Case study on centrality

Social Networks Analysis and Graph Algorithms

Prof. Carlos Castillo — https://chato.cl/teach

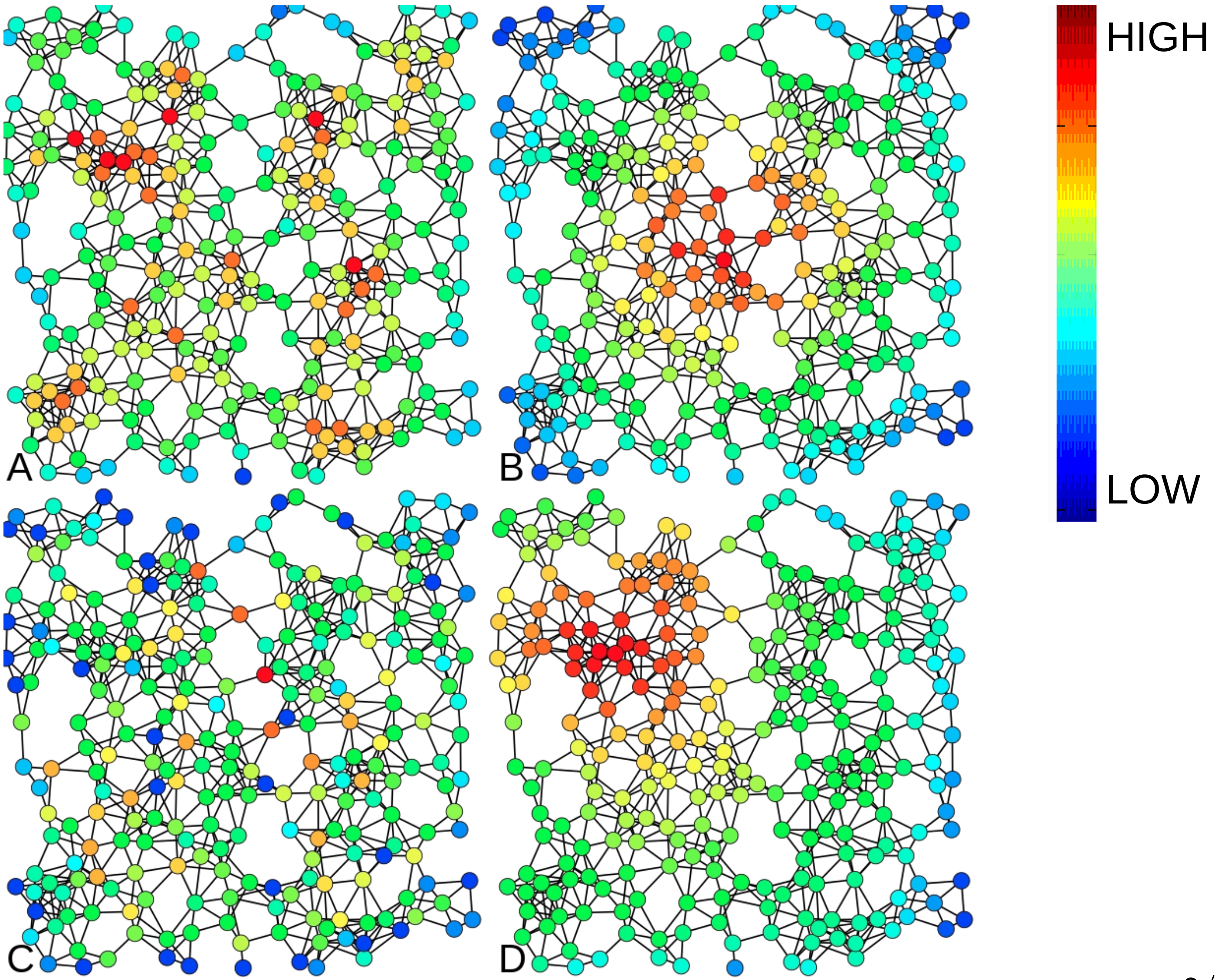
Universitat Pompeu Fabra Barcelona

A: Degree

B: Closeness

C: Betweenness

D. PageRank



Case study:
Noble families in Florence
in the 15th century

Florentine families

- Noble families in Florence around 1430
- Power struggle between two factions led by the Medici and the Strozzi
- The relatively newcomer Medici became, for a while, the wealthiest family in Europe ... they had their own bank!
- Dataset collected by John Padgett from historical documents

