

# K-Core Decomposition

**Social Networks Analysis and Graph Algorithms**

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



**Universitat  
Pompeu Fabra**  
*Barcelona*

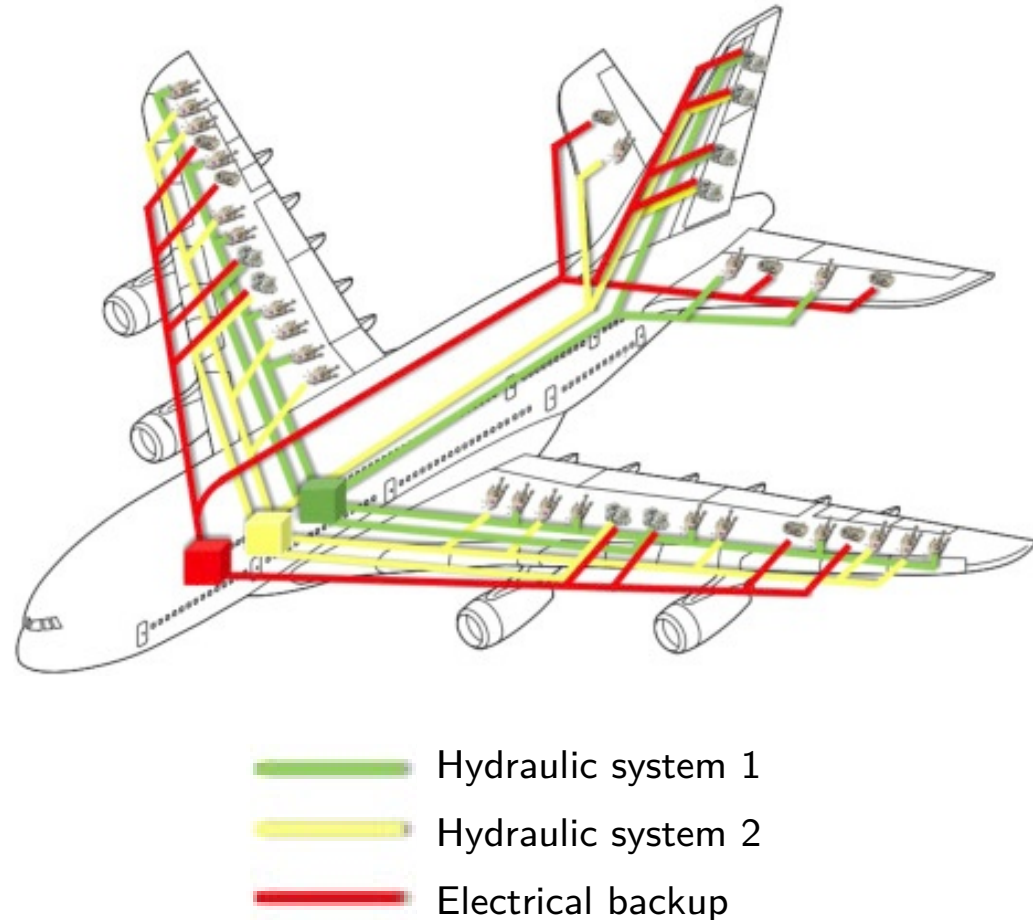
# Sources

- A. L. Barabási (2016). Network Science – [Chapter 09](#)
- A. Beutel, L. Akoglu, C. Faloutsos (2015). [Tutorial at KDD](#)
- A. Frieze, A. Gionis, C. Tsourakakis (2013). “Algorithmic techniques for modeling and mining large graphs (AMAZING)” [Tutorial at KDD](#)
- V. E. Lee, N. Ruan, R. Jin, C. Aggarwal (2010). A survey of algorithms for dense sub-graph discovery. [Chapter 10](#) of “Managing and Mining Graph Data”
- URLs cited in the footer of slides

