4395, 1127, 3250, 340, 854, 1289, 2557, 2603, 2650,
3495, 4366, 3363, 3329, 403, 2040, 5046, 4289, 133, 3805, 6300, 5525, 3055,
5002, 5306, 2548, 3950, 859, 5716, 500, 991, 4197, 5165, 6216, 5583, 2582, 6141,
4661, 3360, 5146, 1299, 3691, 649, 5540, 3359, 5762, 4650, 6313, 3302, 1602,
5516, 3530, 1582, 2664, 154, 2399, 505, 1265, 2031, 5736, 5121, 1347, 6148,
2354, 2184, 6066, 4898, 3594, 2449, 2971, 4454, 3757, 3986, 522, 875, 4441,
867, 3601, 4240, 3231, 2218, 4155, 4704, 5732, 5970, 6160, 4088, 5354, 3135,
2959, 3136, 2203, 4087, 6166, 337, 2430, 3035, 4759, 5613, 3956, 1535, 491, 533,
545, 3712, 4978, 5776, 1892, 5388, 5978, 2728, 5874, 3889, 5310, 3233, 1818,
2731, 4045, 5099, 3245, 6072, 323, 2506, 2422, 6084, 3994, 5294, 3949, 3066}
```

```
In[443]:= Manipulate[
Subgraph[MarvelExample, KCoreComponents[MarvelExample, i]], {i, 1, 20, 1}]
Out[443]=
```



In[444]:=

```
Select[ArrayRules[AdjacencyMatrix[DegreeOrderGraph]], #[[2]] == 1 &]
```

Out[444]=

