Statistical Analysis of Networks

Identifying Subgroups

Learning Goals

- Examine conceptualization of *cohesion*.
- Understand conceptual definitions of cohesion.
- Understand approaches to operationalizing cohesion.

Introduction

- So far…
  - We have covered how to think about the structural *position* of individuals: centrality.
  - Now, we would like to think more about the global properties of a network.

Cohesive Subgroups

- One approach is to focus on *subgroups* within the network.
  - The structure of your group, not just your position in that group, might be of some theoretical importance.
    - Example: structural embeddedness and trust (e.g. Granovetter, 2017)

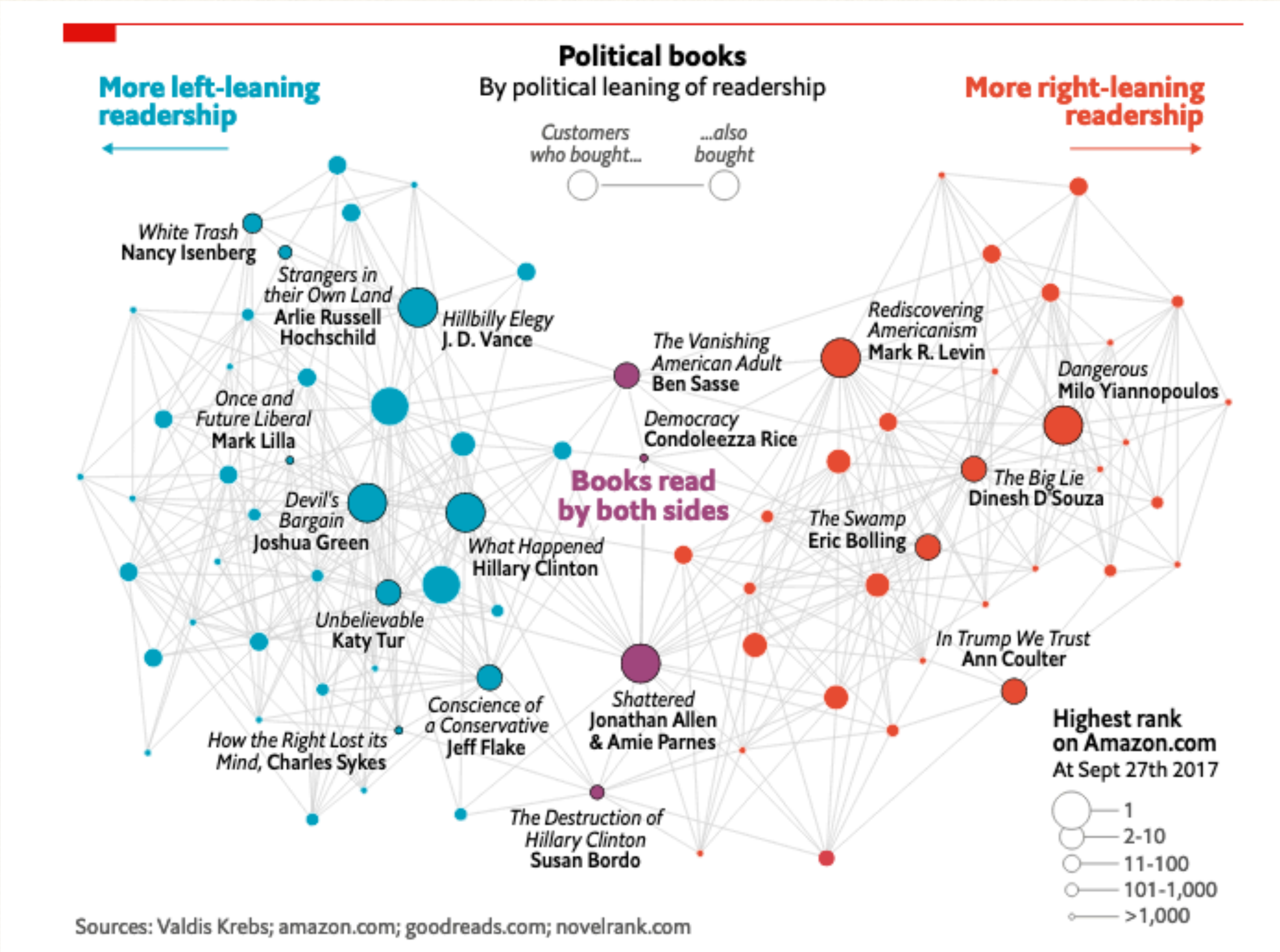
Cohesive Subgroups

- Or…
  - The extent to which there are different groups in a network may be of theoretical importance.
    - Example: fragmented networks and distrust.

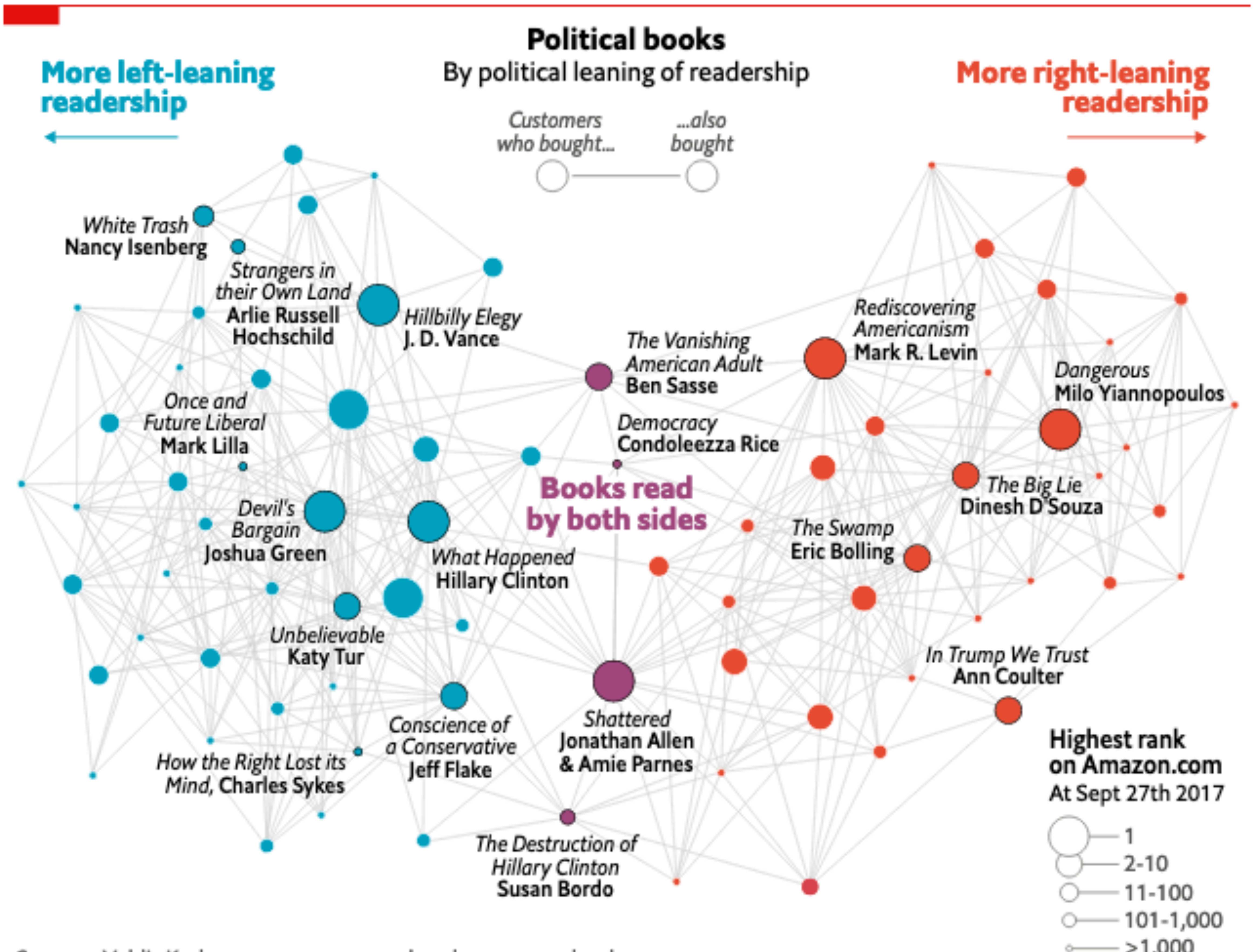
Cohesive Subgroups

- Or…
  - Fragmented networks and the consumption of science.

Empirical Example:
Political Fractionalization and Media Consumption



from: https://www.economist.com/books-and-arts/2017/09/30/many-writers-try-to-span-americas-political-divide and https://www.nature.com/news/how-to-judge-a-book-by-its-network-1.21771



Sources: Valdis Krebs; amazon.com; goodreads.com; novelrank.com

Cohesive Subgroups

- The logic here is that we can identify *cohesive* groups.
  - The methodological task is to conceptualize and operationalize what we mean by "cohesive".

When we say a *group* is “cohesive,” what do we mean conceptually?

Conceptualization

- *How do we define cohesion?*
- Four general properties of cohesive groups (Wasserman & Faust,1994: 251):
    - Mutuality of ties
    - The closeness or reachability of subgroup members
    - The frequency of ties among members
    - The relative frequency of ties among subgroup members compared to non-members

Conceptualization

- These are conceptual definitions, from which we can derive operational definitions.
  - *What would a graph with these properties look like?*

Approaches

- **Bottom-up**
  - Dense connections are built-up from simpler dyads and triads to more extended dense clusters.
    - Example: clique, n-clique
- **Top-down**
  - Looking at the whole network, sub-structures are areas of the graph that seem to be locally dense, but separated to some degree, from the rest of the graph.
    - Example: components, k-cores, community detection

Graph Notation Refresher

- Definition of a **graph**: $G = (N, L)$
  - Node/Vertex set: $N = \{n_1, n_2 \ldots, n_g\}$
  - Line/Edge set: $L = \{l_1, l_2 \ldots, l_L\}$
    - There are N nodes/vertices and L lines/edges in a graph.

Graph Notation Extended

- Definition of a **subgraph**:
  - Subgraph G_s of graph G consists of the subset of nodes N_s.
    - Where N_s is a subset of the node/vertex set N.

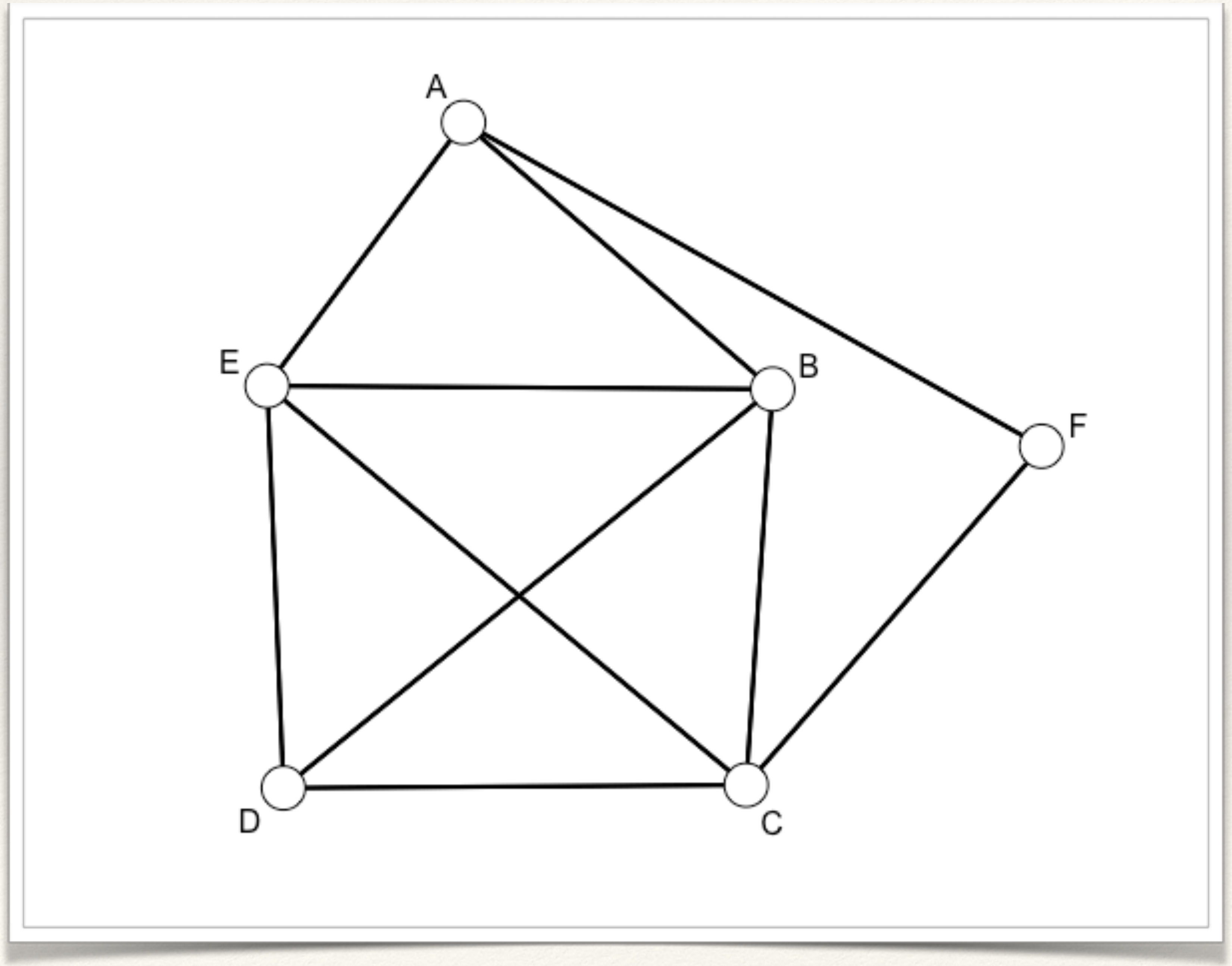
- The **distance**, $d(n_i,n_j)$, is the length of the path between i and j.
- And a **geodesic** is the shortest path between two nodes.

Undirected Networks

Cliques

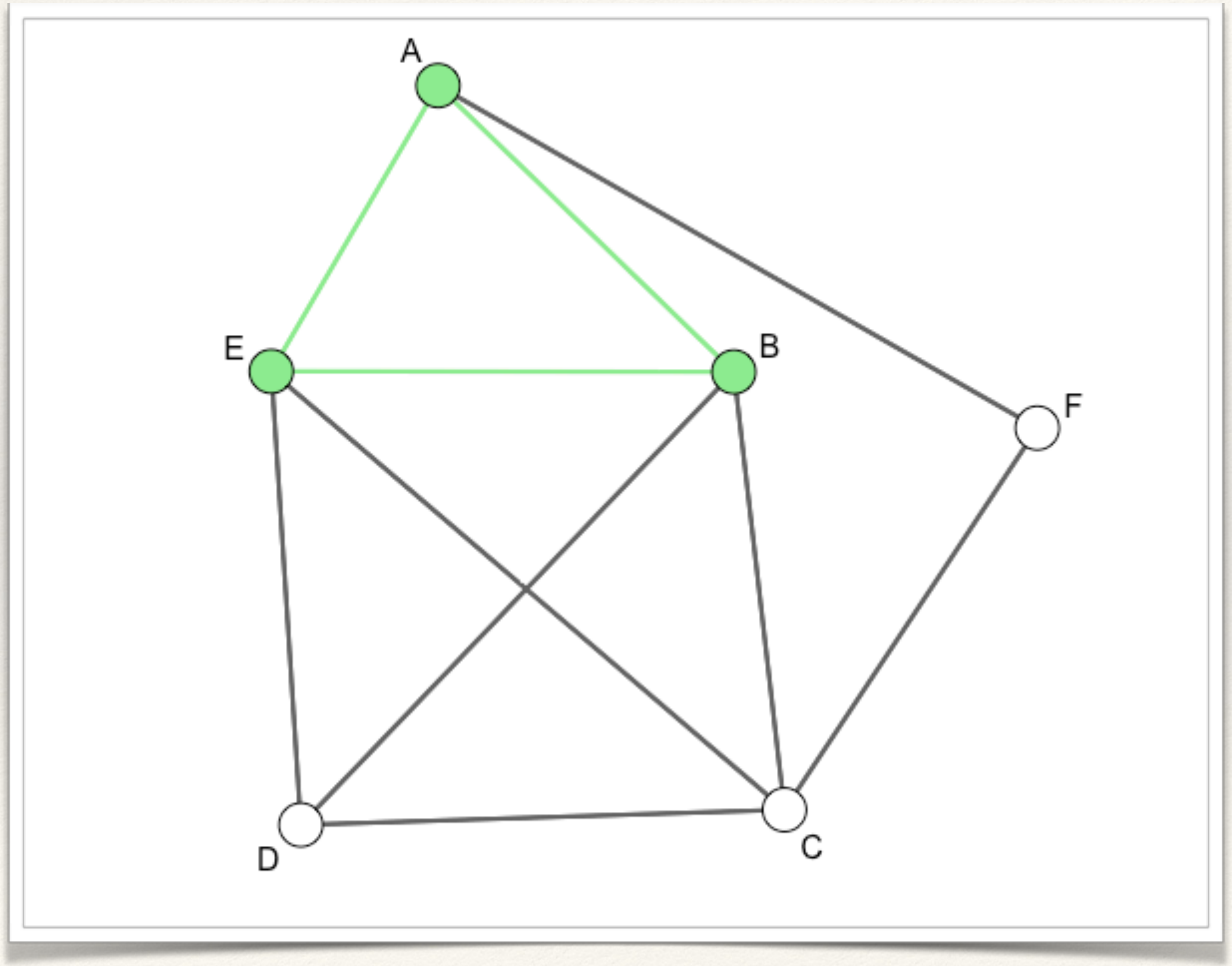
- A *clique* in a graph is a subgraph of three or more nodes such that:
  - all nodes are **adjacent** to all other nodes
  - and there are **no other nodes** that are also adjacent to all other nodes.

Cliques



What are the subgraphs that meet the conditions we defined?