# Robustness

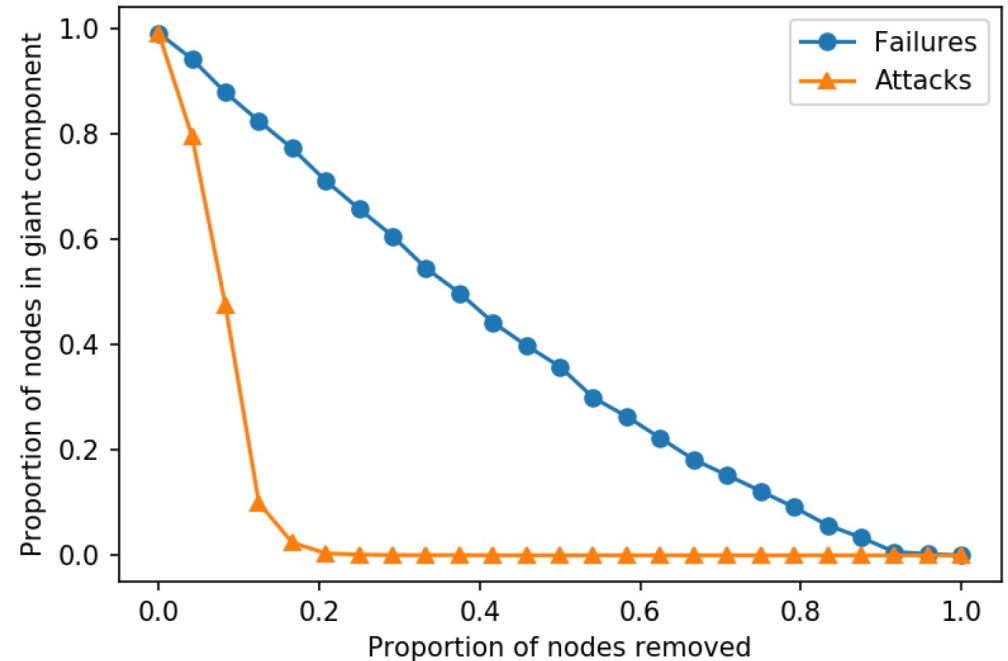
# Robustness to failure

- A system is **robust** if the failure of some of its components does not affect its function
- Aircrafts, for instance, have:
  - Separate primary and back-up flight instruments (airspeed, altimeter, ...);
  - 3+ independent hydraulic systems
  - Primary and emergency landing gear;
  - Multiple sources of power



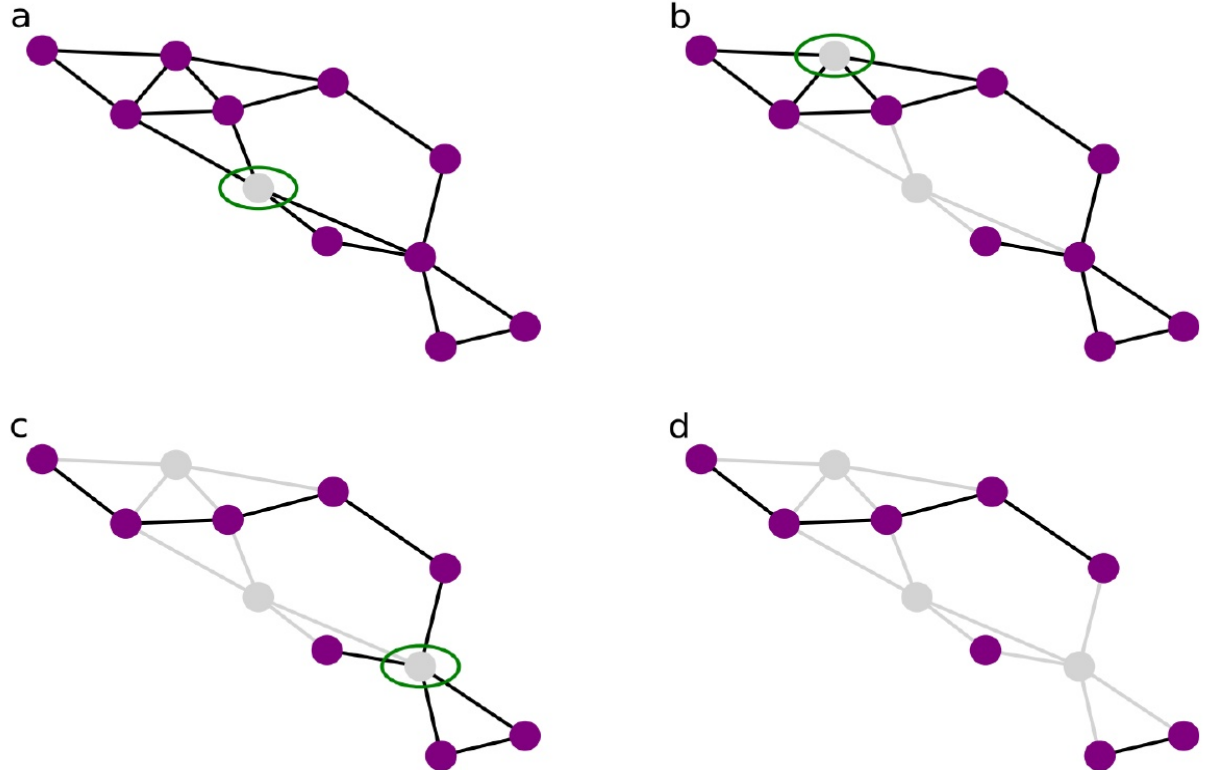
# Robust networks maintain connectedness