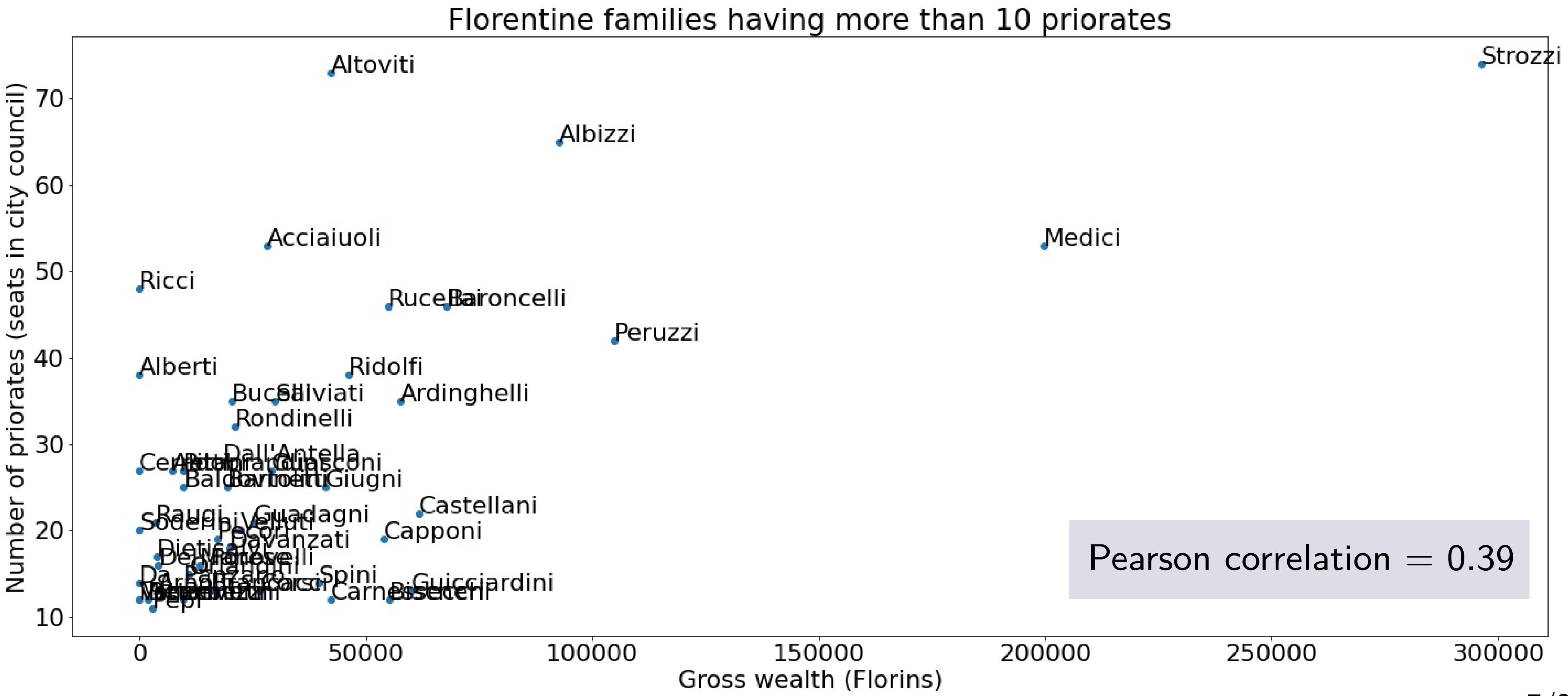
Wealth and political power

- The dataset contains 116 families
- **Gross wealth** in Florins (1 florin $\sim$ 3.5g of gold)
  - These are all approximations assuming *florins* and *ducats* have similar value:
    - Leonardo da Vinci was paid $\sim$100 florin per year ($\sim$1 painting), until he worked with the king of France, who paid $\sim$400 florin per year
    - Michelangelo Buonarroti got paid $\sim$200-450 florins per sculpture
    - A palace would cost a few thousand florins
- **Priorates** is the cumulative number of seats in the city council along mulriple years

Wealth and political power (cont.)

```
plt.scatter(
  families[families.Npriors > MIN_PRIORS].Gwealth,
  families[families.Npriors > MIN_PRIORS].Npriors)


families.Gwealth.corr(
  families.Npriors, method='pearson'))
```

Wealth and political power (cont.)