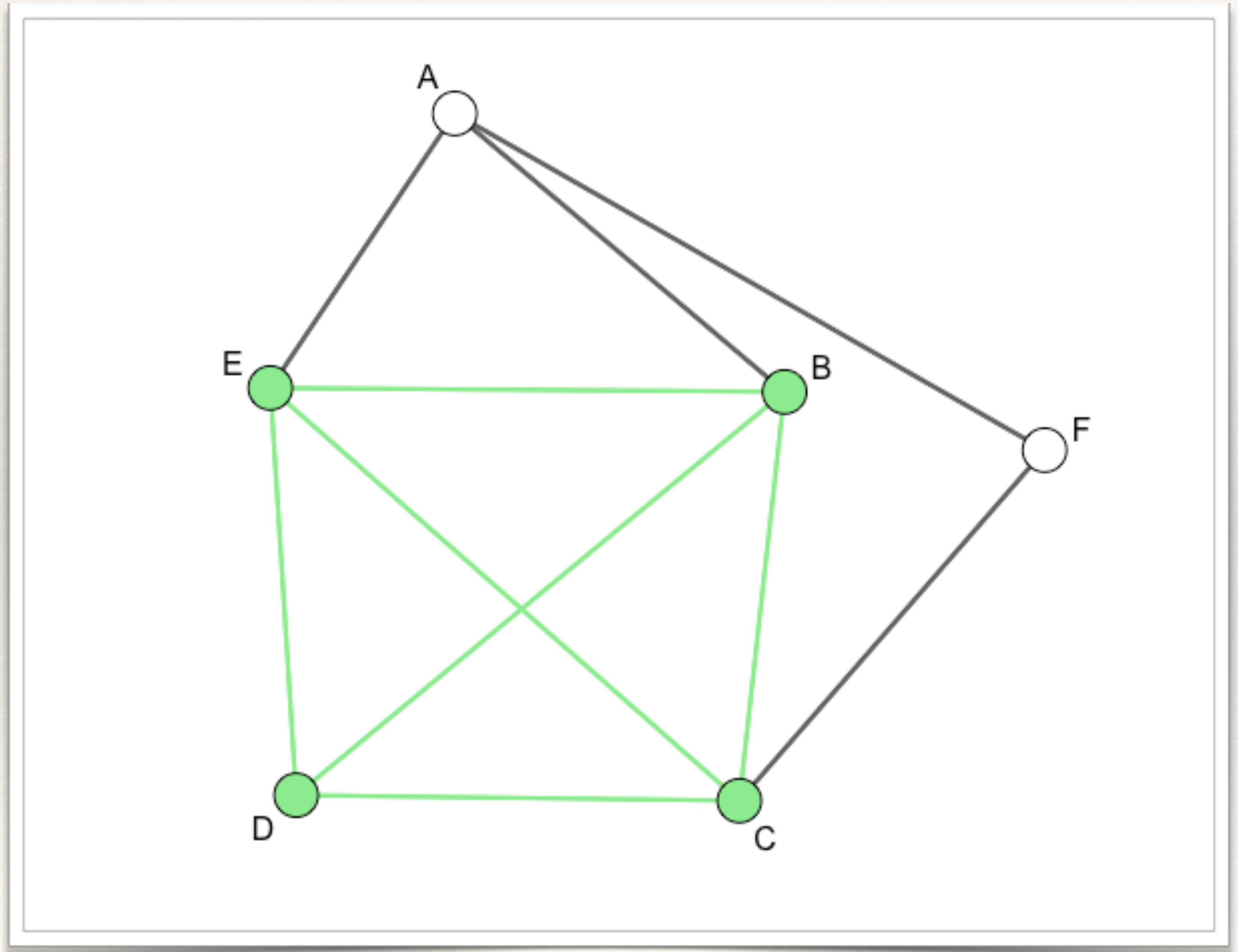
Cliques



What are the subgraphs that meet the conditions we defined?

{A, B, E}

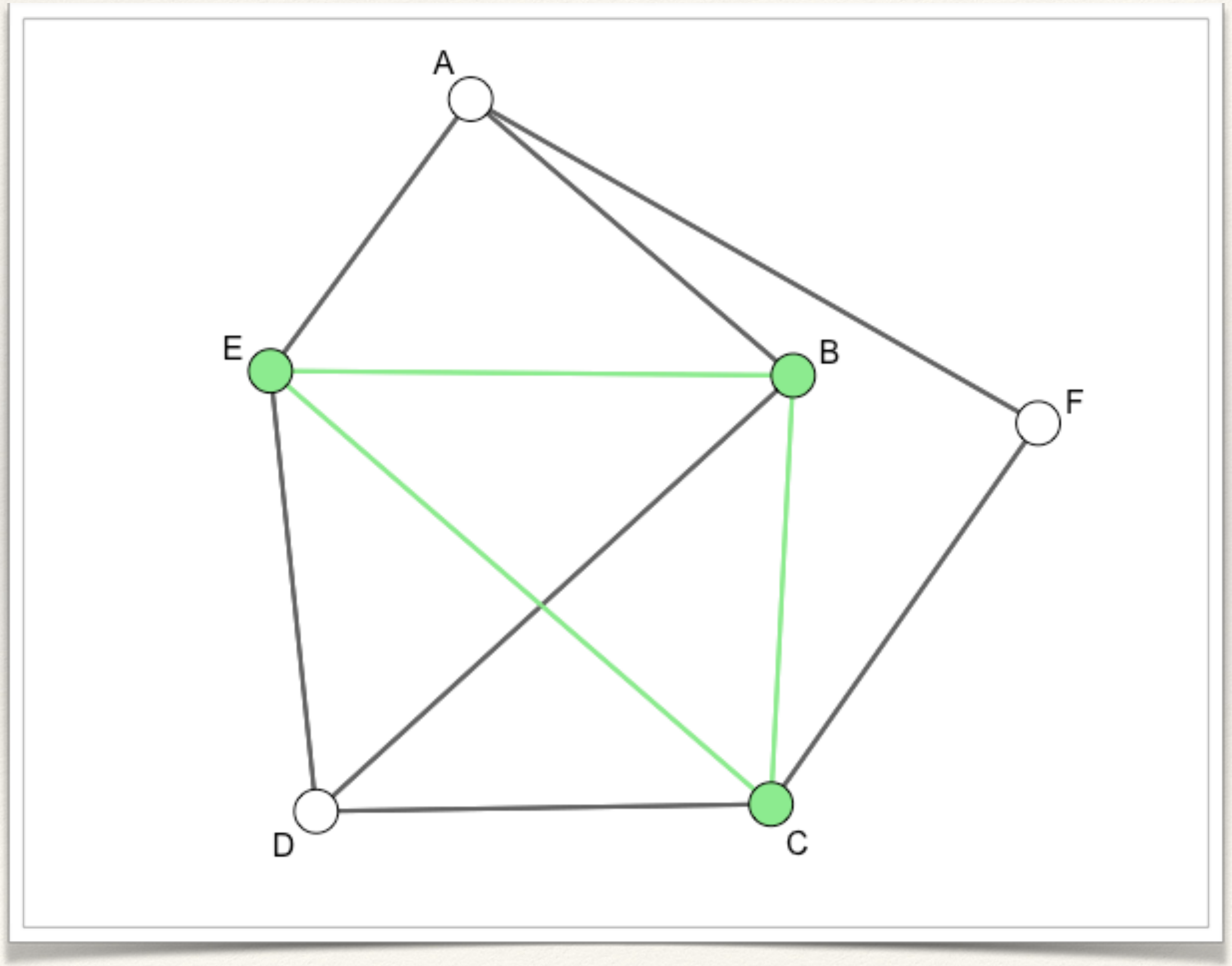
Cliques



What are the subgraphs that meet the conditions we defined?

{B, C, D, E}

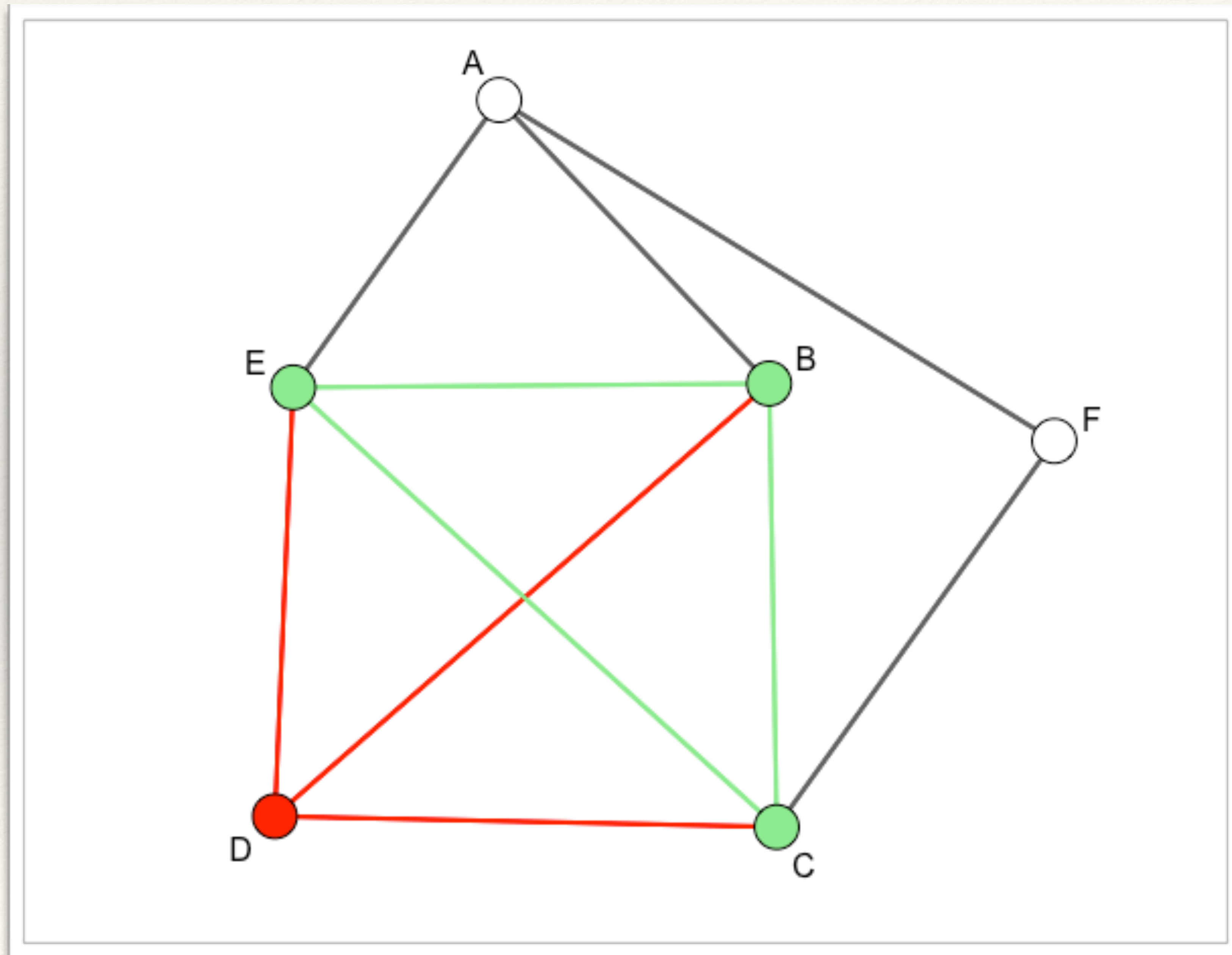
Cliques



What are the subgraphs that meet the conditions we defined?

Why not {E, B, C}?

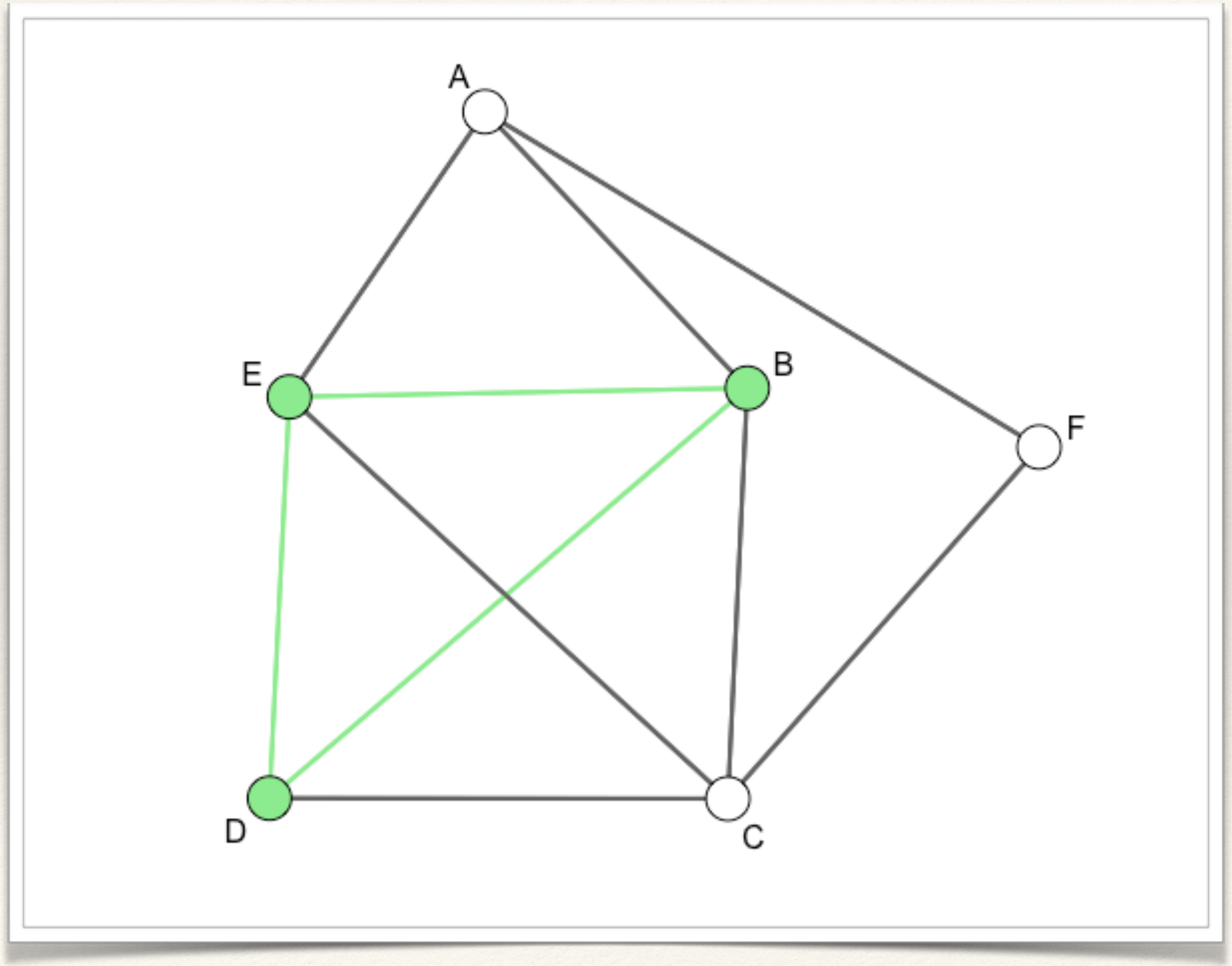
Cliques



What are the subgraphs that meet the conditions we defined?

Because of D!

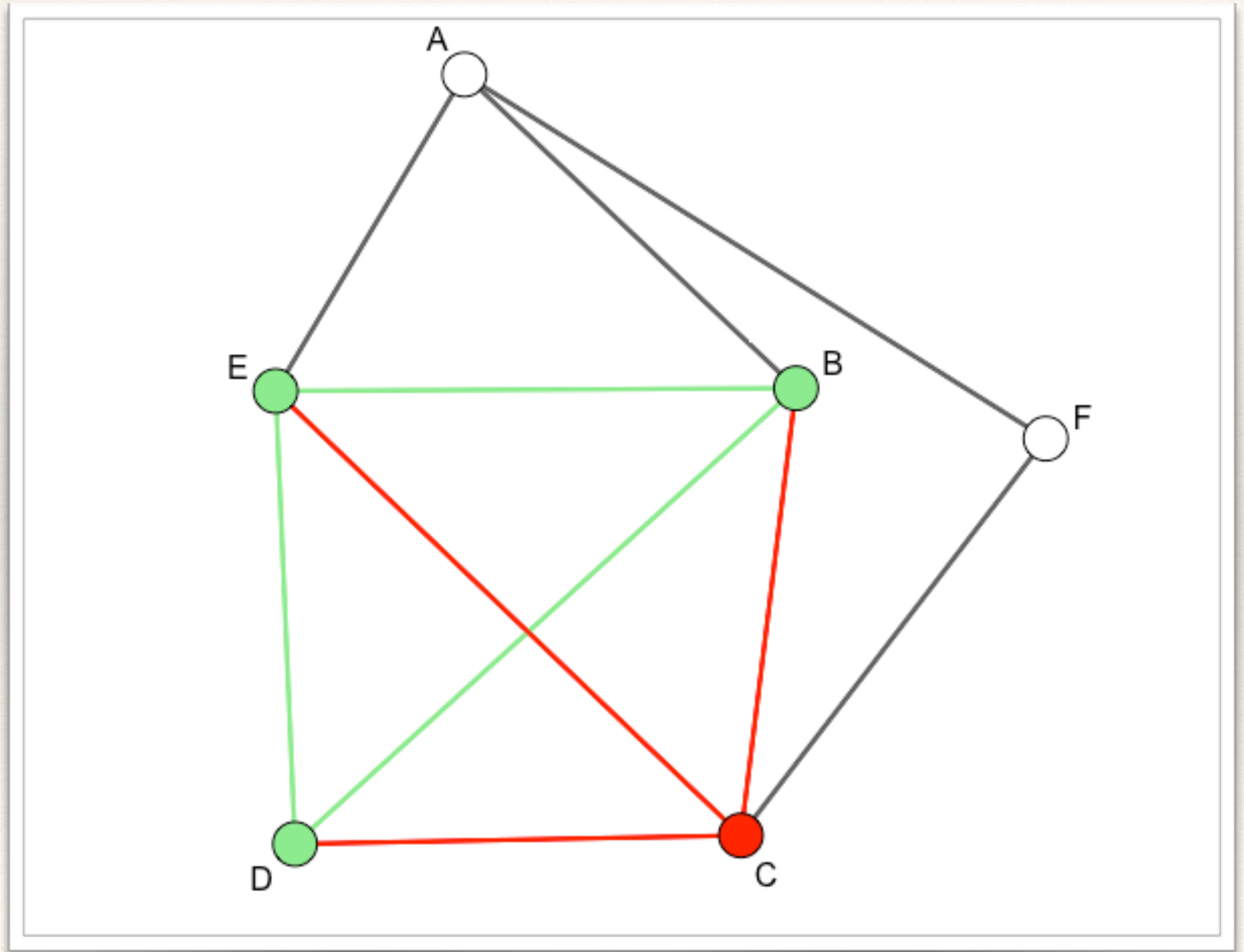
Cliques



What are the subgraphs that meet the conditions we defined?

Why not {E, B, D}?

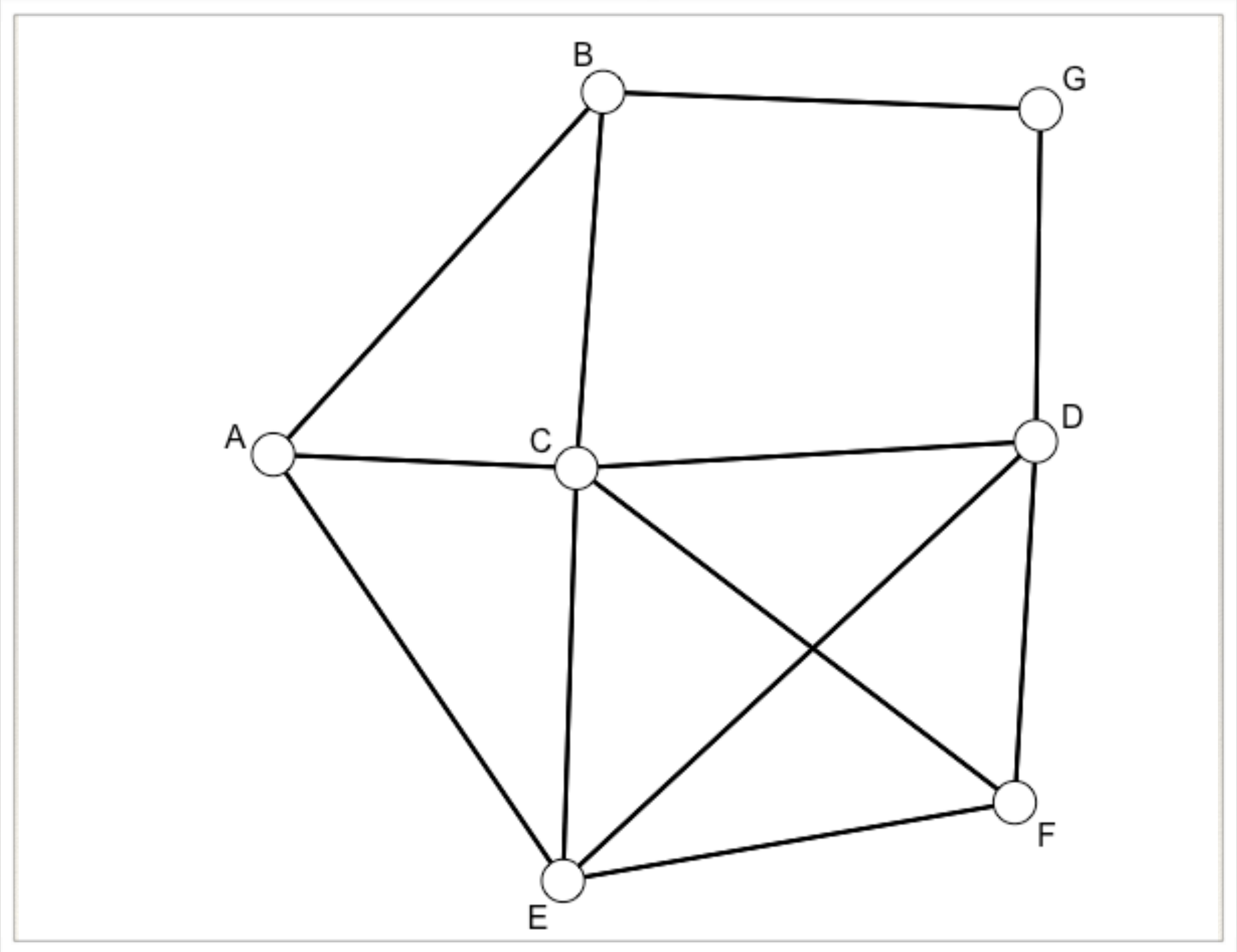
Cliques



What are the subgraphs that meet the conditions we defined?