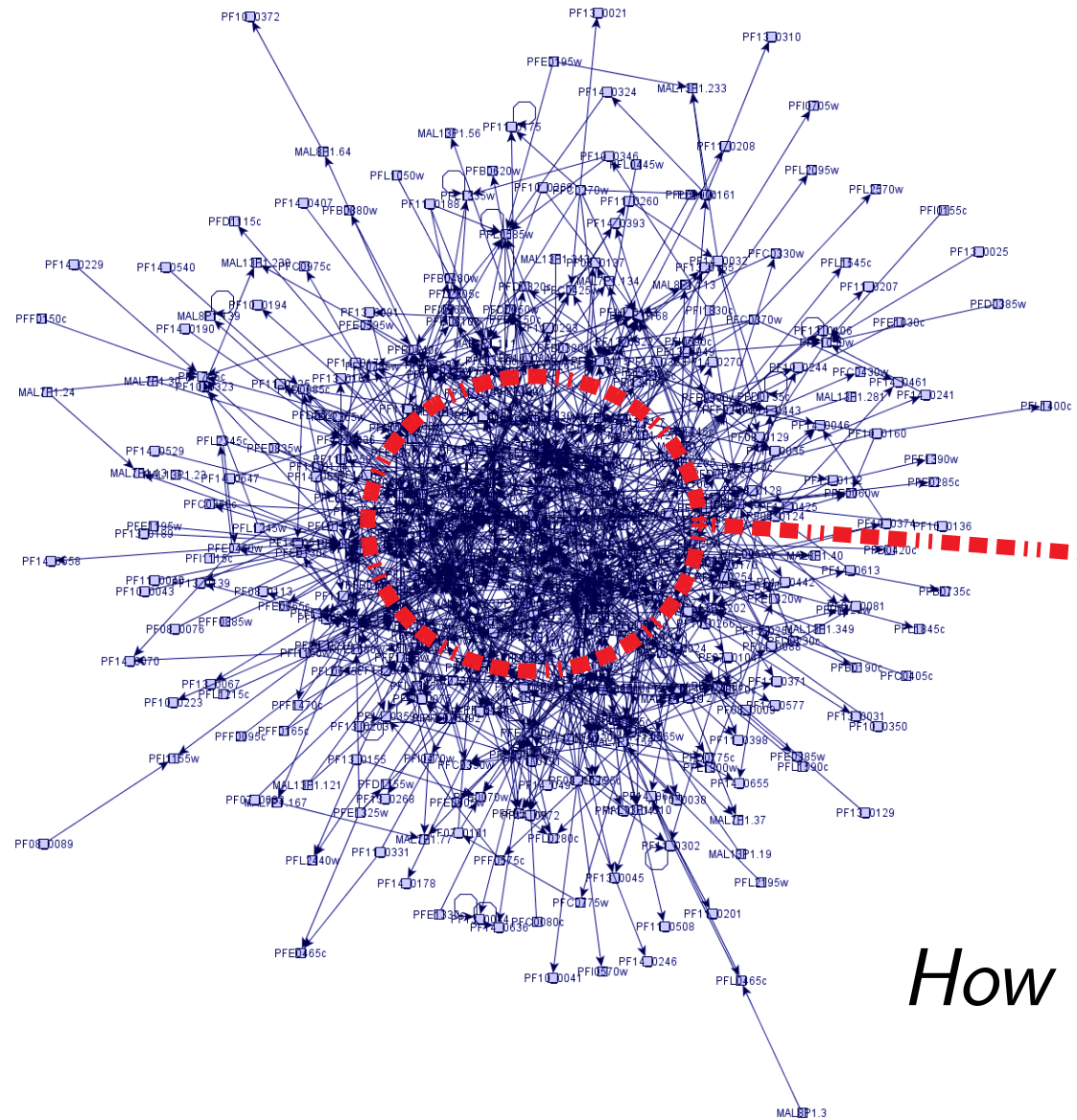
The size of the giant component in a scale-free network is reduced slowly by random removals (“failures”), but reduced quickly by removing high-degree edges (“attacks”)