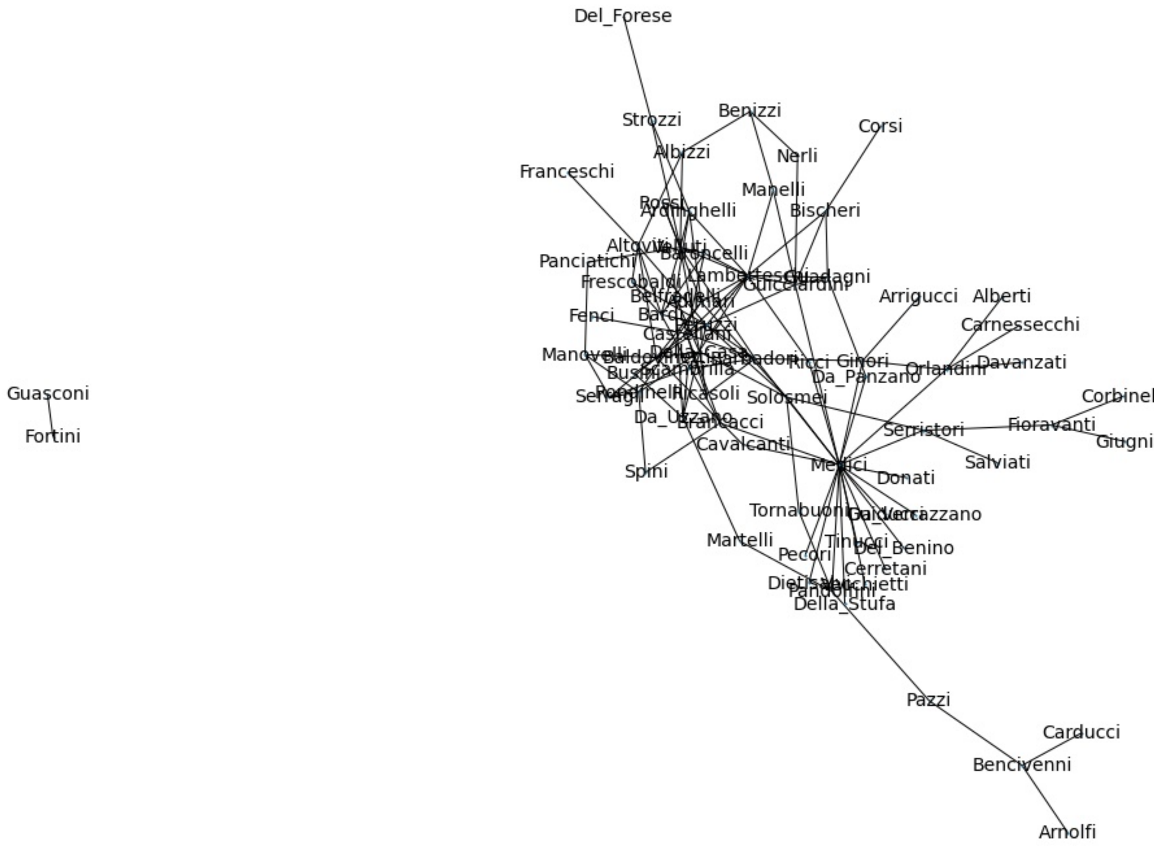


Credit graph

[Image source]