```
{\{1, 2\} \rightarrow 1, \{1, 3\} \rightarrow 1, \{1, 4\} \rightarrow 1, \{1, 5\} \rightarrow 1, \{1, 6\} \rightarrow 1, \{1, 7\} \rightarrow 1, \{1, 8\} \rightarrow 1, \{1, 9\} \rightarrow 1,
\{1, 10\} \rightarrow 1, \{1, 11\} \rightarrow 1, \{1, 12\} \rightarrow 1, \{1, 13\} \rightarrow 1, \{1, 14\} \rightarrow 1, \{1, 15\} \rightarrow 1, \{1, 16\} \rightarrow 1, \{1, 17\} \rightarrow 1,
\{1, 18\} \rightarrow 1, \{1, 19\} \rightarrow 1, \{1, 20\} \rightarrow 1, \{1, 21\} \rightarrow 1, \{1, 22\} \rightarrow 1, \{1, 23\} \rightarrow 1, \{1, 24\} \rightarrow 1, \{1, 26\} \rightarrow 1,
\{1, 29\} \rightarrow 1, \{1, 30\} \rightarrow 1, \{1, 34\} \rightarrow 1, \{1, 35\} \rightarrow 1, \{1, 39\} \rightarrow 1, \{1, 44\} \rightarrow 1, \{1, 48\} \rightarrow 1,
\{1, 49\} \rightarrow 1, \{1, 57\} \rightarrow 1, \{1, 62\} \rightarrow 1, \{1, 63\} \rightarrow 1, \{1, 64\} \rightarrow 1, \{1, 69\} \rightarrow 1, \{1, 76\} \rightarrow 1,
\{1, 95\} \rightarrow 1, \{1, 96\} \rightarrow 1, \{1, 97\} \rightarrow 1, \{1, 98\} \rightarrow 1, \{1, 99\} \rightarrow 1, \{1, 100\} \rightarrow 1, \{1, 104\} \rightarrow 1,
\{1, 117\} \rightarrow 1, \{1, 134\} \rightarrow 1, \{1, 135\} \rightarrow 1, \{1, 136\} \rightarrow 1, \{1, 157\} \rightarrow 1, \{1, 158\} \rightarrow 1, \{1, 159\} \rightarrow 1,
\{1, 160\} \rightarrow 1, \{1, 190\} \rightarrow 1, \{1, 191\} \rightarrow 1, \{1, 192\} \rightarrow 1, ... 1712 ..., \{162, 43\} \rightarrow 1, \{163, 2\} \rightarrow 1,
\{163, 58\} \rightarrow 1, \{164, 2\} \rightarrow 1, \{164, 33\} \rightarrow 1, \{165, 6\} \rightarrow 1, \{165, 83\} \rightarrow 1, \{166, 7\} \rightarrow 1, \{166, 11\} \rightarrow 1,
\{167, 4\} \rightarrow 1, \{167, 7\} \rightarrow 1, \{168, 16\} \rightarrow 1, \{168, 31\} \rightarrow 1, \{169, 13\} \rightarrow 1, \{169, 17\} \rightarrow 1, \{170, 8\} \rightarrow 1,
\{170, 13\} \rightarrow 1, \{171, 8\} \rightarrow 1, \{171, 13\} \rightarrow 1, \{172, 96\} \rightarrow 1, \{173, 39\} \rightarrow 1, \{174, 64\} \rightarrow 1,
\{175, 44\} \rightarrow 1, \{176, 5\} \rightarrow 1, \{177, 60\} \rightarrow 1, \{178, 3\} \rightarrow 1, \{179, 3\} \rightarrow 1, \{180, 3\} \rightarrow 1, \{181, 3\} \rightarrow 1,
\{182, 3\} \rightarrow 1, \{183, 21\} \rightarrow 1, \{184, 21\} \rightarrow 1, \{185, 24\} \rightarrow 1, \{186, 24\} \rightarrow 1, \{187, 24\} \rightarrow 1,
\{188, 24\} \rightarrow 1, \{189, 4\} \rightarrow 1, \{190, 1\} \rightarrow 1, \{191, 1\} \rightarrow 1, \{192, 1\} \rightarrow 1, \{193, 1\} \rightarrow 1, \{194, 1\} \rightarrow 1,
\{195, 15\} \rightarrow 1, \{196, 15\} \rightarrow 1, \{197, 2\} \rightarrow 1, \{198, 2\} \rightarrow 1, \{199, 2\} \rightarrow 1, \{200, 2\} \rightarrow 1, \{201, 2\} \rightarrow 1,
\{202, 2\} \rightarrow 1, \{203, 2\} \rightarrow 1, \{204, 2\} \rightarrow 1, \{205, 2\} \rightarrow 1, \{206, 2\} \rightarrow 1, \{207, 2\} \rightarrow 1, \{208, 38\} \rightarrow 1}
```

Size in memory: 306.7 kB

[+ Show more](#)[Show all](#)[Iconize ▾](#)[Store full expression in notebook](#)

Assignment 6 - Core-Periphery

- Compute the Core-Periphery partition “Network of Thrones”
 - with and without weights (nodes degree is the sum of weights)
- You need to do:
 - 1. Generate the power shift diagram (according to degree (weighted and unweighted))
 - 2. Generate the Core graph. What is the size of the core? Who is in the core?
 - 3. Generate the Periphery graph. What is the size of the core?
 - 4. Generate the matrix and show the symmetry point
 - Compare the results to a shift diagram of random order of nodes and to a shift diagram with random configuration model of the same degree sequence
- Bonus: Do the same but with your favorite centrality measure
 - How different is the core?

Slide

In[445]:=

```
Core = Subgraph[MarvelExample, Rich[MarvelExample, 10], VertexLabels →
  Table[VertexList[MarvelExample][[i]] → Placed[namelistcore[[i]], Tooltip],
  {i, 1, VertexCount[MarvelExample]}], VertexSize →
  Thread[VertexList[MarvelExample] → Rescale[VertexDegree[MarvelExample]]],
  VertexStyle → Red, EdgeStyle → Red];
Periphery = Subgraph[MarvelExample, Complement[VertexList[MarvelExample],
  VertexList[Core]], VertexStyle → Green, EdgeStyle → Green];
```

In[447]:=

```
HighlightGraph[MarvelExample, {Style[VertexList[Core], Red],
  Style[VertexList[Periphery], Green], Style[EdgeList[Core], Red],
  Style[EdgeList[Periphery], Green]}, VertexSize → Thread[
  VertexList[MarvelExample] → 1 + 5 * Rescale[VertexDegree[MarvelExample]]]]
```

Out[447]=