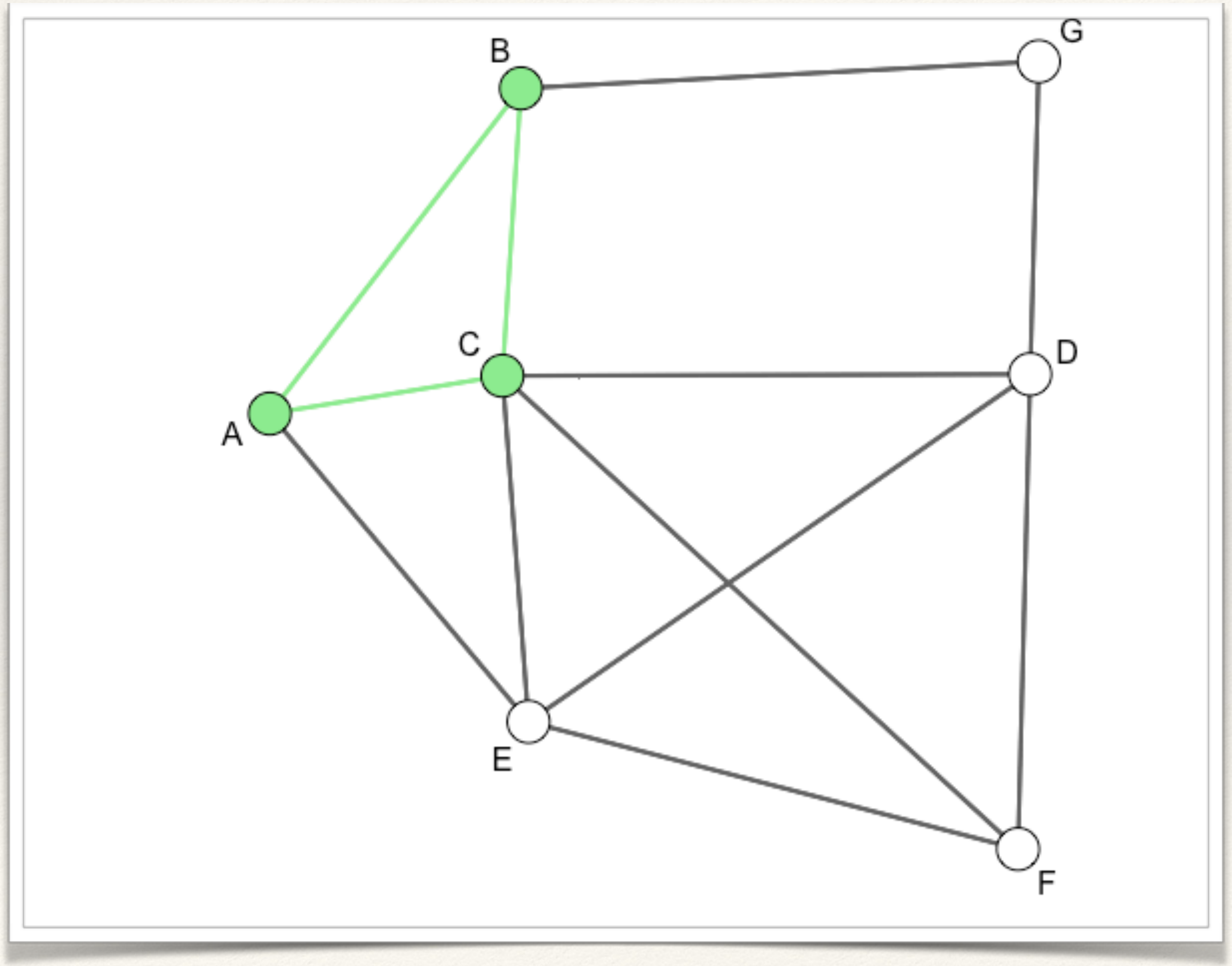
Because of C!

Another example



What are the subgraphs that meet the conditions we defined?

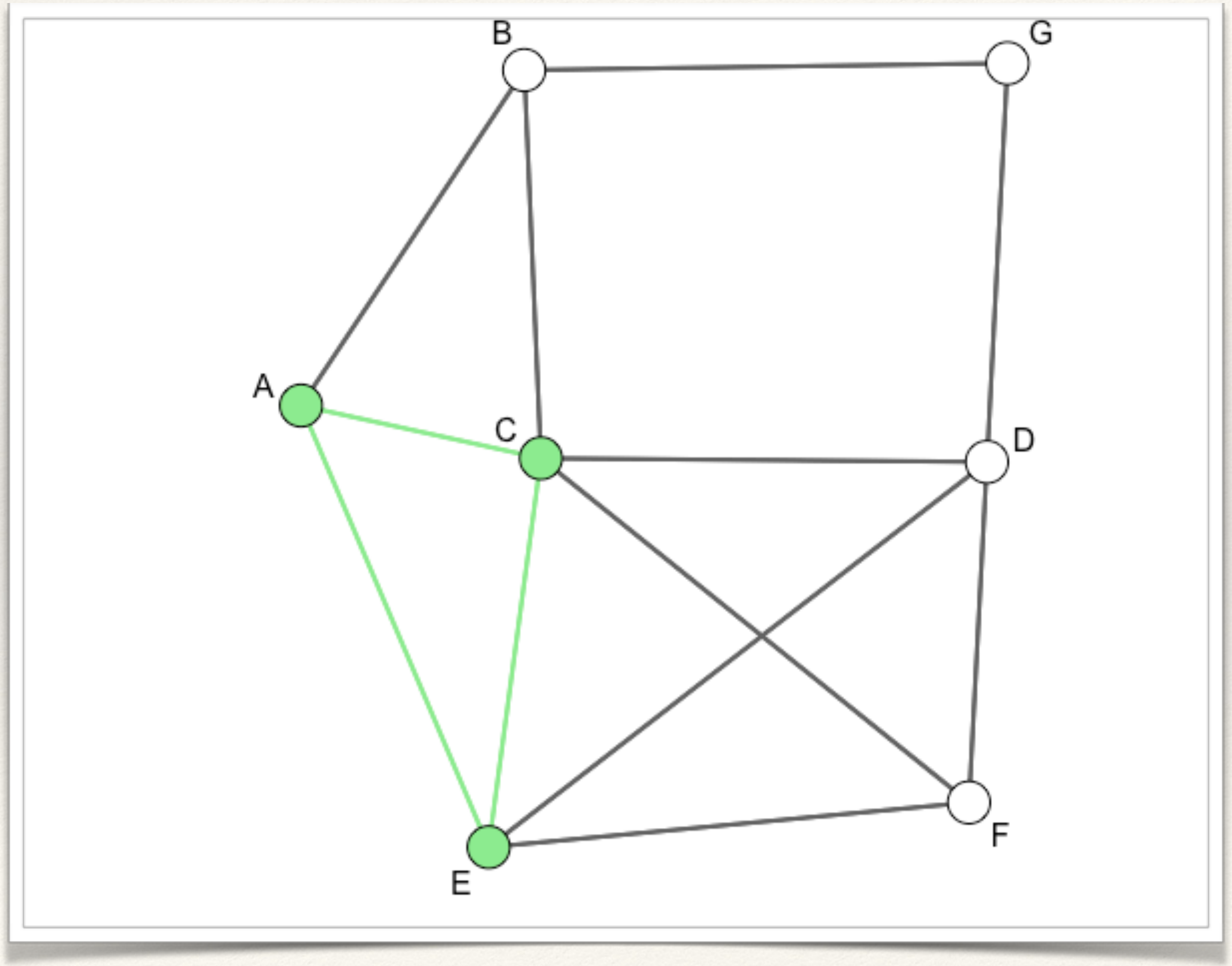
Another example



What are the subgraphs that meet the conditions we defined?

{A, B, C}

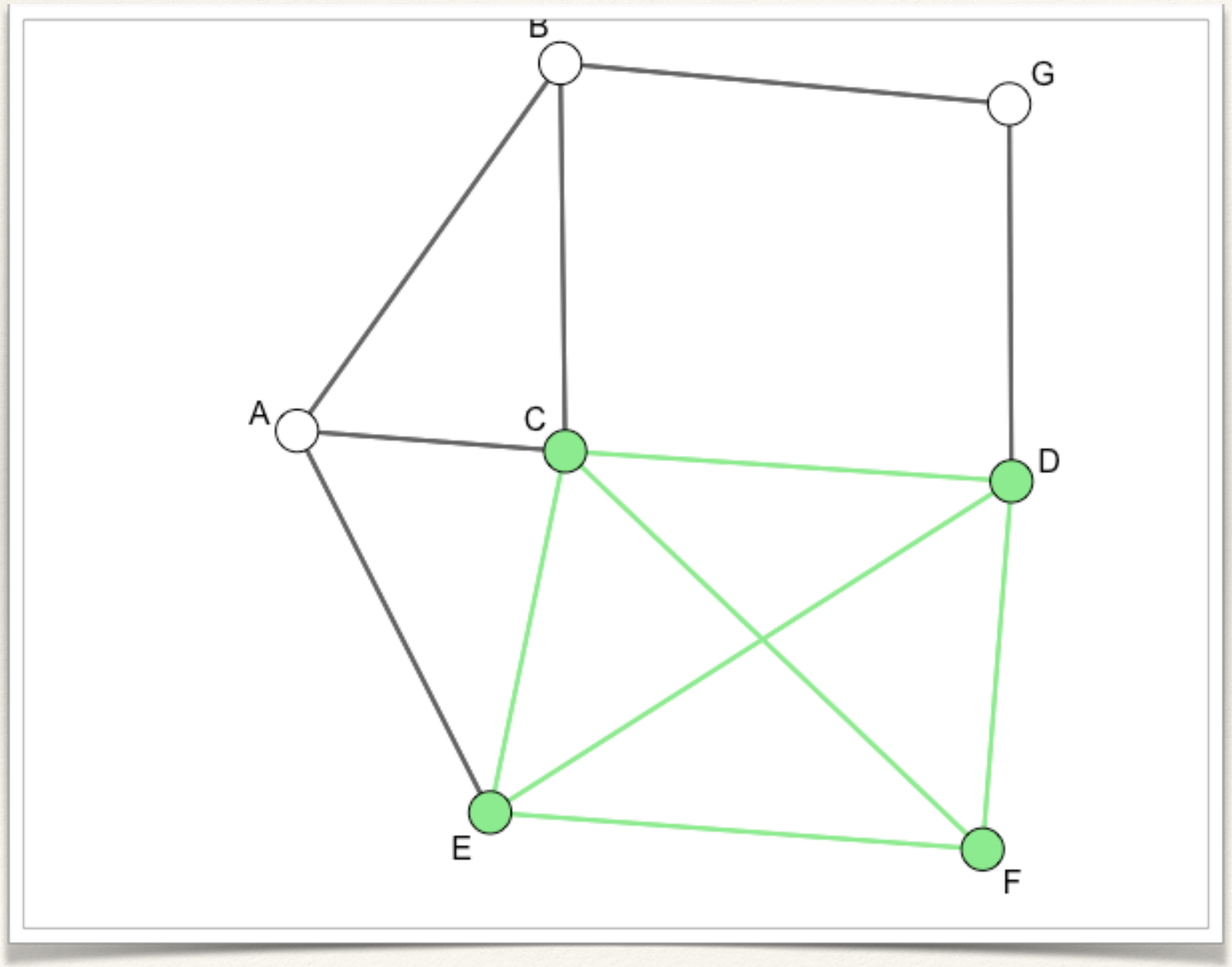
Another example



What are the subgraphs that meet the conditions we defined?

{A, C, E}

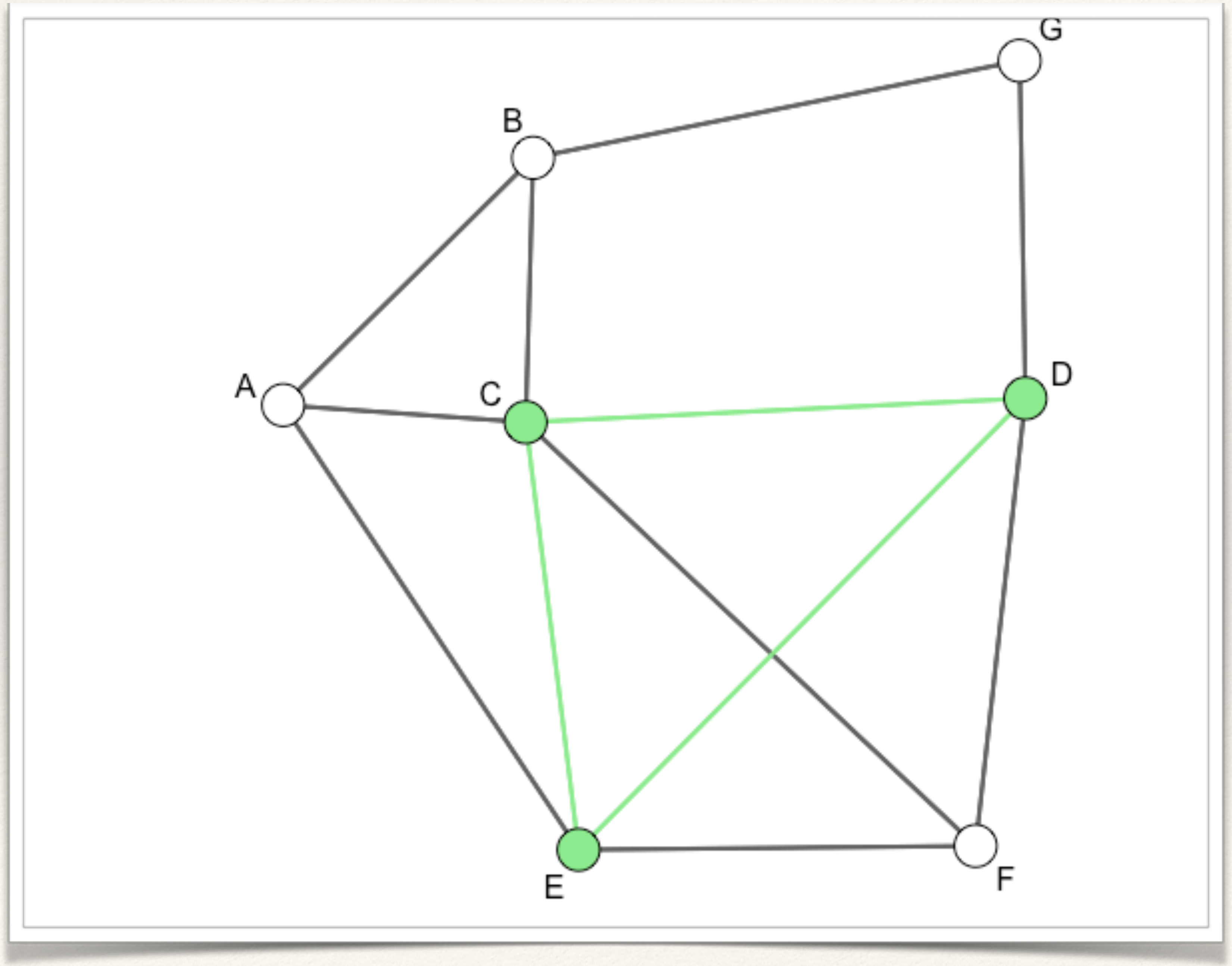
Another example



What are the subgraphs that meet the conditions we defined?

{C, D, E, F}

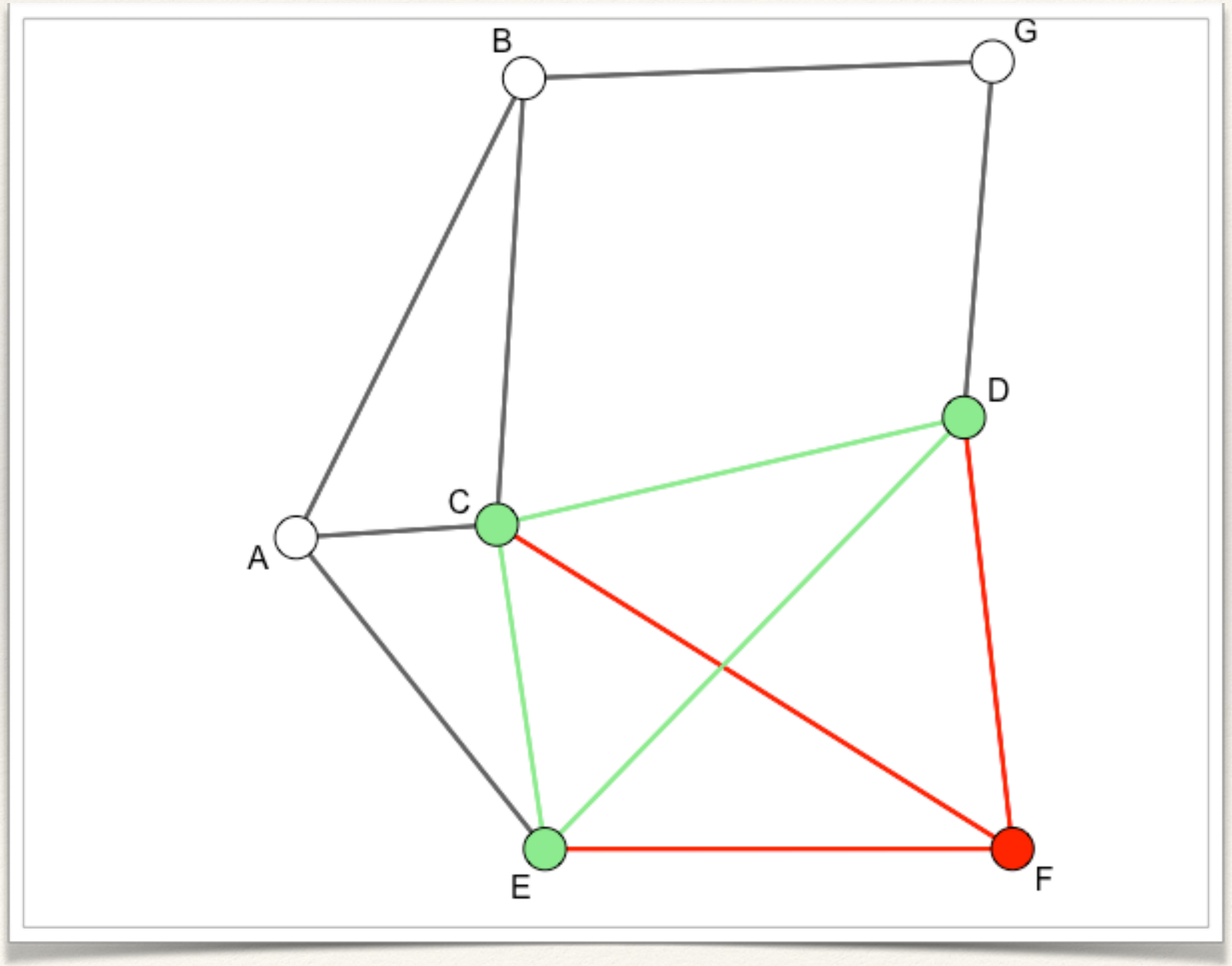
Another example



What are the subgraphs that meet the conditions we defined?