Credit graph

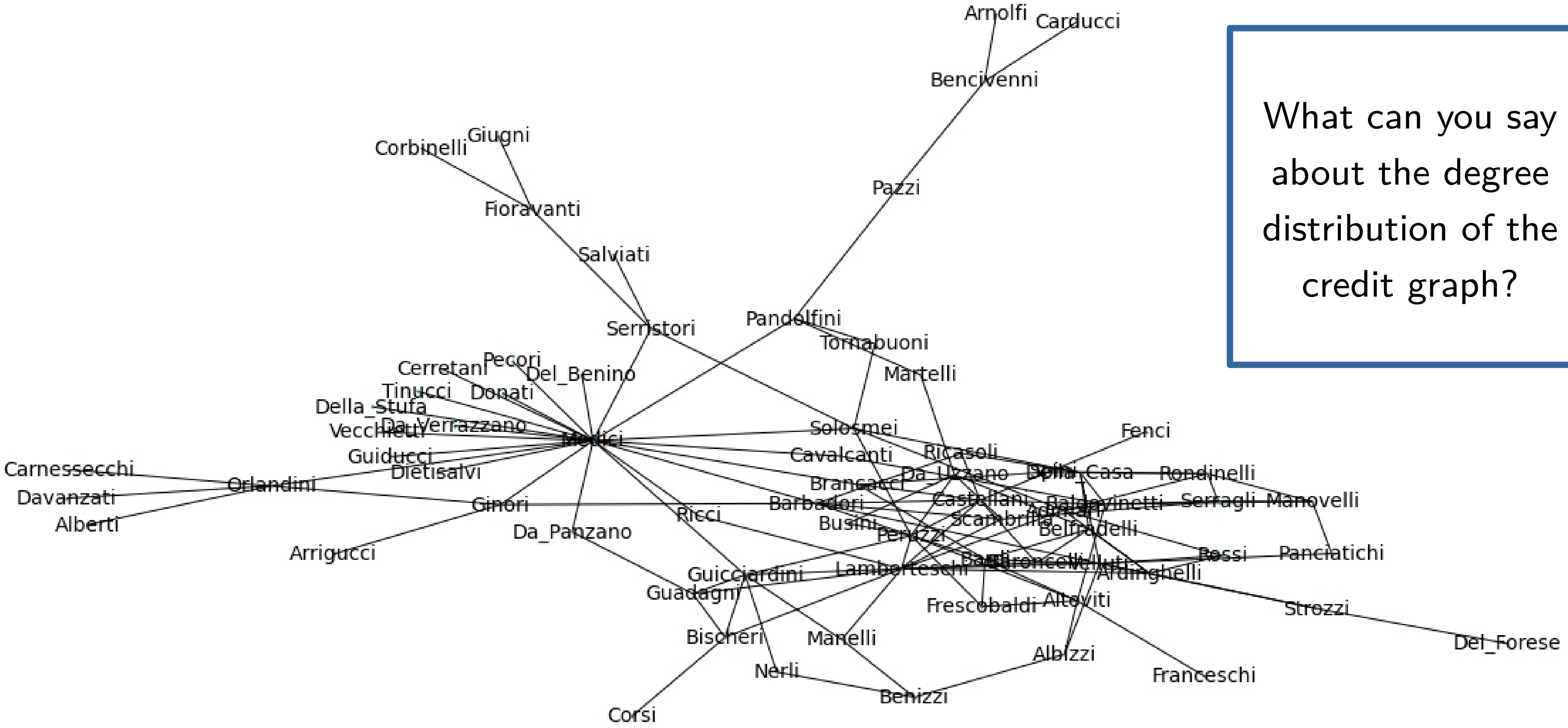
- 72 nodes (families)
- 125 edges (loans)
- Loan given by one family to a member of the other
- Undirected in this dataset