# Targeted removal of nodes

The targeted removal of 4 nodes **disconnects** this graph into three components





Many networks look like “hairballs”

They have a **core** and a **periphery**

*How can we find the core?*

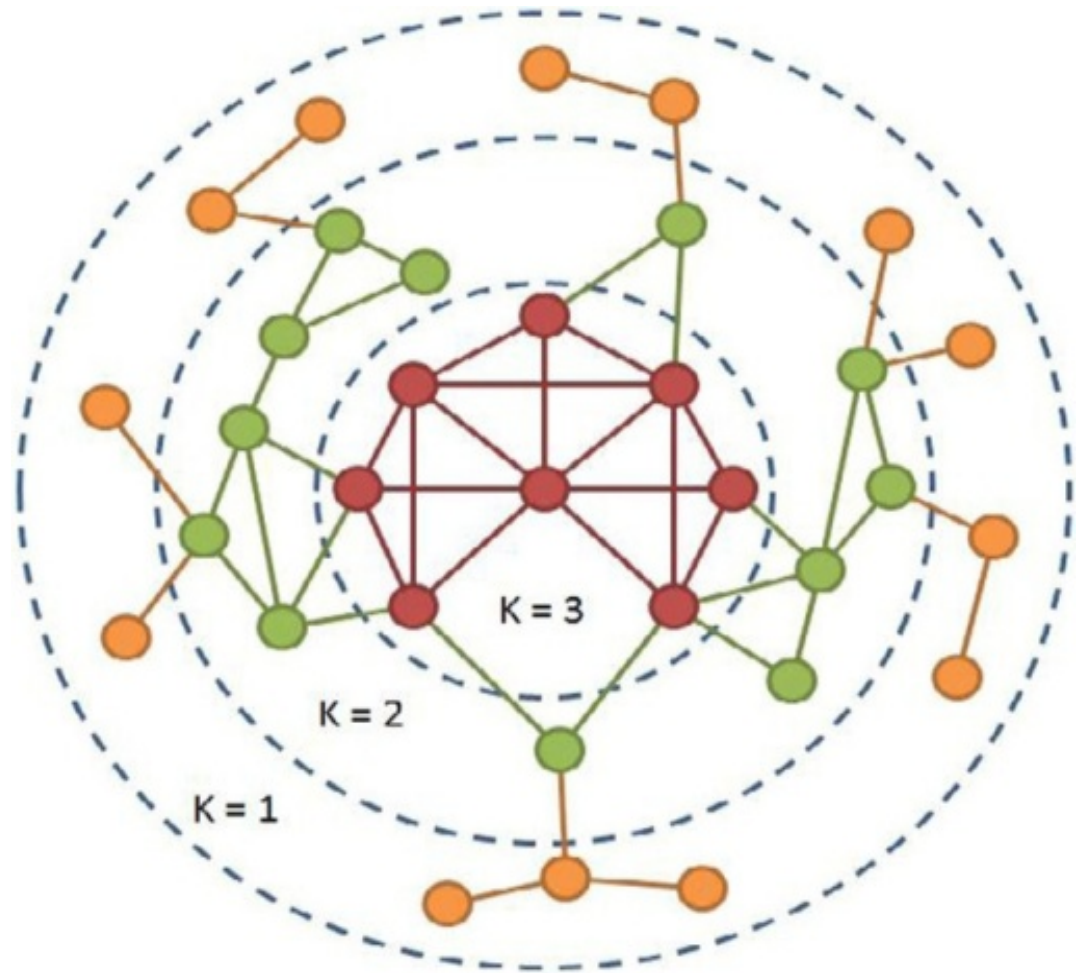
**k-core decomposition**  
is a method to decompose  
a graph into *layers*