Why not {C, D, E}?

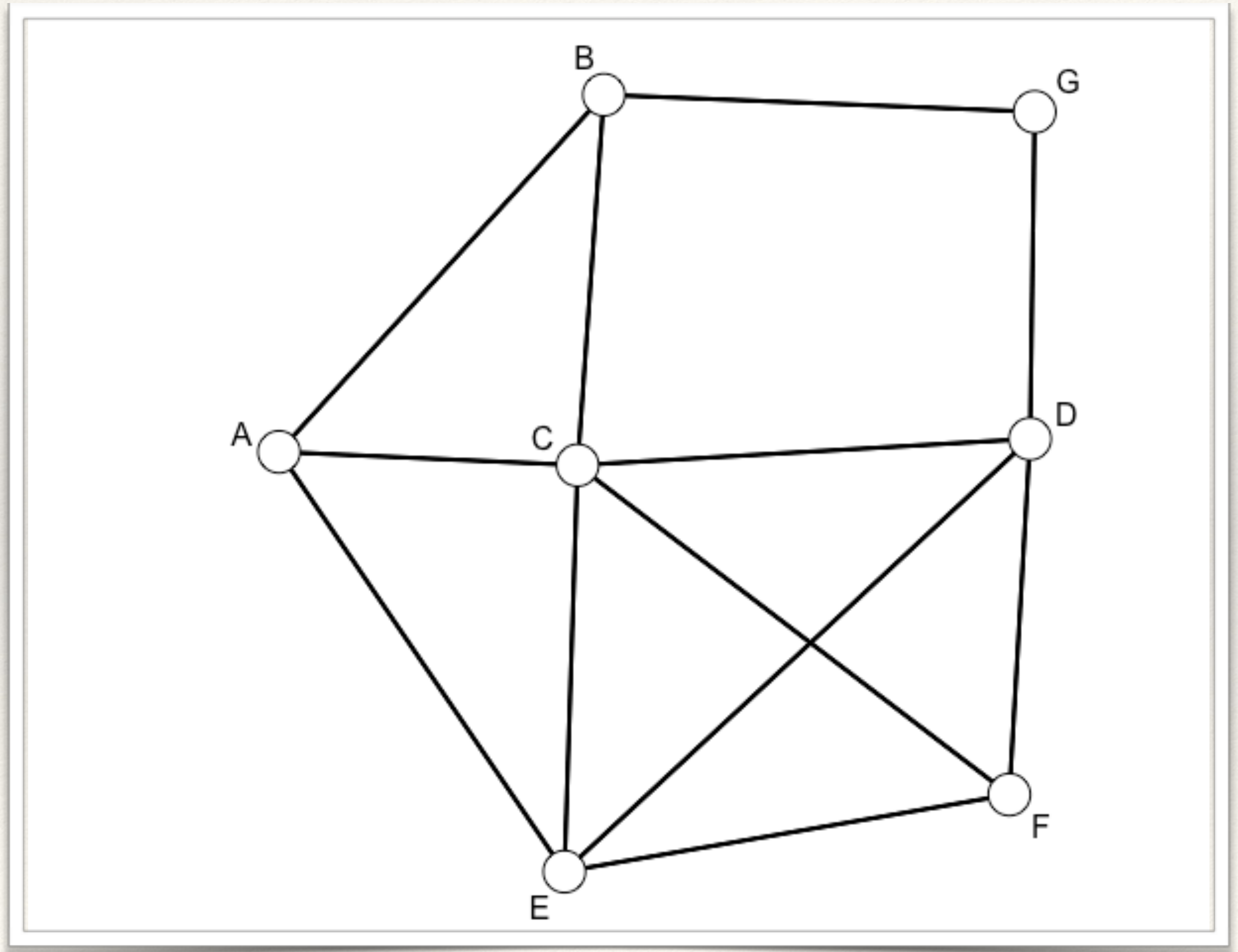
Another example



What are the subgraphs that meet the conditions we defined?

F!!!

Another example



Same issue for:

{C, E, F}

{C, D, F}

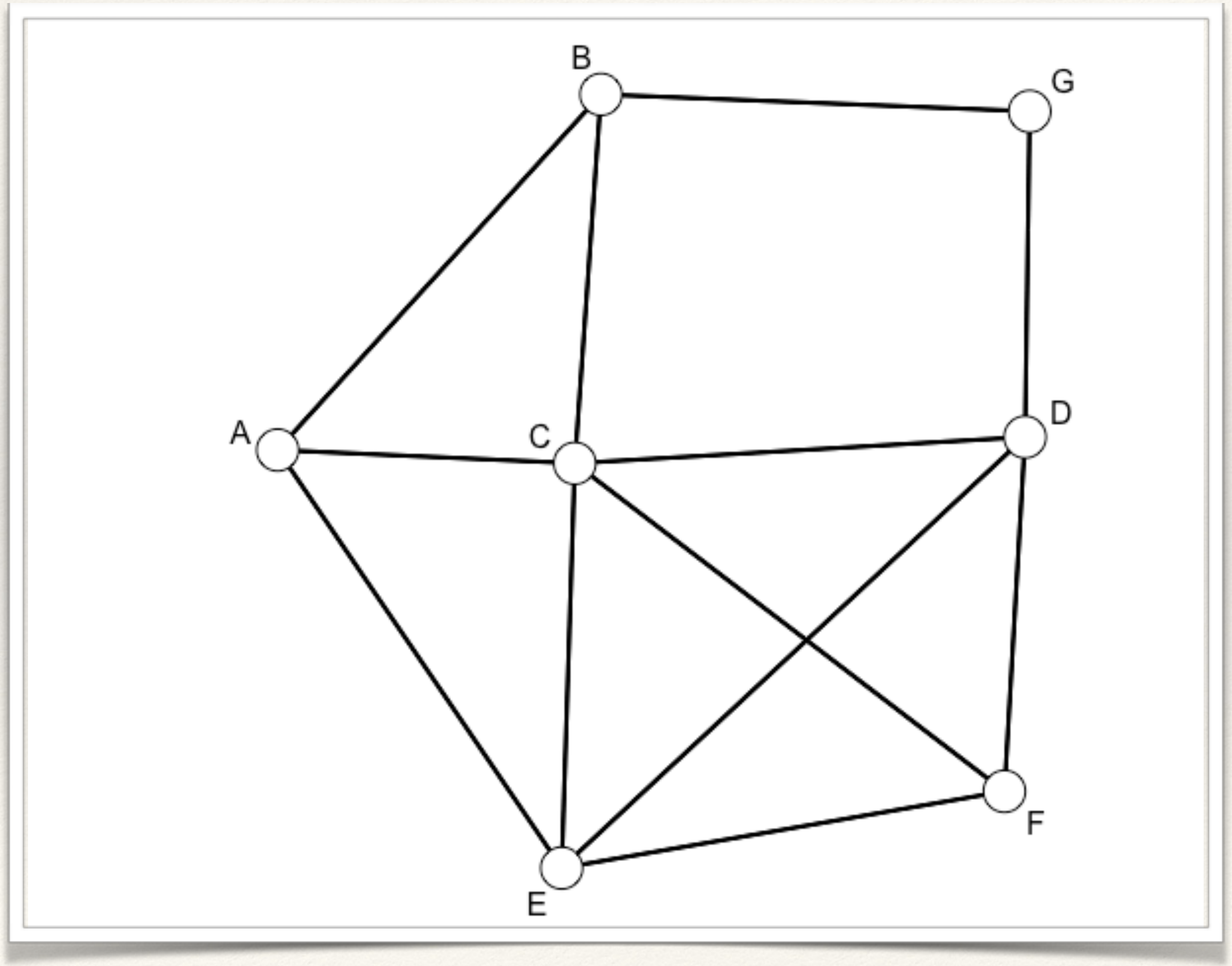
{D, E, F}

Limitations of Clique Definition

- Although cliques in a graph may overlap, no clique can be entirely contained within an other clique.
  - This means the subgraph is **maximal** in that the property holds for the subgraph but does not hold if additional nodes and the lines incident with them are added (see Wasserman & Faust, 1994: 253).

- We could relax this, and allow cliques to be *nested* within one another.
  - This is what the `cliques()` function in `igraph` does.

For example…



These are all identified as cliques:

{C, D, E}
{C, D, F}
{C, E, F}
{D, E, F}
{C, D, E, F}

Limitations of Clique Definition

- Relaxing the **maximal** criteria creates a lot of cliques.
- Also, the clique definition creates lots of overlapping cliques.
  - *Are these cohesive groups?*
- Another issue is that there is no internal differentiation since all positions are identical in the subgraph.
  - *No core members or leaders.*
- Finally, data collection may prevent cliques in graphs.
  - *Nominate 3 people…*

Generalizations of the Clique Concept

- Solutions?
    - Operationalize **cohesiveness** based on:
        - *Reachability*, such that nodes are not necessarily adjacent, but connecting paths are short
            - Example: *n*-cliques
        - *Degree*, such that nodes are adjacent to many other nodes, thereby reducing the vulnerability of the structure.
            - Examples: *k*-cores

n-Cliques

- An *n*-clique is a subgraph in which the largest geodesic distance between any two nodes is no greater than *n*.

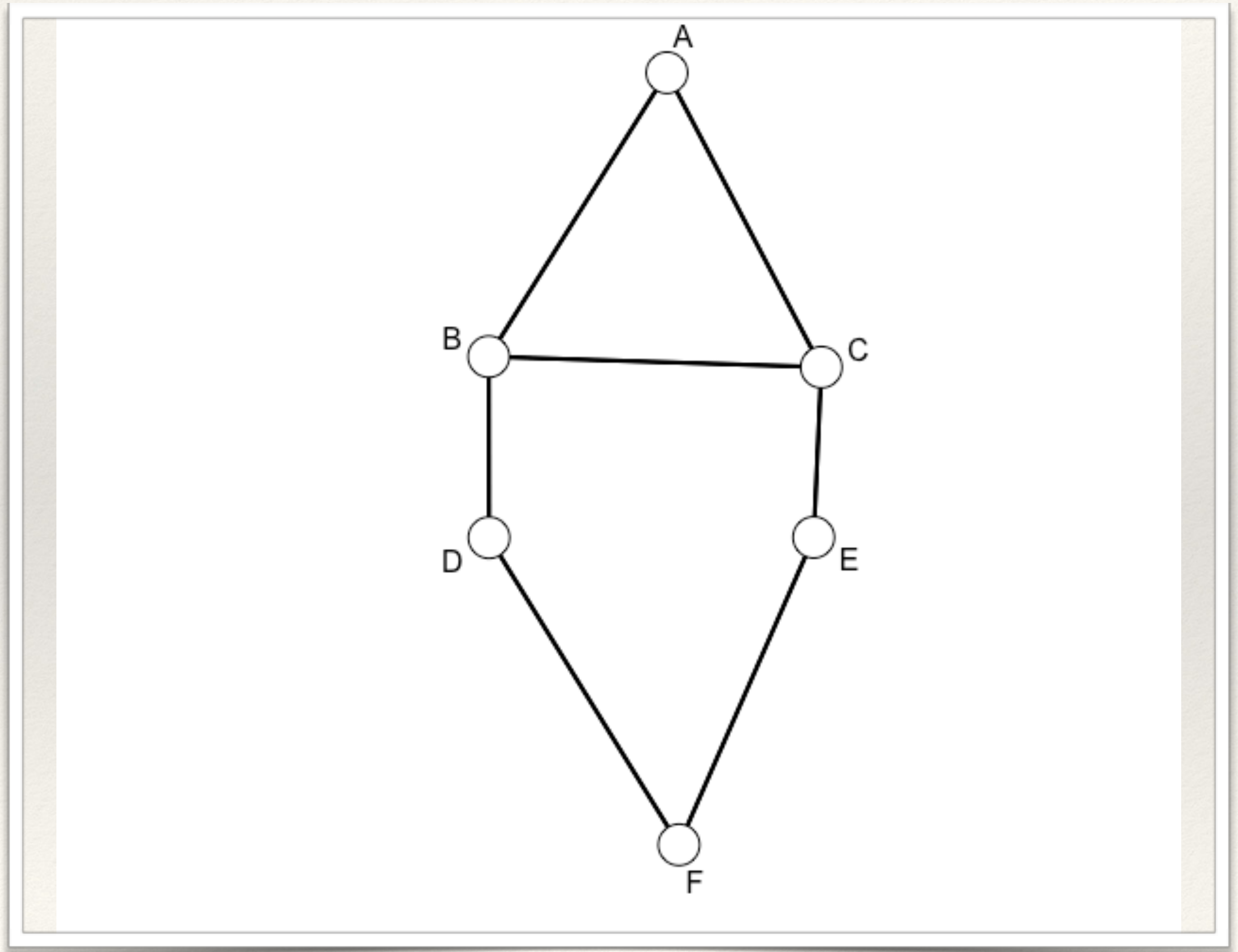
n-Cliques

$$d(i, j) \leq n \text{ for all } n_i, n_j \in N_s$$

Distance between i and j.

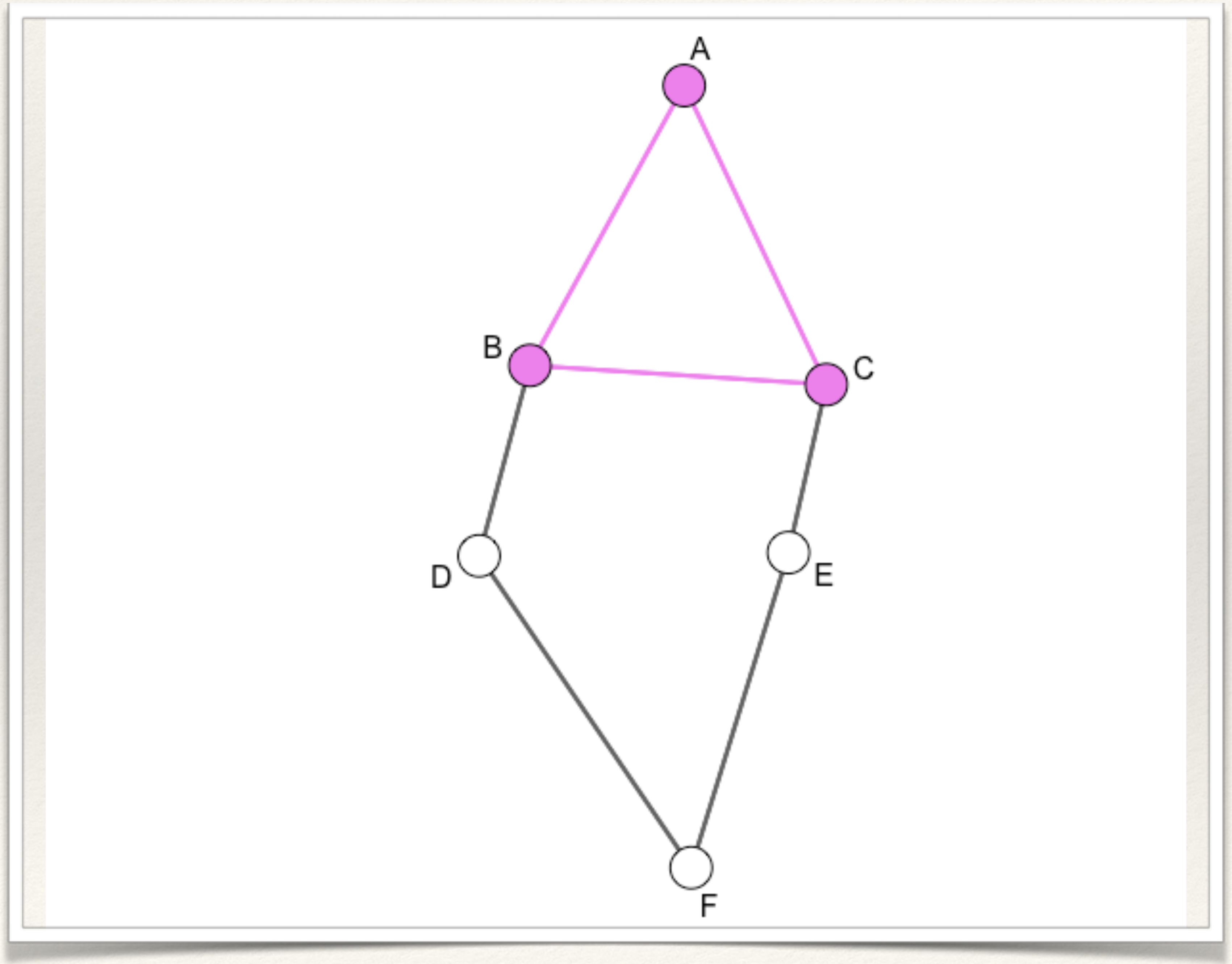
Is less than some arbitrary n for all nodes in the subgraph N_s

n-Cliques



Set $n = 1$, what is the *1*-clique?