Credit graph in NetworkX

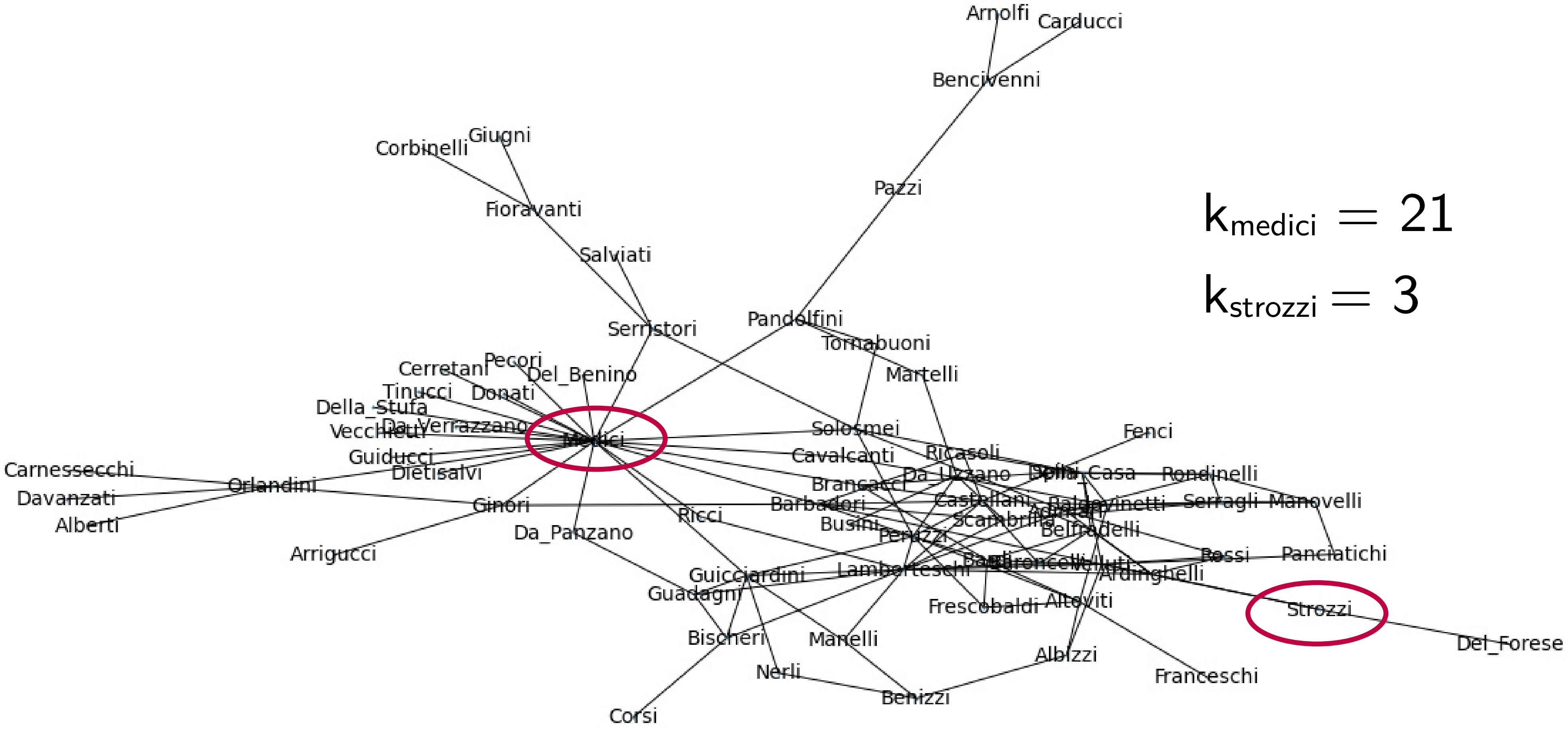
```
credits_list = pd.read_csv(INPUT_CREDIT,
   usecols=['FamilyA', 'FamilyB'])

credits = nx.from_pandas_edgelist(credits_list,
   "FamilyA", "FamilyB")

…

credits_components = sorted(
   nx.connected_components(credits), key=len, reverse=True)

credits_gcc = credits.subgraph(credits_components[0])
```

Credit - giant connected component (70 nodes, 97%)



What can you say about the degree distribution of the credit graph?

Credit - giant connected component (70 nodes, 97%)



$k_{medici} = 21$

$k_{strozzi} = 3$

Closeness computation

```
c_closeness = pd.DataFrame.from_dict(
   nx.closeness_centrality(credits_gcc),
   orient='index', columns=['c_closeness'])

families = families.join(c_closeness, how='inner')
```

Closeness, betweenness, eigencentrality

Closeness

- Peruzzi 0.39
- Medici 0.48
- Strozzi 0.28

Betweenness

- Peruzzi 0.11
- Medici 0.53
- Strozzi 0.03

Eigencentrality

- Peruzzi 0.30
- Medici 0.31
- Strozzi 0.07

What can you say about the correlations of this with wealth/power?

Computing and visualizing correlations

```
corr = families.corr()

corr
    .style.background_gradient(cmap='Reds')
    .format(precision=2)
```

Correlations

	Gwealth	Npriors	c_degree	c_closeness	c_betweenness	c_eigencentrality
Gwealth	1.00	0.39	0.42	0.21	0.40	0.34
Npriors	0.39	1.00	0.27	0.04	0.20	0.19
c_degree	0.42	0.27	1.00	0.67	0.84	0.88
c_closeness	0.21	0.04	0.67	1.00	0.59	0.79
c_betweenness	0.40	0.20	0.84	0.59	1.00	0.59
c_eigencentrality	0.34	0.19	0.88	0.79	0.59	1.00