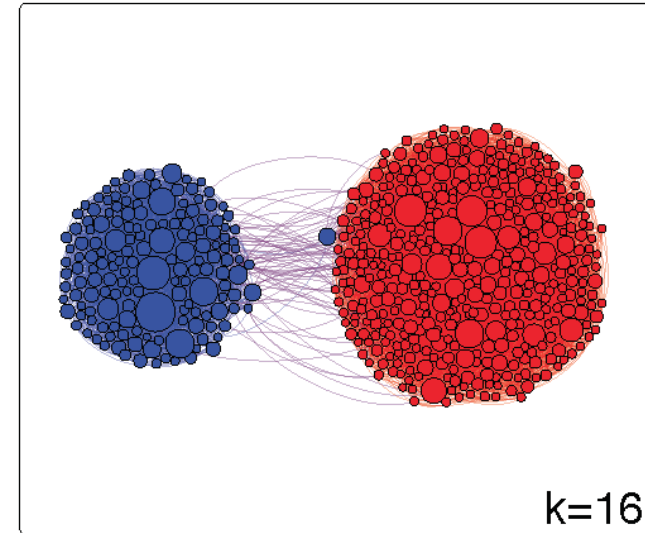
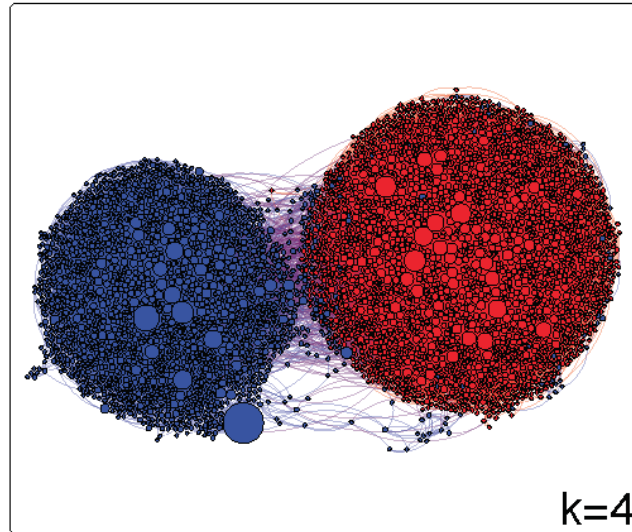
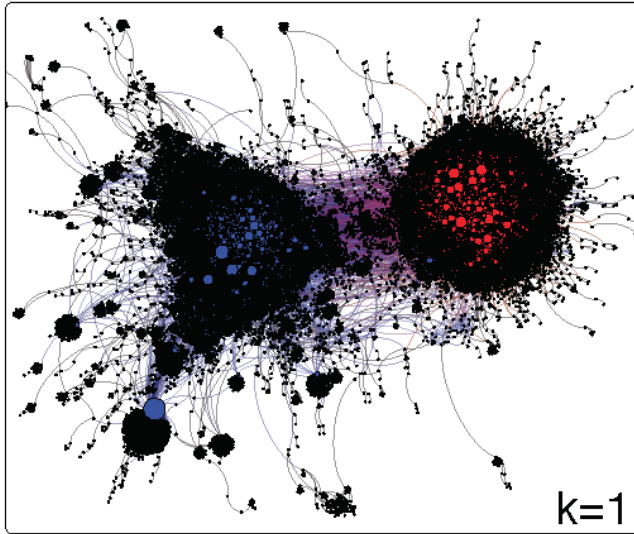
# k-core decomposition

- Remove all nodes having degree  $\leq 1$  until there are no such nodes
  - Those are in the 1-core
- Remove all nodes having degree  $\leq 2$  until there are no such nodes
  - Those nodes are in the 2-core
- Remove all nodes having degree  $\leq 3$  until there are no such nodes
  - Those nodes are in the 3-core
- Etc.