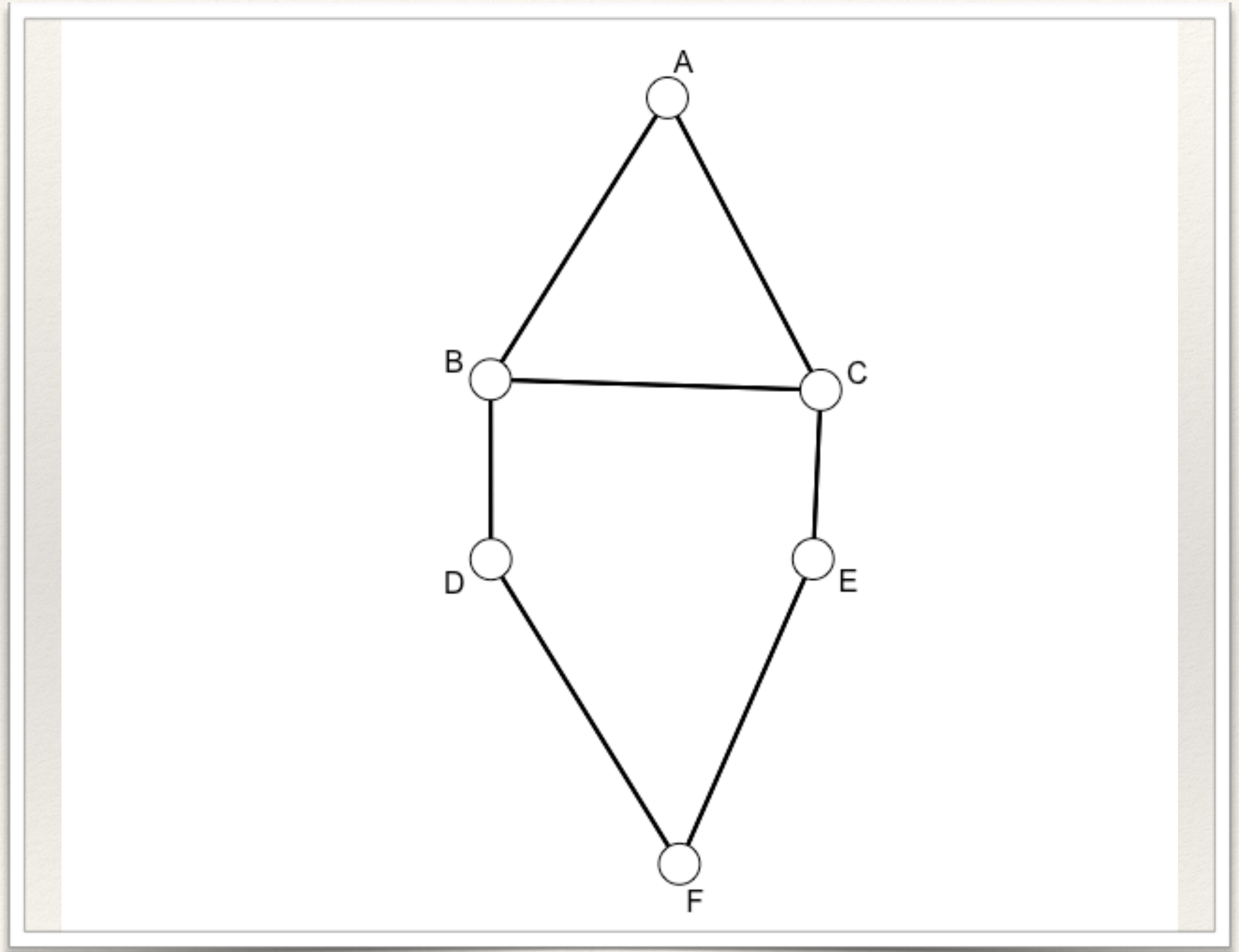
n-Cliques



Set $n = 1$, what is the *1*-clique?

{A, B, C}

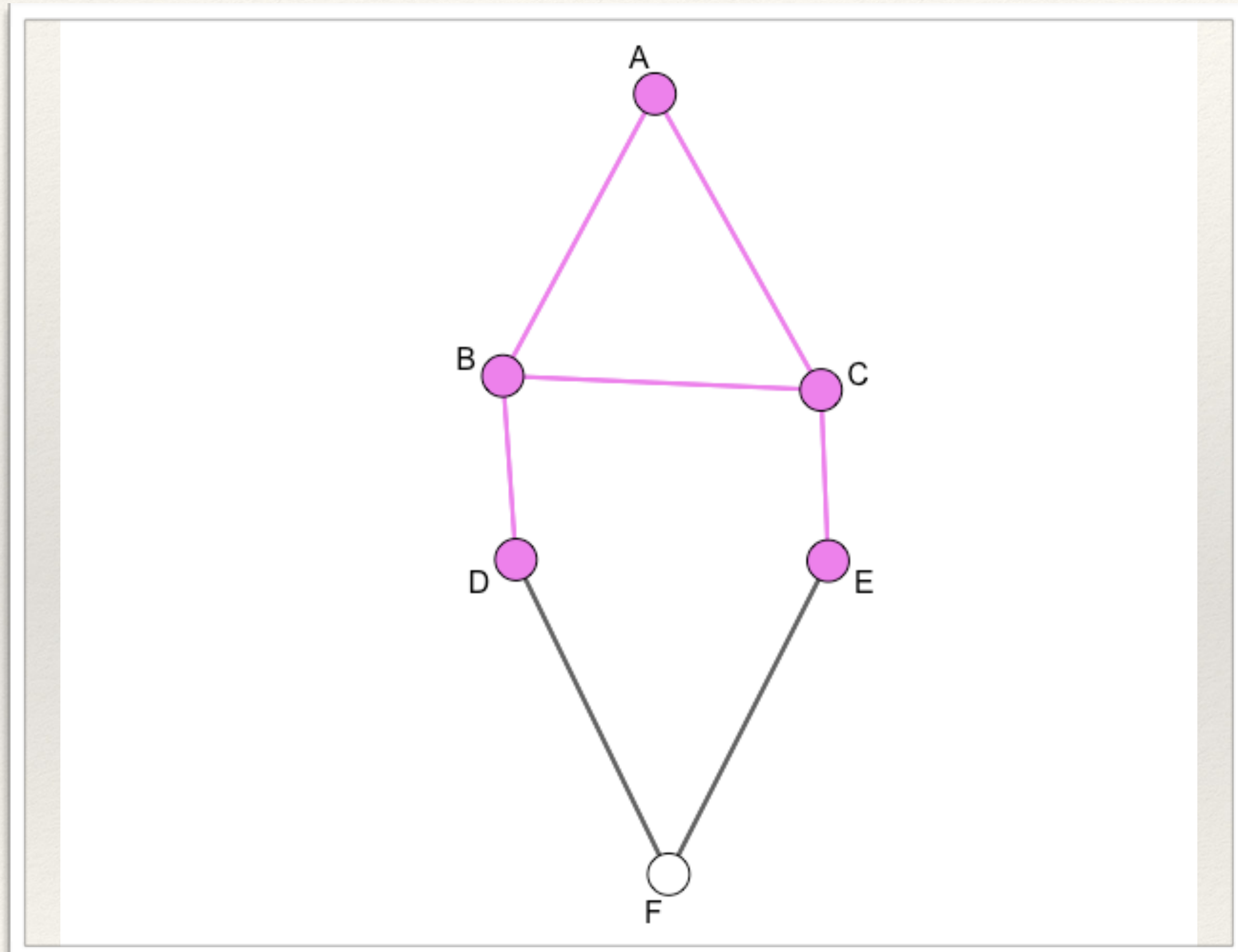
n-Cliques



Set $n = 2$, what is the 2-clique?

$$d(i, j) \leq 2 \text{ for all } n_i, n_j \in N_s$$

n-Cliques

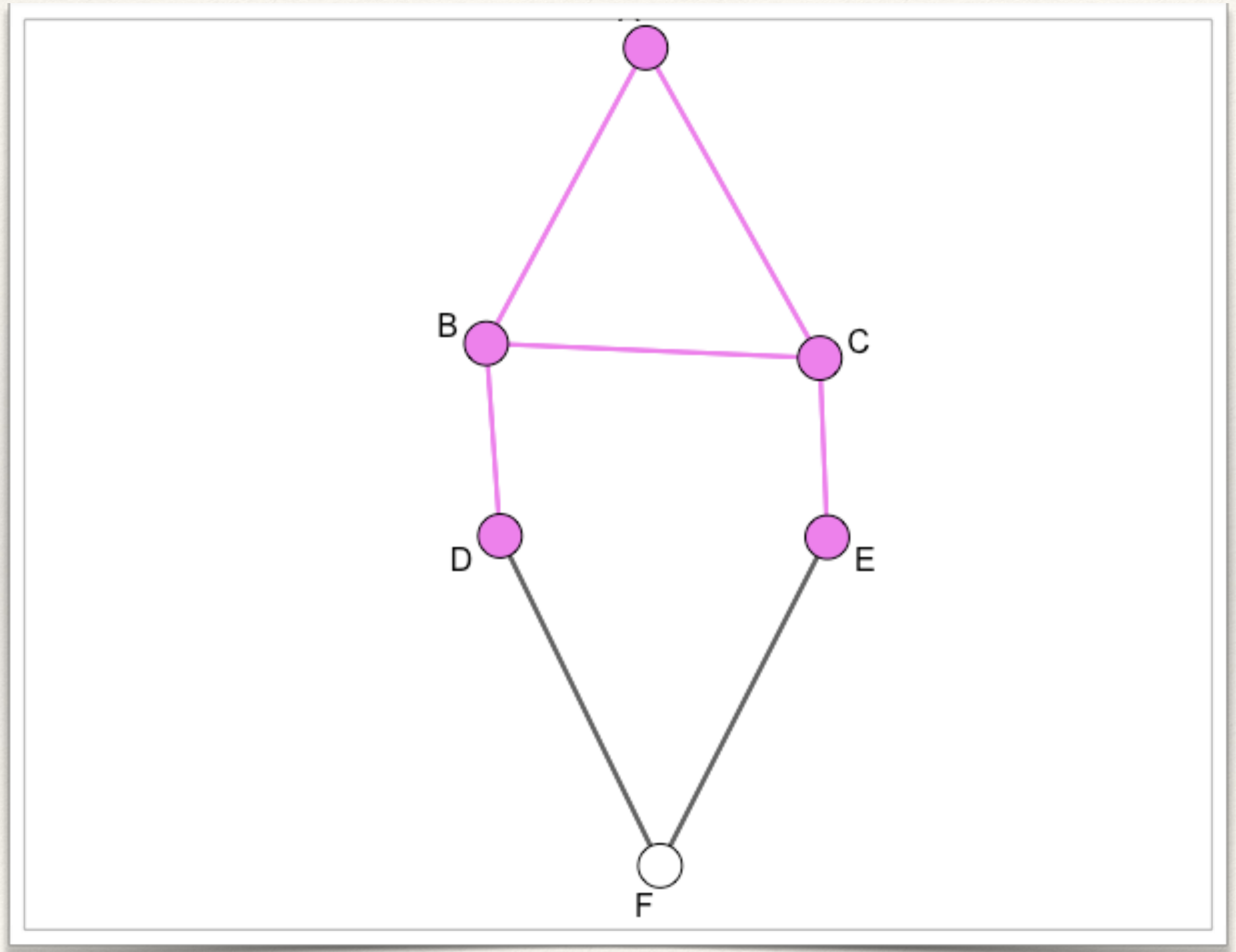


$$d(i, j) \leq 2 \text{ for all } n_i, n_j \in N_s$$

Set $n = 2$, what is the 2-clique?

$\{A, B, C, D, E\}$

n-Cliques



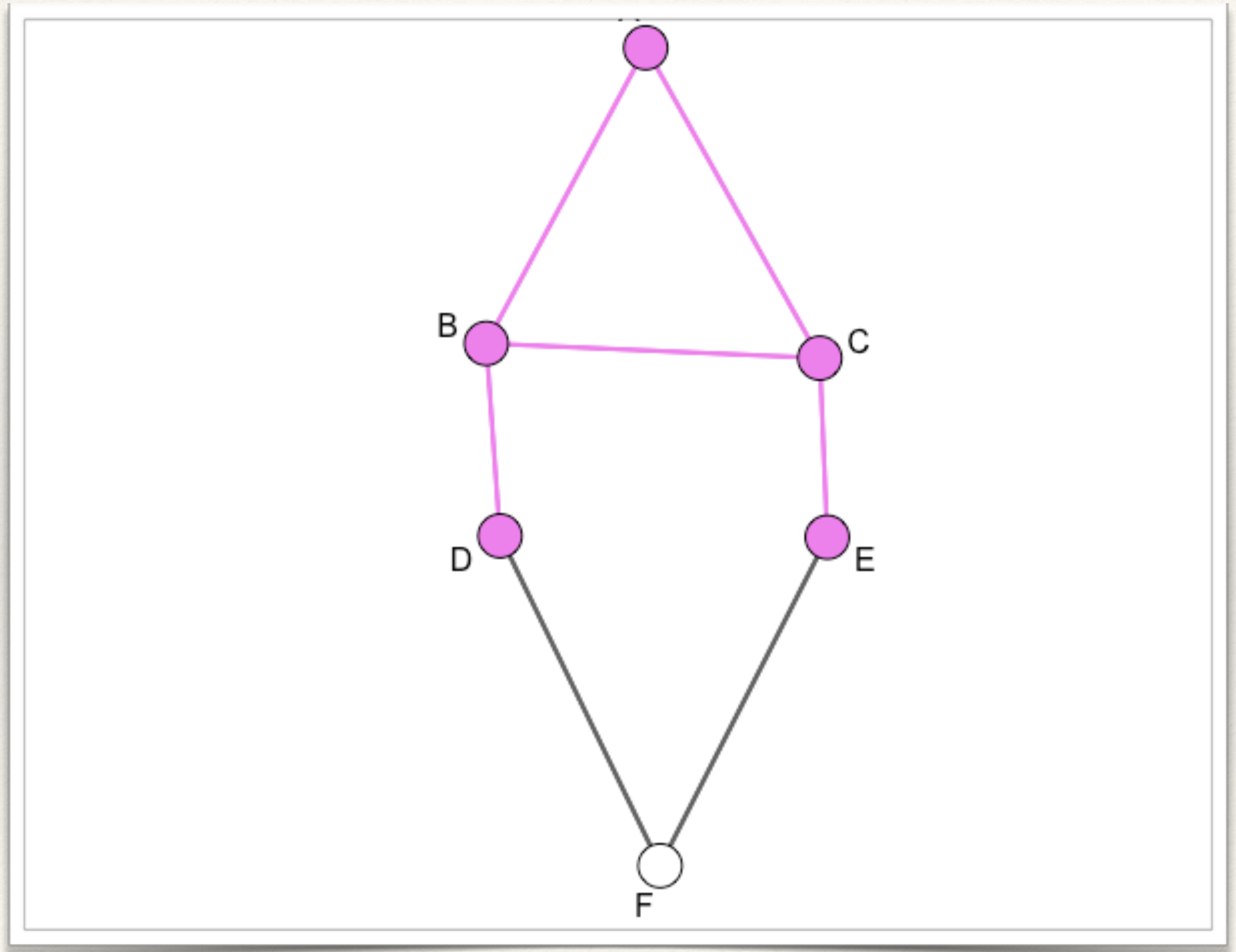
Set $n = 2$, what is the 2-clique?

$\{A, B, C, D, E\}$

The maximum geodesic for all nodes is 2.

$$d(i, j) \leq 2 \text{ for all } n_i, n_j \in N_s$$

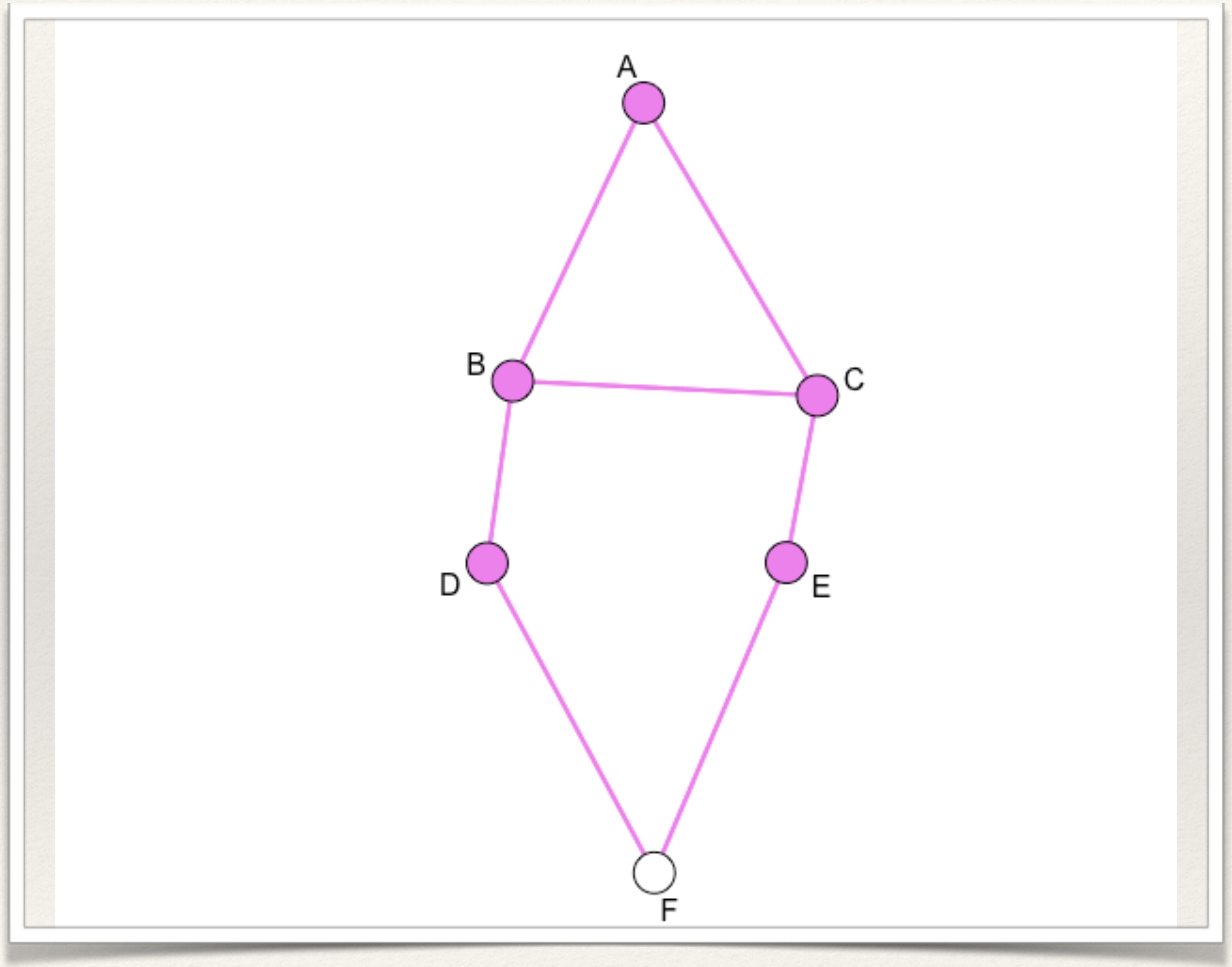
n-Cliques



Anything peculiar about this 2-clique, particularly regarding D and E?

$$d(i,j) \leq 2 \text{ for all } n_i, n_j \in N_s$$

n-Cliques



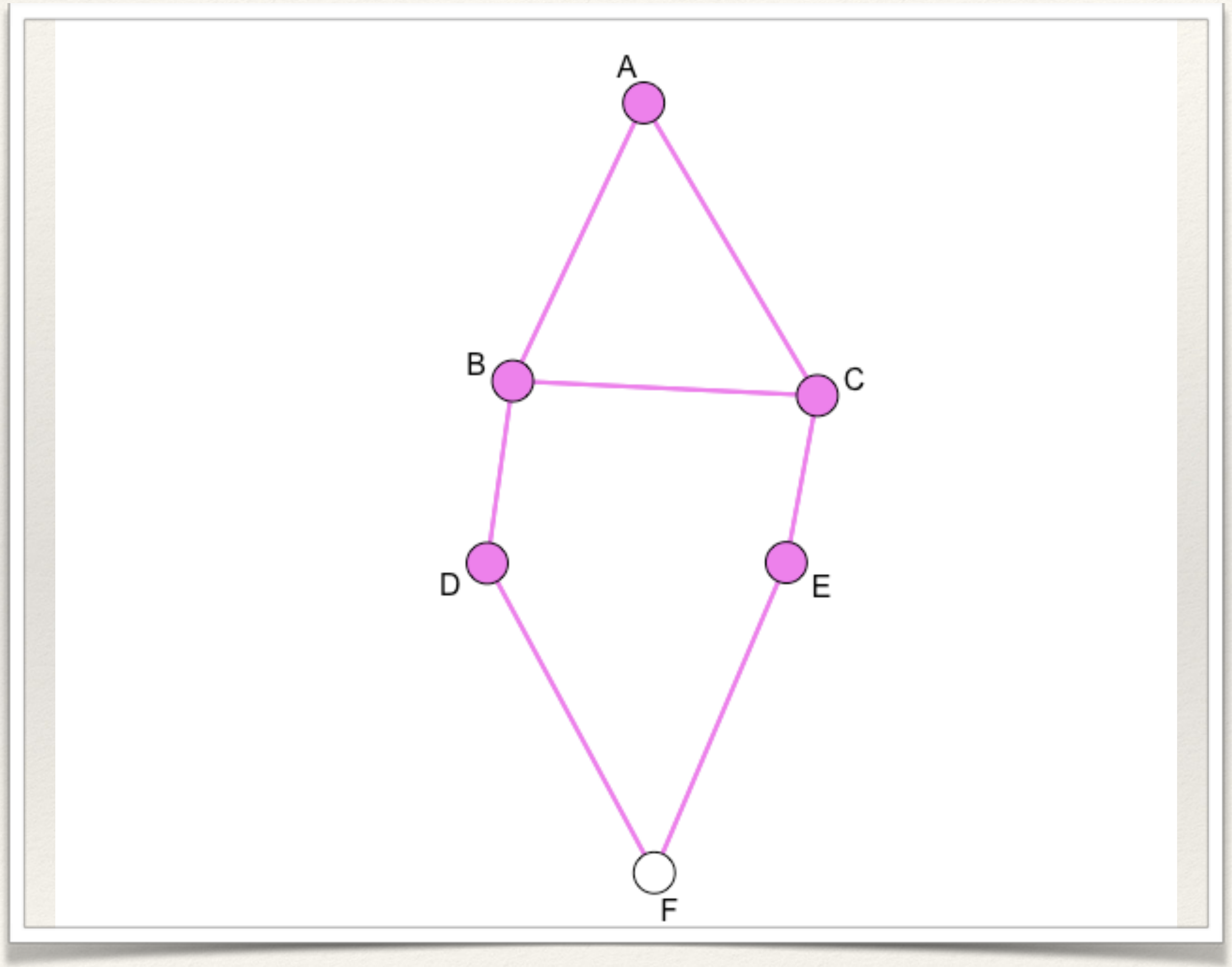
$$d(i, j) \leq 2 \text{ for all } n_i, n_j \in N_s$$

Anything peculiar about this 2-clique, particularly regarding D and E?