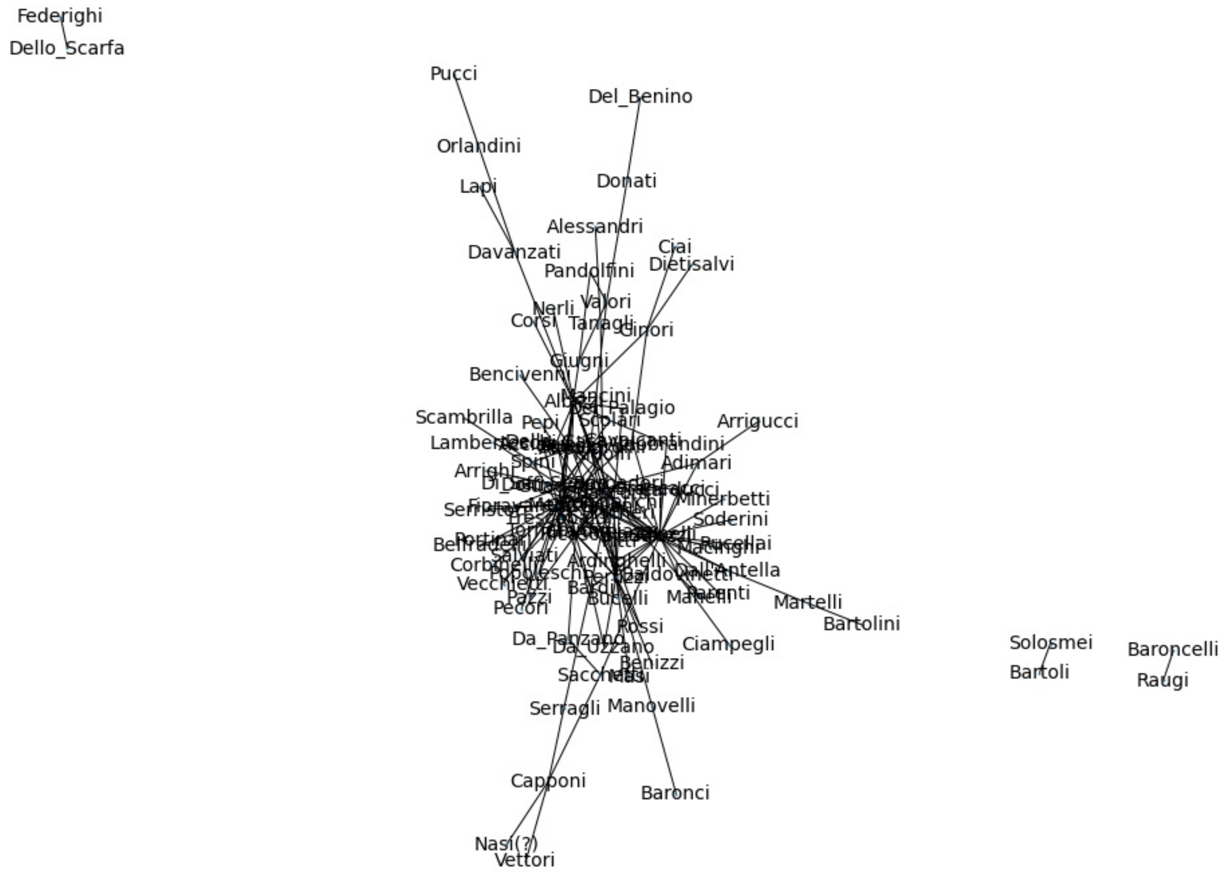
Do you see the block structure in this matrix? What does it mean?