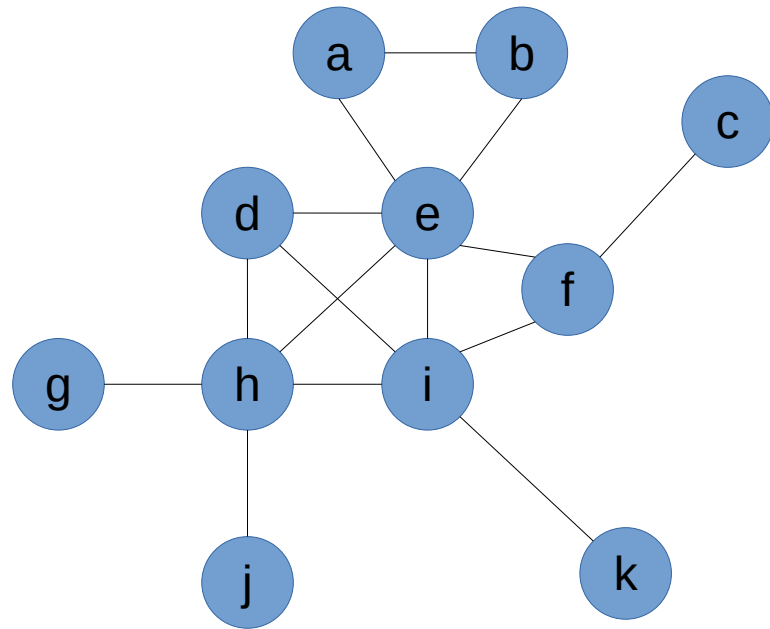
# Example 1



# Example 2



# Exercise



For each node in the graph, indicate the max k-core to which it belongs

(Mark each node with a number, upload an image of the result.)

[Source]

Pin board: <https://upfbarcelona.padlet.org/chato/tt19-k-core-decomposition-575mnnke44c5ybyl>



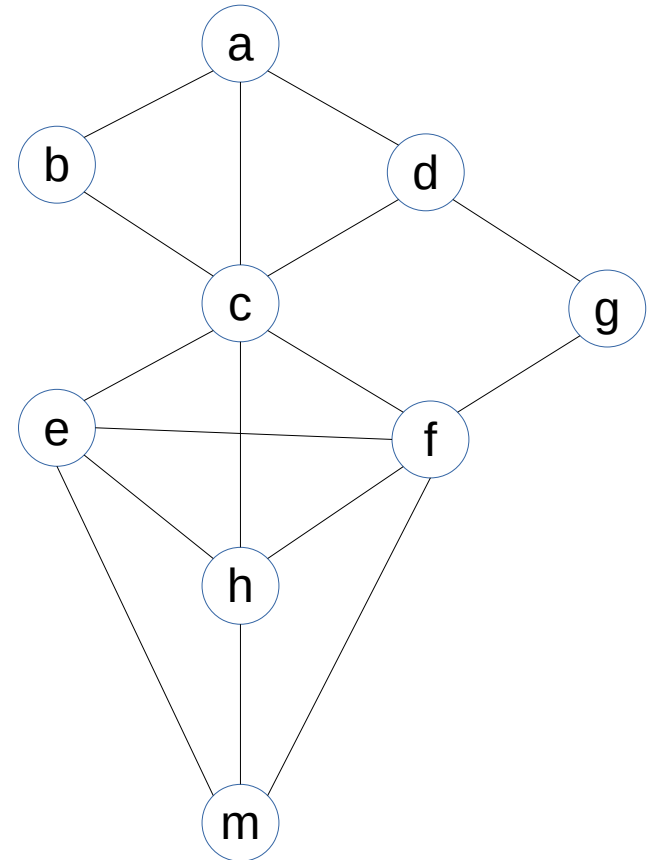
# Summary

# Things to remember

- What does it mean for a network to be robust?
- What is the  $k$ -core decomposition
- How to compute it on a graph

# Practice on your own

Find the 3-core of this graph



Solution by Vivekanand Khyade (start at 01:23)  
<https://youtu.be/8sNZ5d8eNC8?t=83>