The geodesic between D and E is not D-B-C-E

It is D-F-E

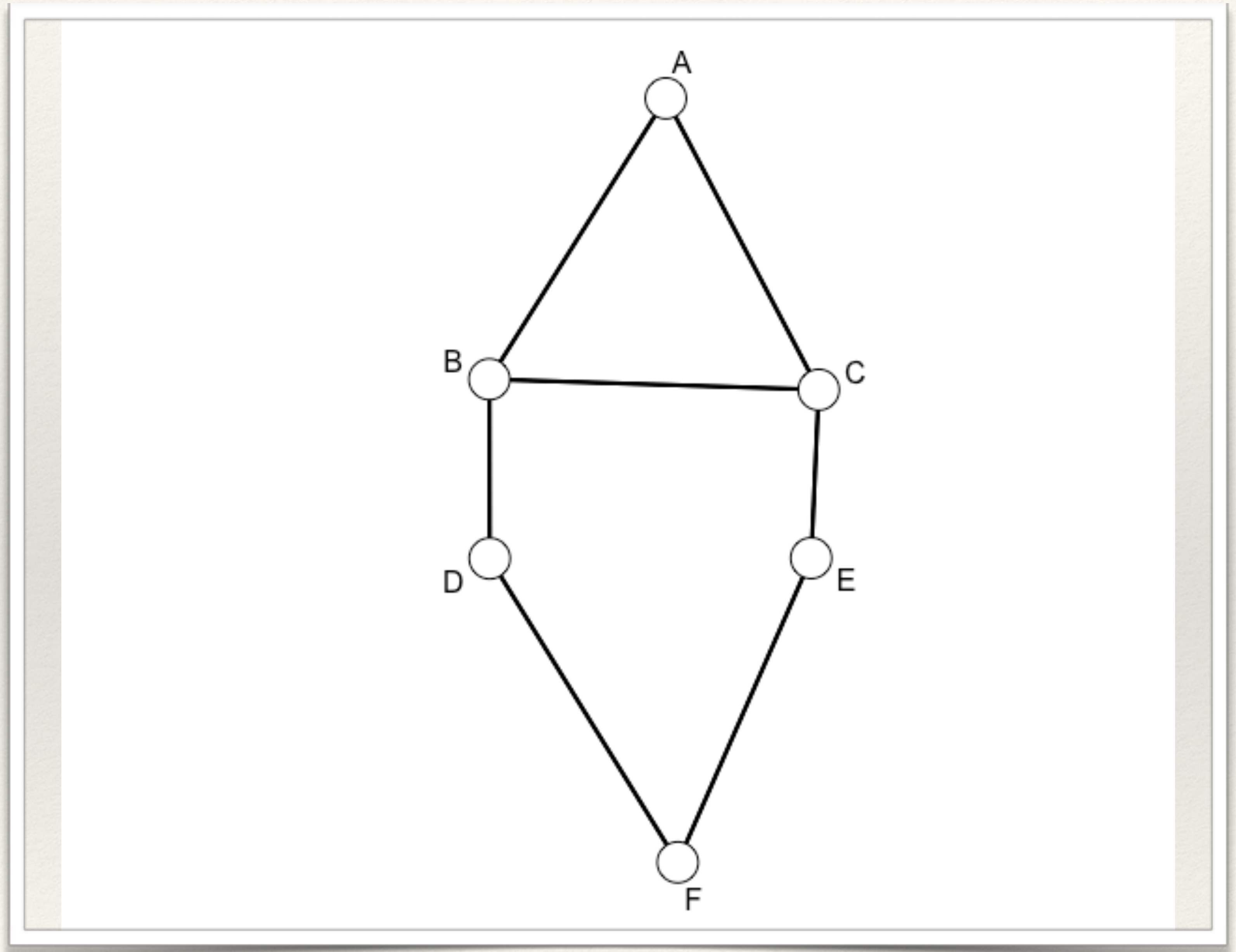
n-Cliques



So, an n-clique can include nodes that are not in the subgraph.

$$d(i, j) \leq 2 \text{ for all } n_i, n_j \in N_s$$

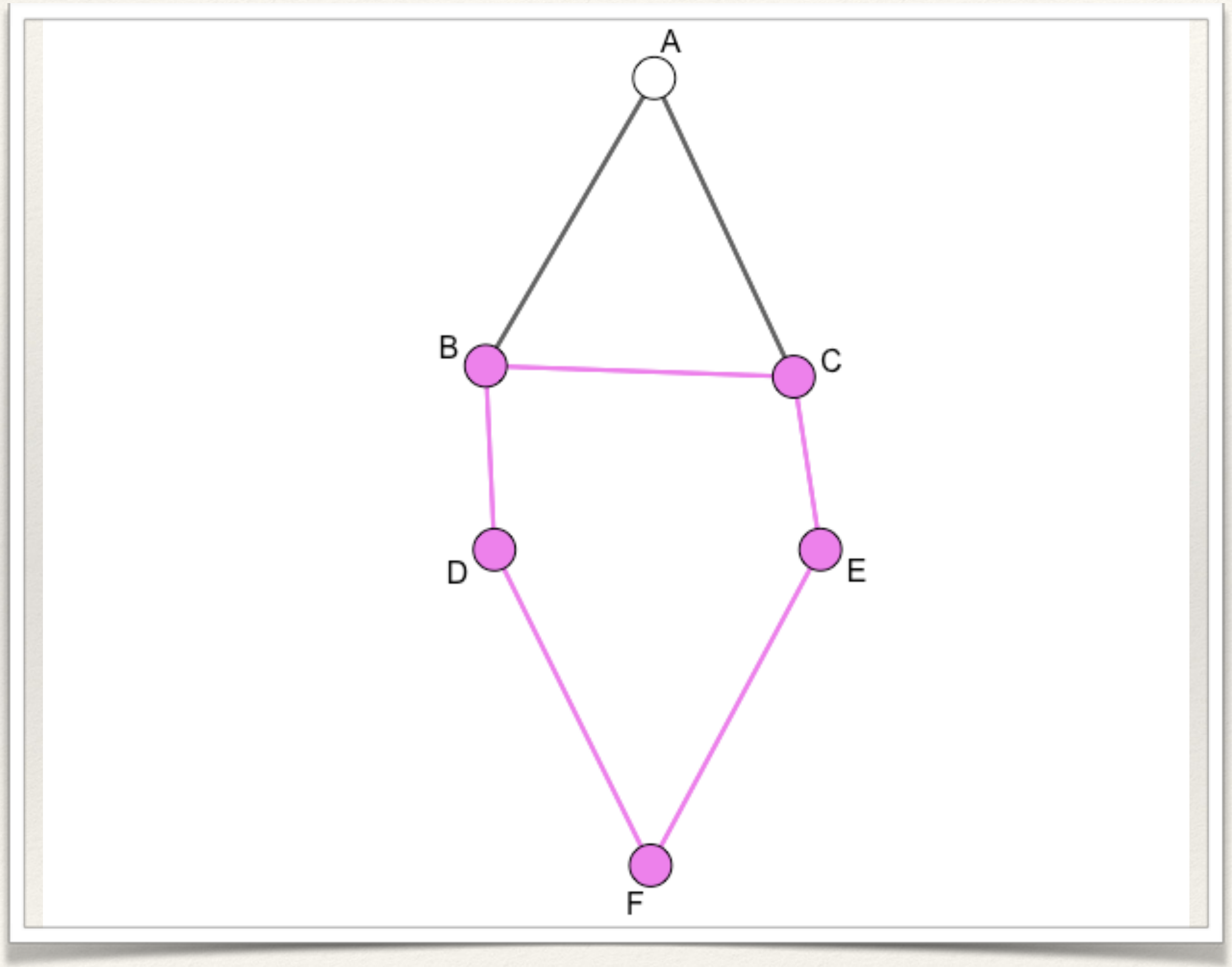
n-Cliques



$$d(i, j) \leq 2 \text{ for all } n_i, n_j \in N_s$$

Set $n = 2$, what is the other 2-clique?

n-Cliques

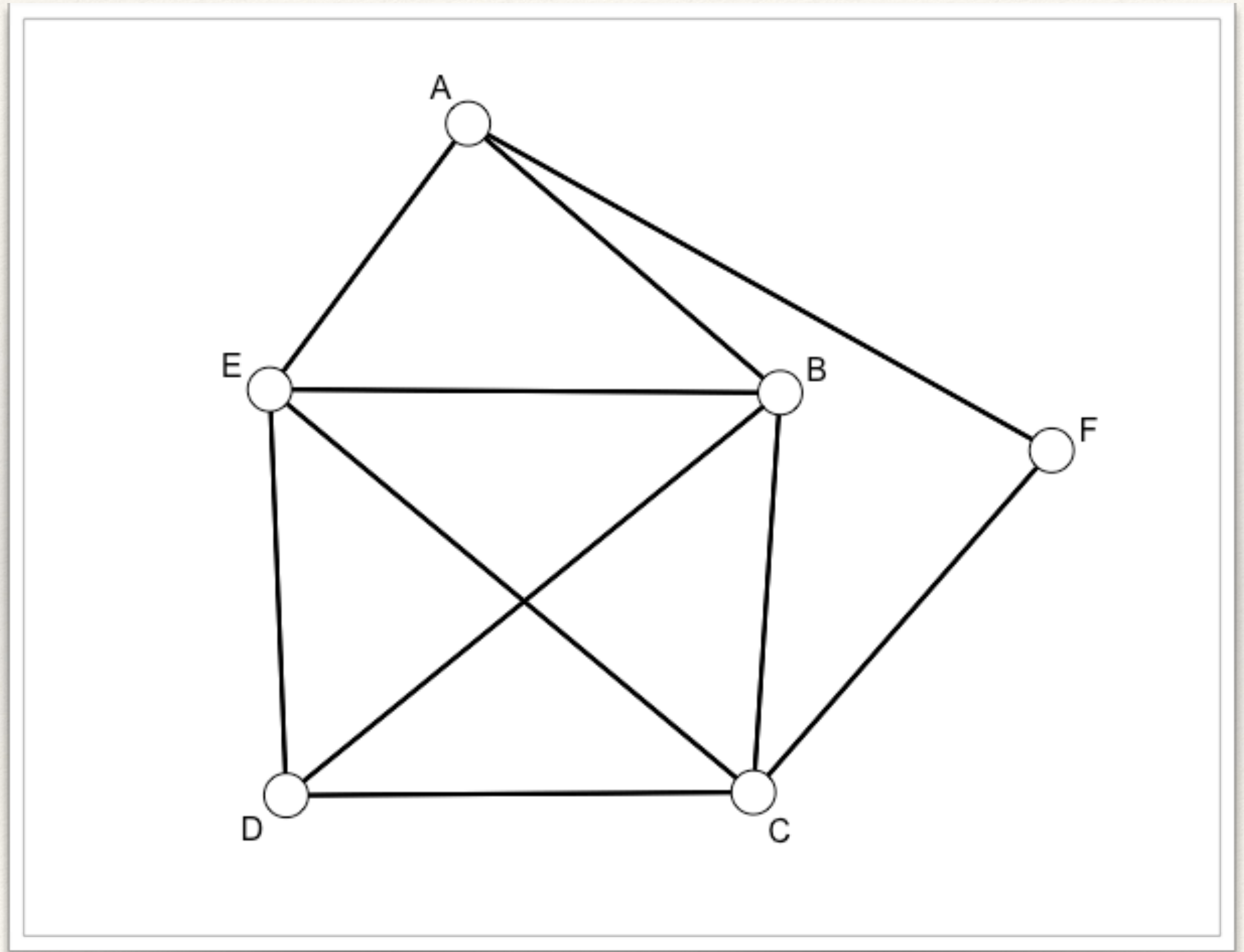


$$d(i, j) \leq 2 \text{ for all } n_i, n_j \in N_s$$

Set $n = 2$, what is the other 2-clique?

{*B*, *C*, *D*, *E*, *F*}

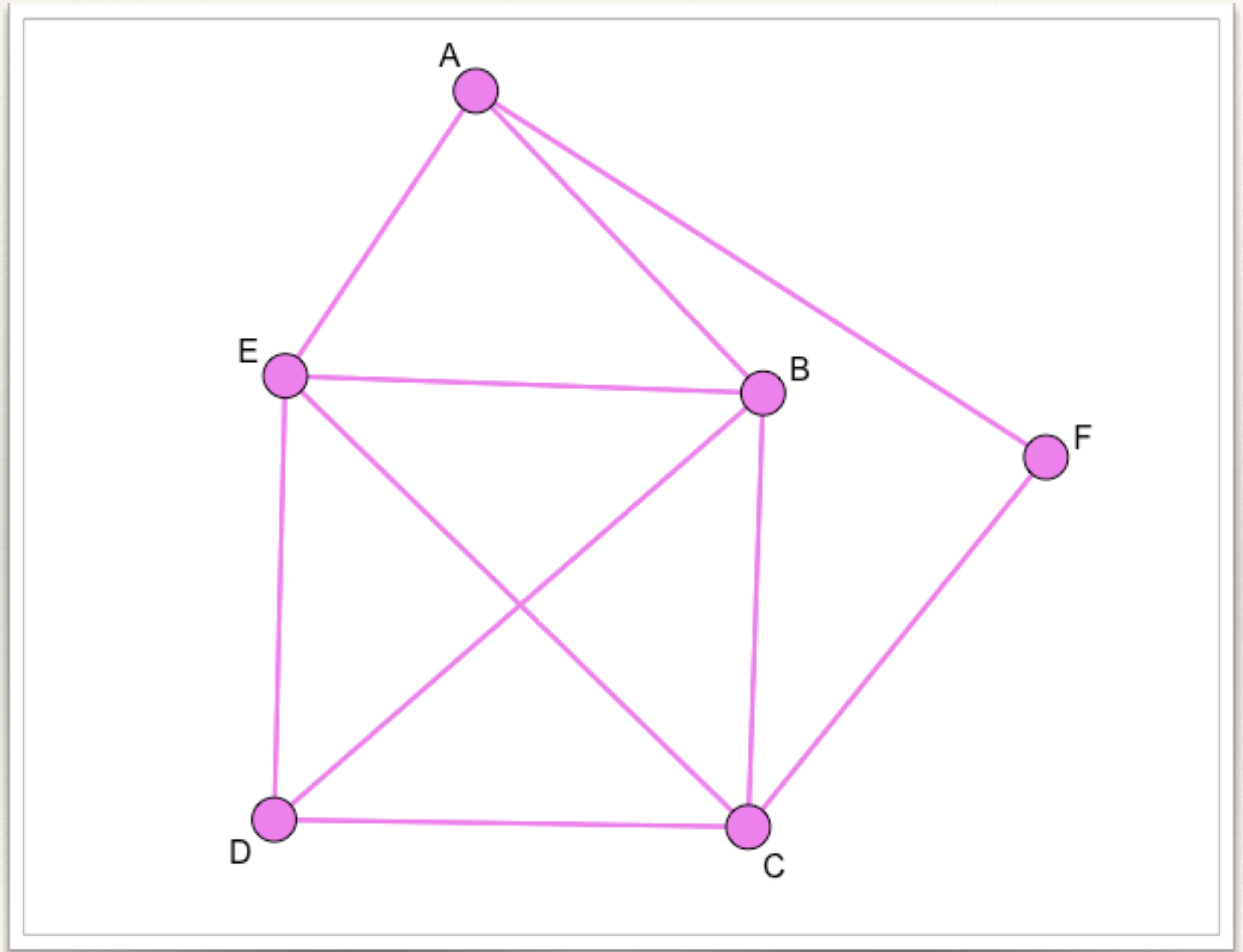
n-Cliques



Set $n = 2$, what is the 2-clique?

$$d(i, j) \leq 2 \text{ for all } n_i, n_j \in N_s$$

n-Cliques



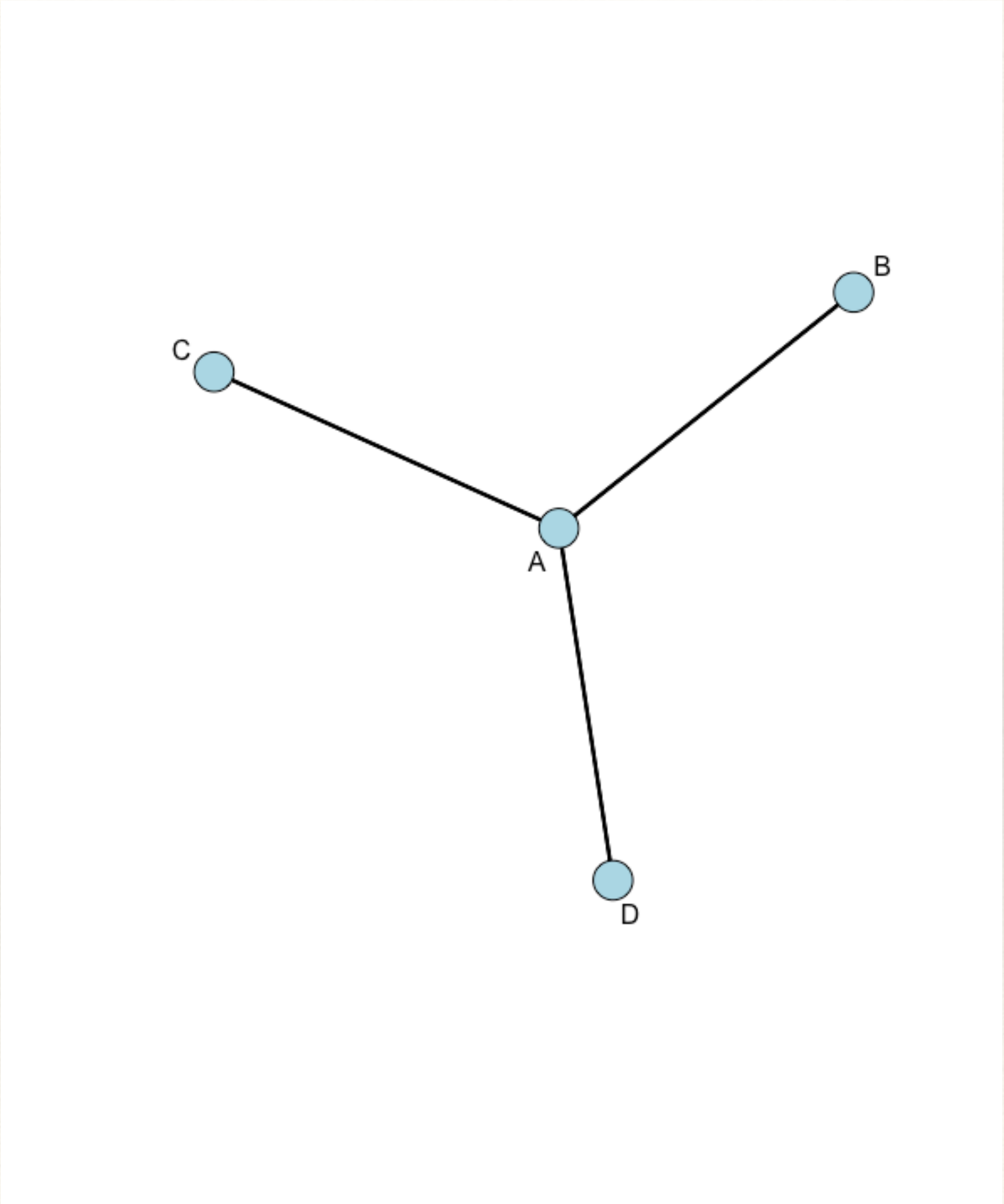
$$d(i, j) \leq 2 \text{ for all } n_i, n_j \in N_s$$

Set $n = 2$, what is the 2-clique?