Marriages graph

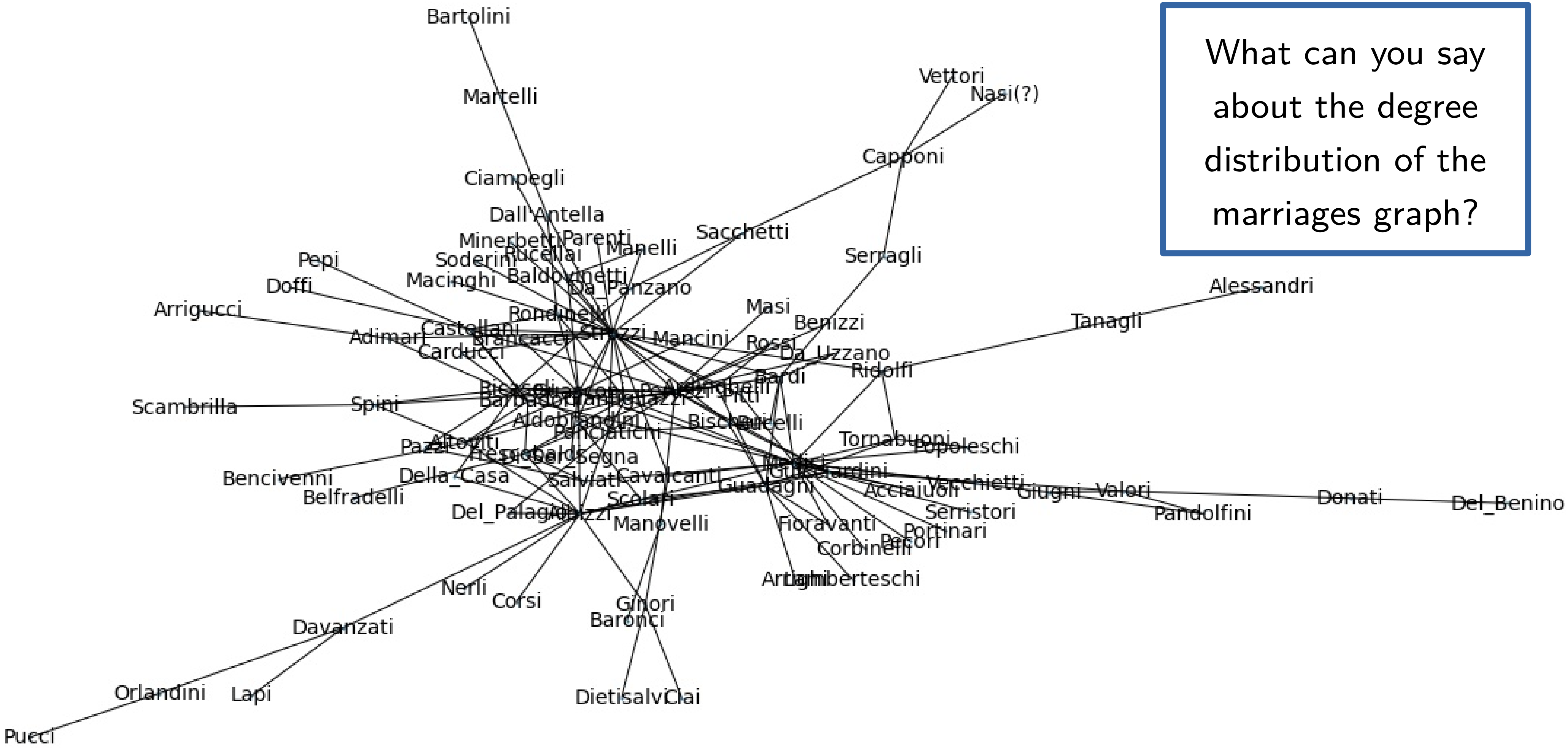
[Image source]