{*A*, *B*, *C*, *D*, *E*, *F*}

n-Cliques

- For *n*-cliques, we could gradually increase the number to show greater reachability.
  - But, this would increase the vulnerability of the network.
    - The removal of "A" here, disconnects the graph.



k-core

- An alternative approach, that avoids vulnerability to disruption, is to use *degree*.
  - A *k*-core is a subgraph in which each node is adjacent to at least a minimum number, *k*, other nodes in the subgraph.
    - A nodes minimum degree within the subgraph will be at least *k*.

$$d_s(i) \geq k \text{ for all } n_i \in N_s$$

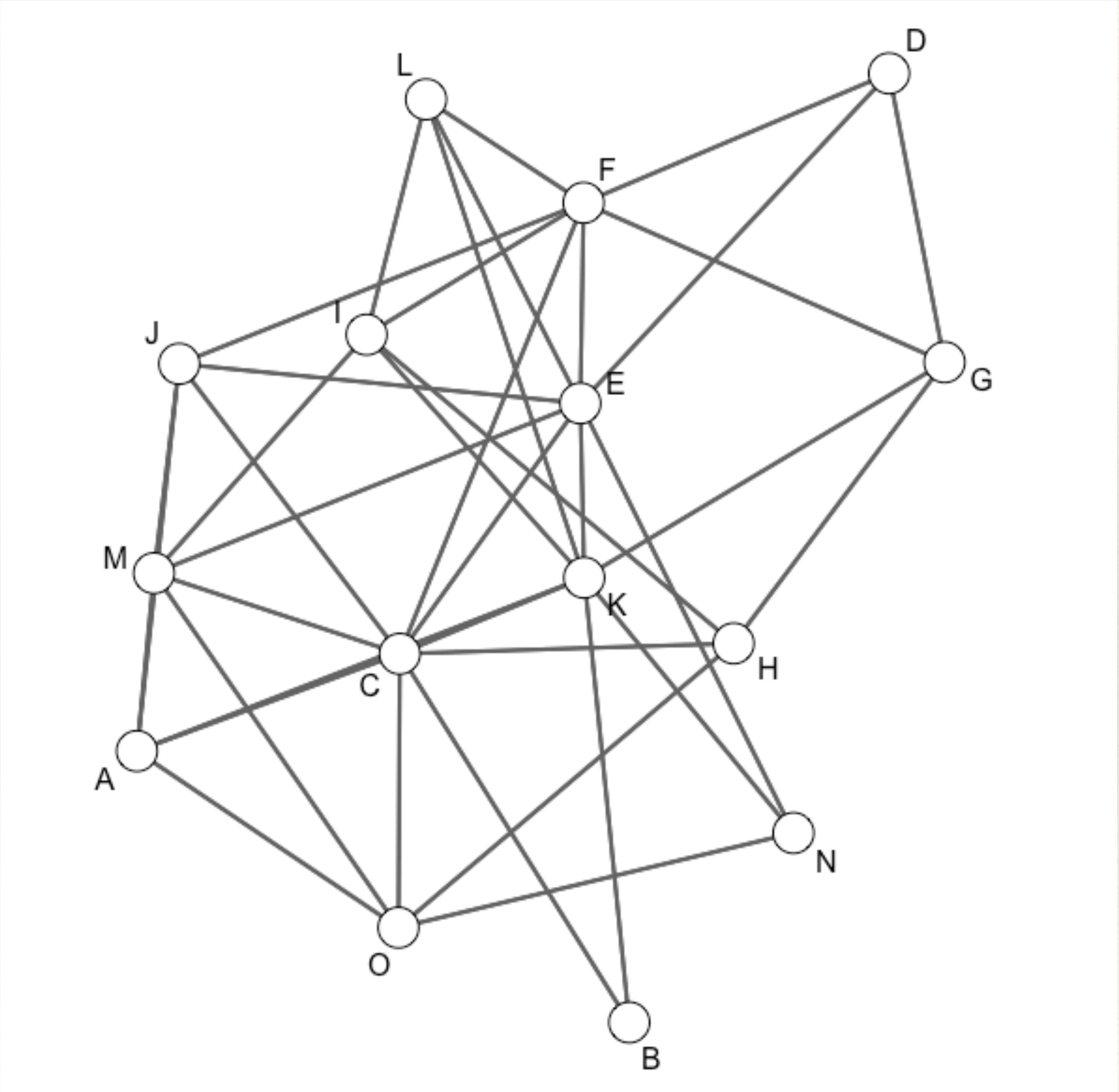
k-core

- k-cores will be nested (e.g., 1-core, then 2-core, …).
  - Like an onion!

k-core

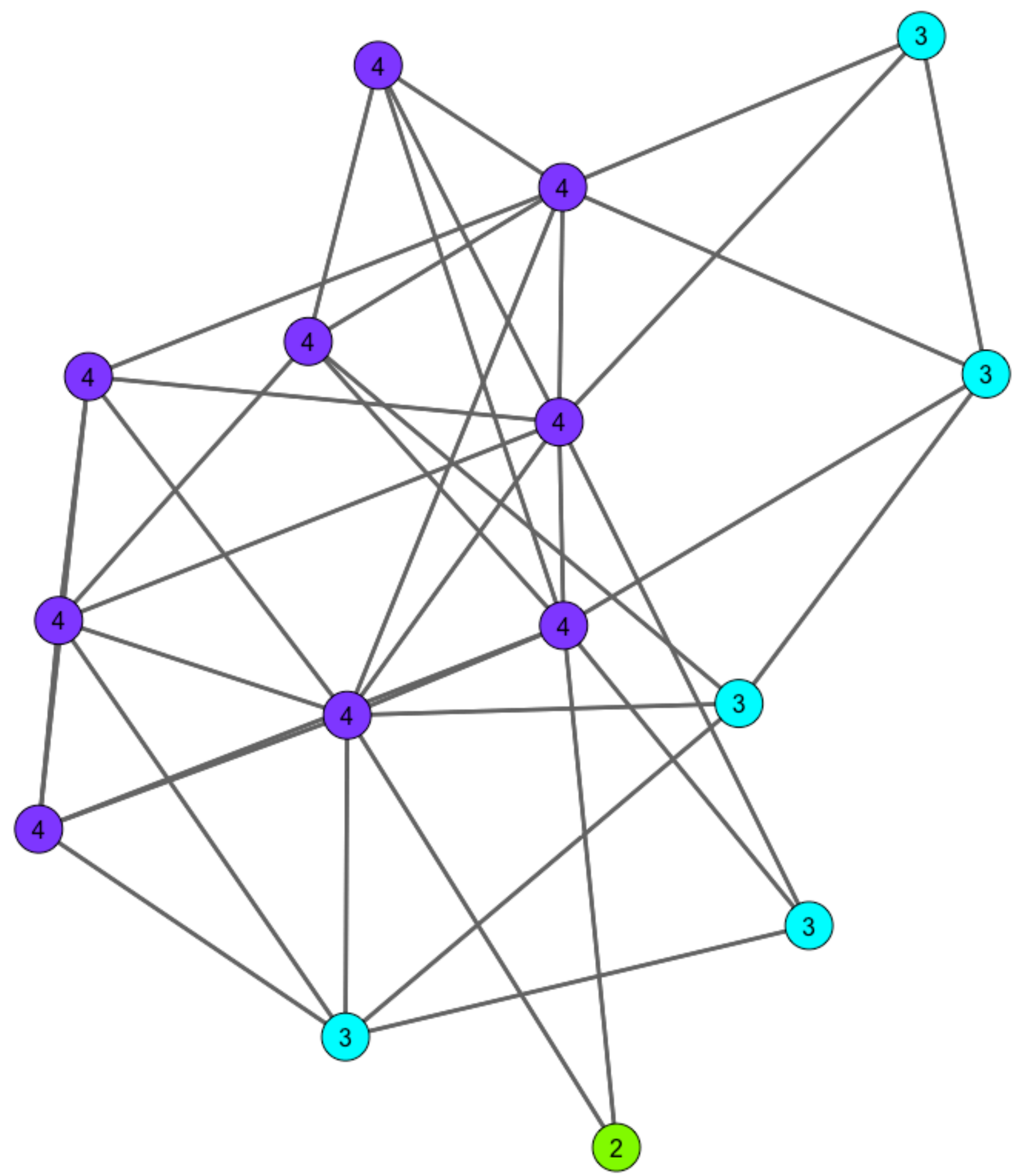
- k-cores will be nested (e.g., 1-core, then 2-core, …).
  - Like an onion!

k-core example

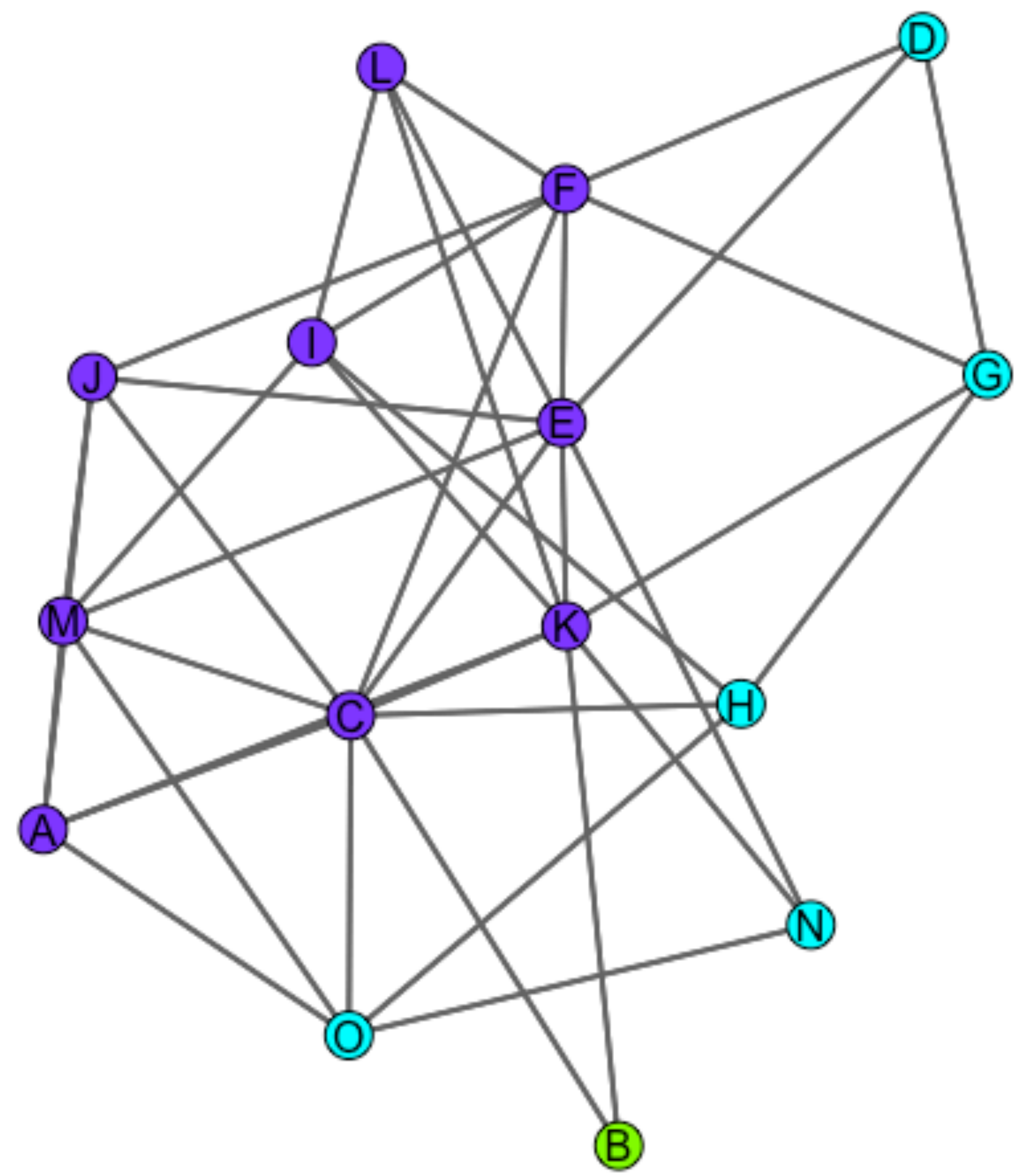


First, let's examine the *k*-core set.