Marriages graph

- 96 nodes (families)
- 157 edges (marriages)
- Undirected and unweighted

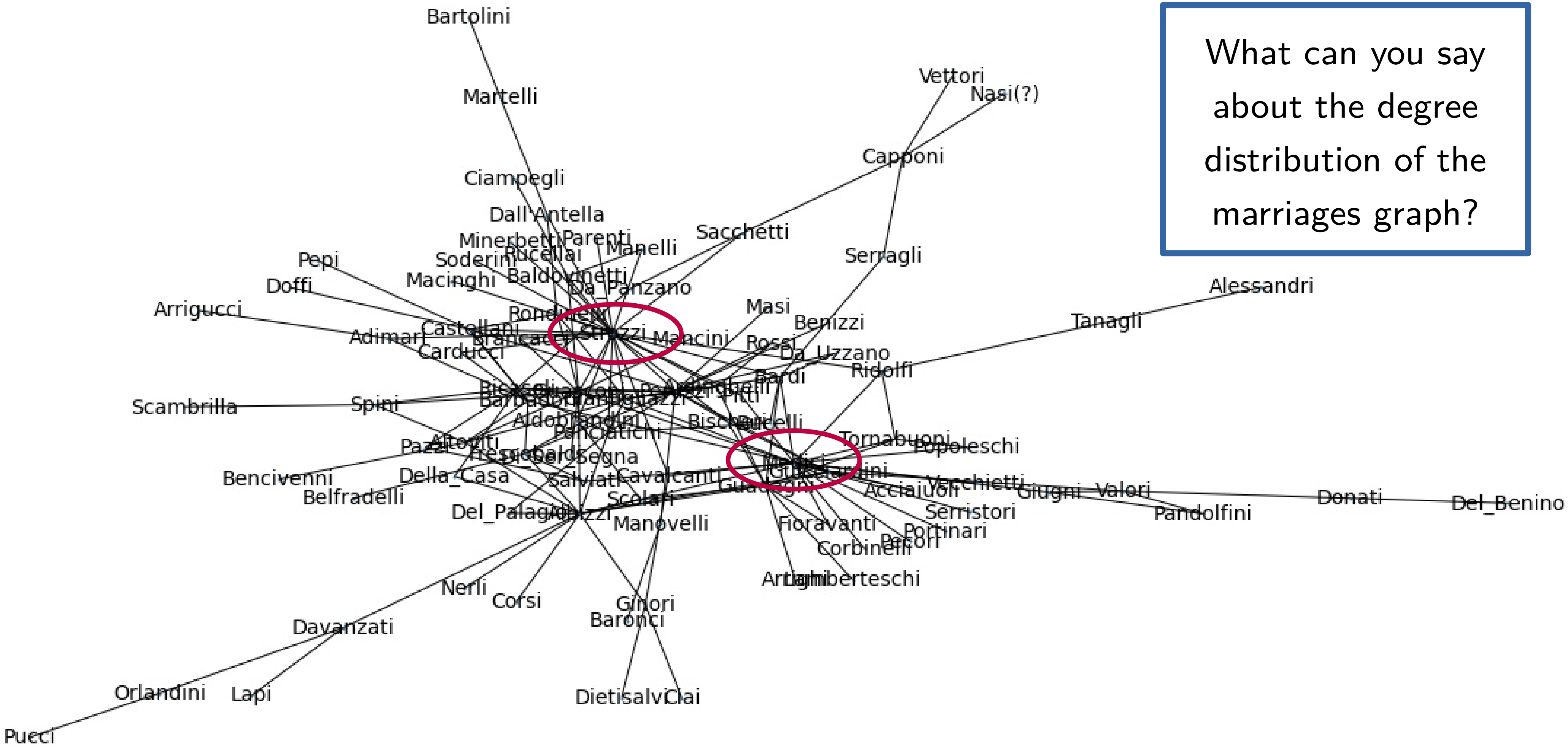


Marriages - giant connected component (90 nodes, 94%)



What can you say about the degree distribution of the marriages graph?

Marriages - giant connected component (90 nodes, 94%)



What can you say about the degree distribution of the marriages graph?

Closeness, betweenness, eigencentrality

Closeness	Betweenness	Eigencentrality
• Peruzzi 0.42	• Peruzzi 0.15	• Peruzzi 0.32
• Medici 0.44	• Medici 0.26	• Medici 0.27
• Strozzi 0.46	• Strozzi 0.35	• Strozzi 0.40

What can you say about the correlations of this with wealth/power?

Correlations

	Gwealth	Npriors	m_degree	m_closeness	m_betweenness	m_eigencentrality	c_degree	c_closeness	c_betweenness	c_eigencentrality
Gwealth	1.00	0.44	0.79	0.67	0.77	0.76	0.39	0.22	0.40	0.33
Npriors	0.44	1.00	0.69	0.53	0.71	0.63	0.31	0.03	0.24	0.19
m_degree	0.79	0.69	1.00	0.77	0.95	0.93	0.48	0.30	0.45	0.42
m_closeness	0.67	0.53	0.77	1.00	0.66	0.90	0.42	0.27	0.29	0.44
m_betweenness	0.77	0.71	0.95	0.66	1.00	0.81	0.43	0.25	0.45	0.33
m_eigencentrality	0.76	0.63	0.93	0.90	0.81	1.00	0.45	0.29	0.32	0.46
c_degree	0.39	0.31	0.48	0.42	0.43	0.45	1.00	0.70	0.84	0.87
c_closeness	0.22	0.03	0.30	0.27	0.25	0.29	0.70	1.00	0.61	0.81
c_betweenness	0.40	0.24	0.45	0.29	0.45	0.32	0.84	0.61	1.00	0.57
c_eigencentrality	0.33	0.19	0.42	0.44	0.33	0.46	0.87	0.81	0.57	1.00

Do you see the block structure in this matrix? What does it mean?
What is a good predictor of wealth/power?

Summary

Things to remember

- The analysis of social networks requires defining suitable graphs
- There is usually a step in which one compares this with domain-specific metrics