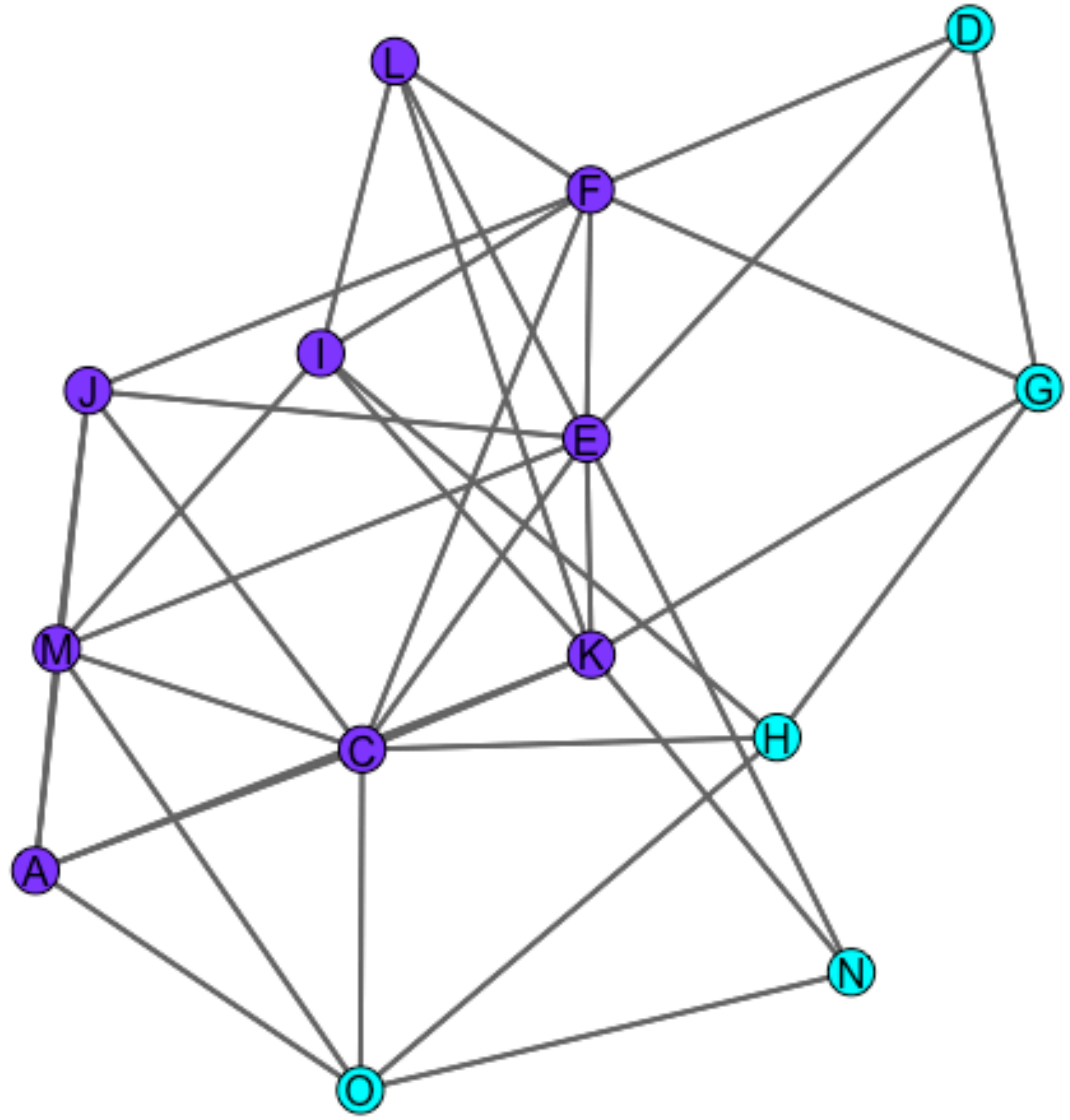
Plot of All k-cores



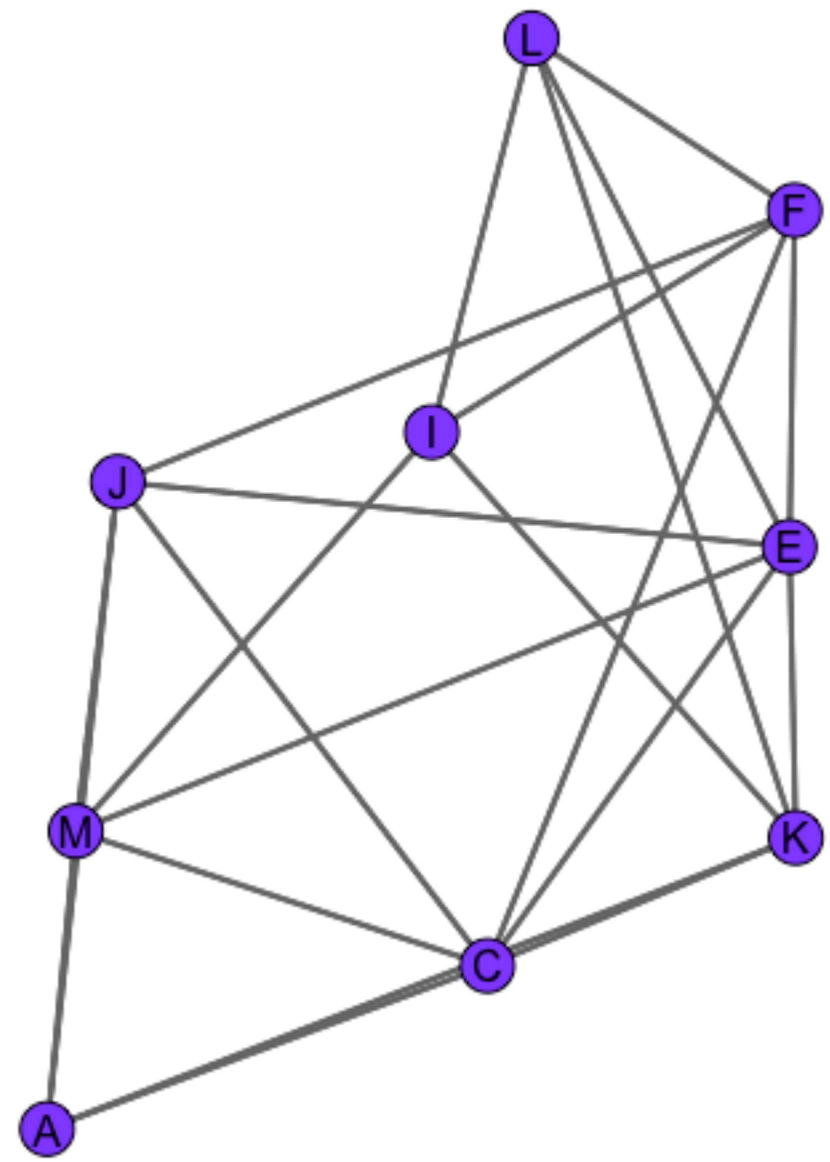
All 2:4-cores



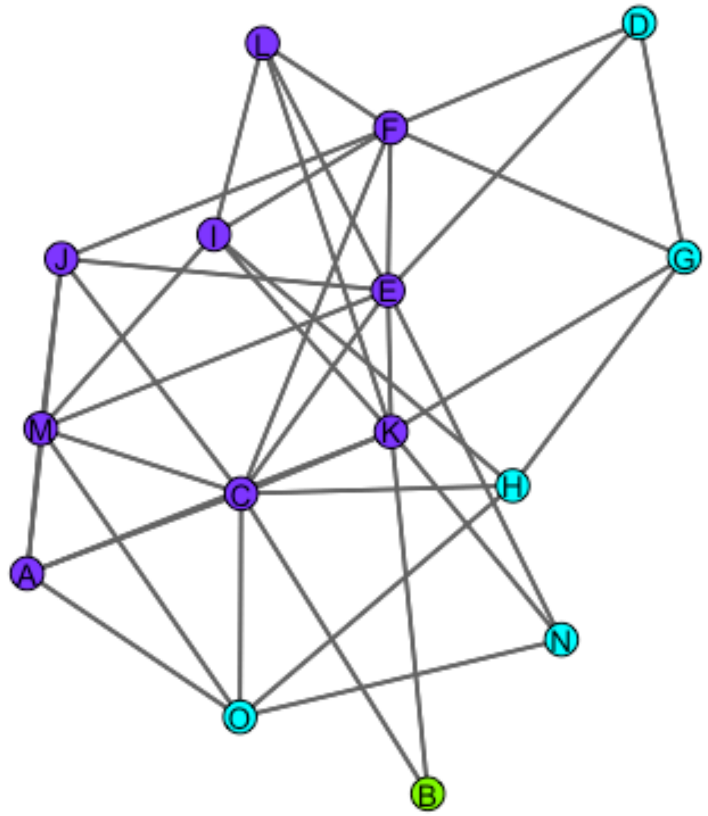
All 3:4-cores



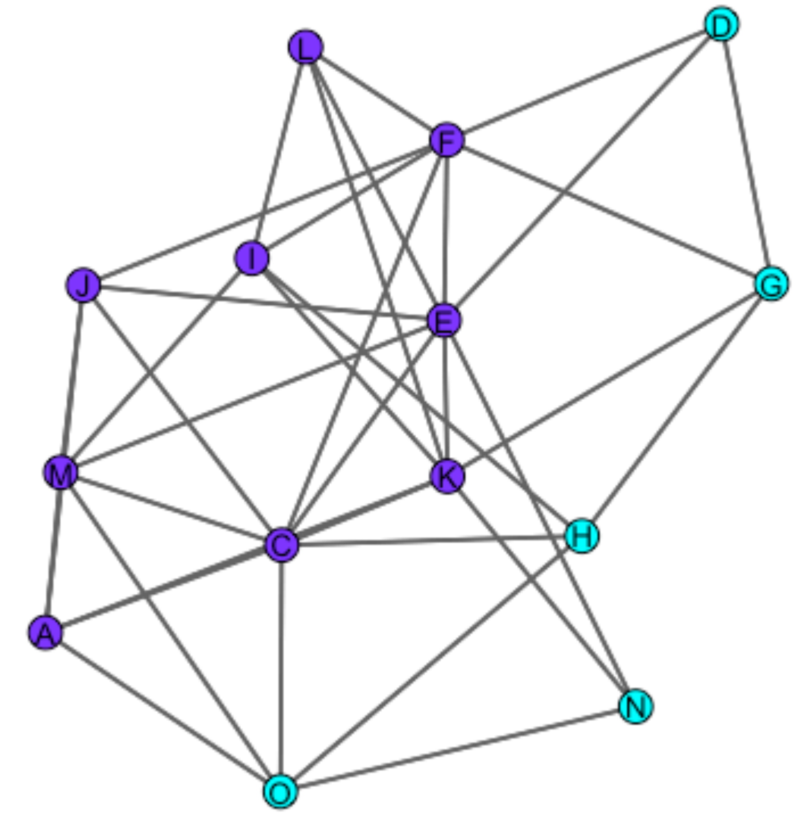
All 4-cores



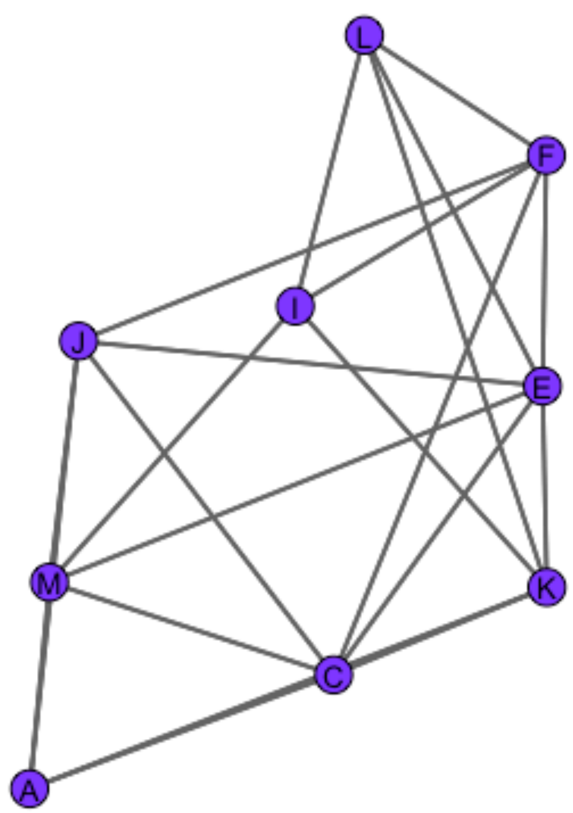
All 2:4-cores



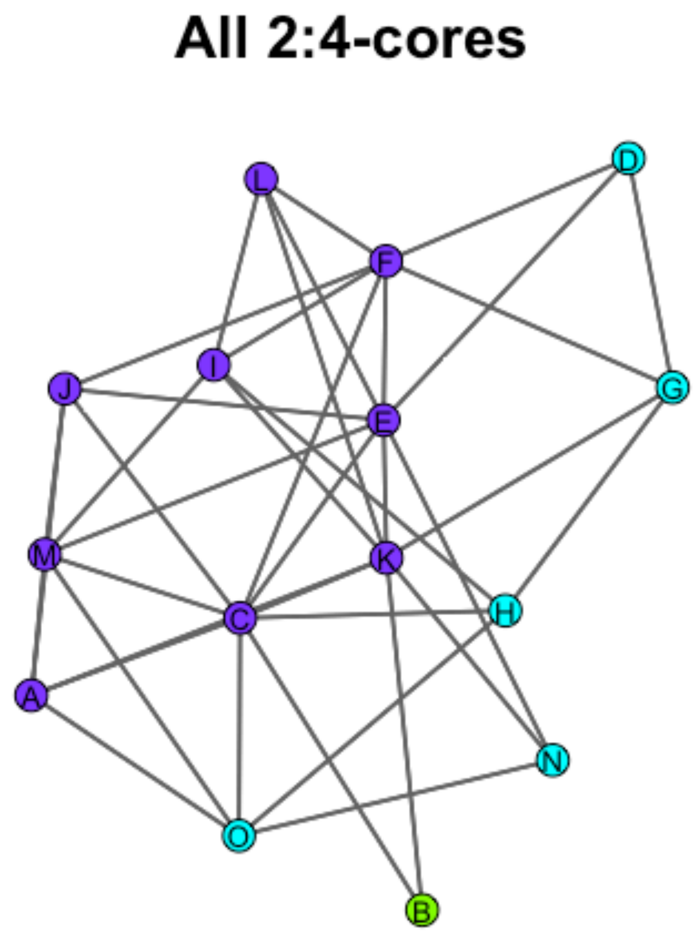
All 3:4-cores

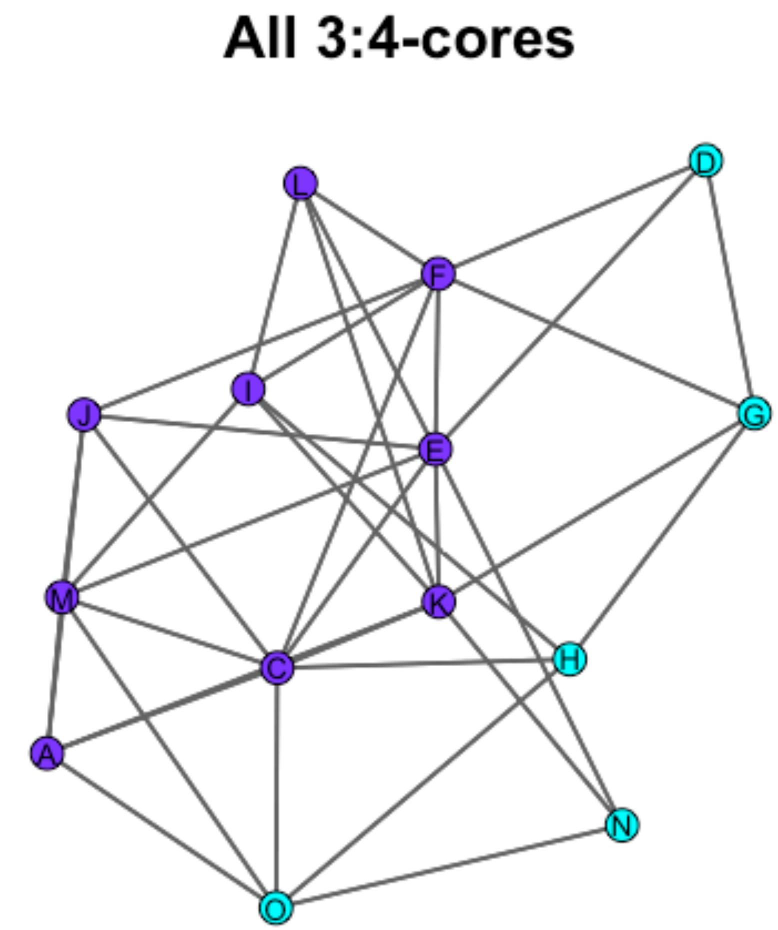


All 4-cores

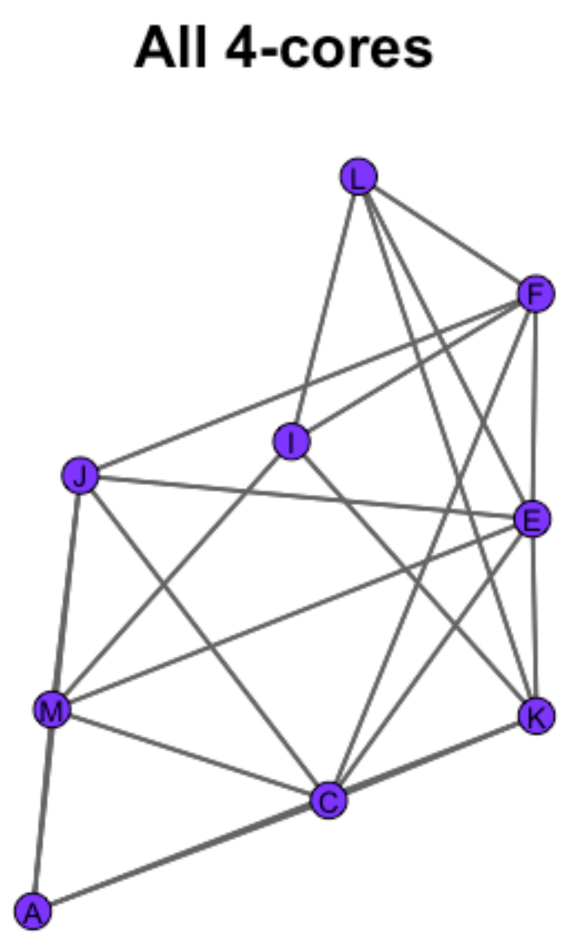


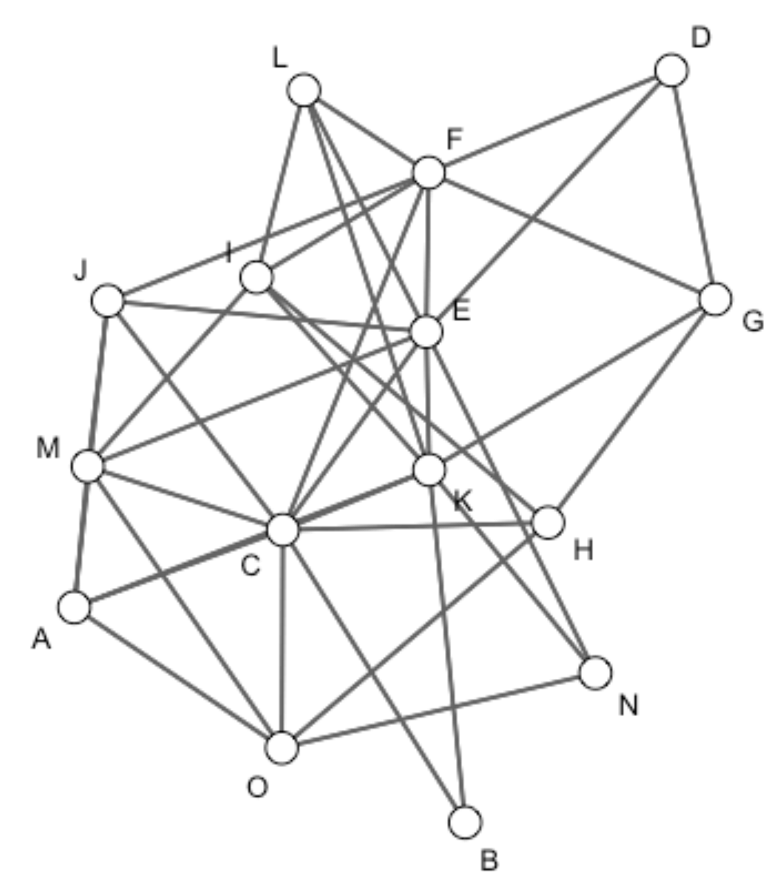
All of the cores, gradually peeled away.





You get a sense of the "layers"

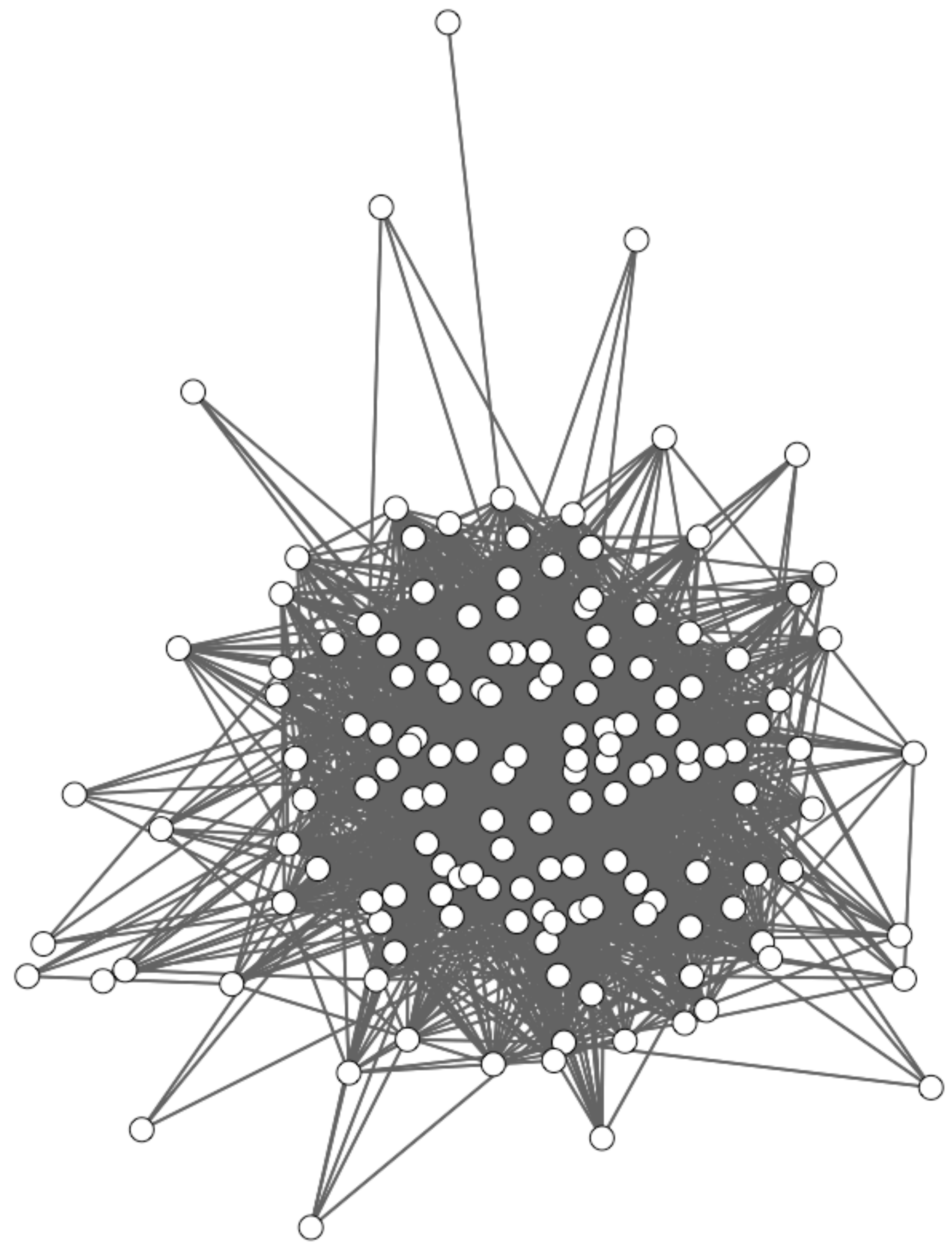


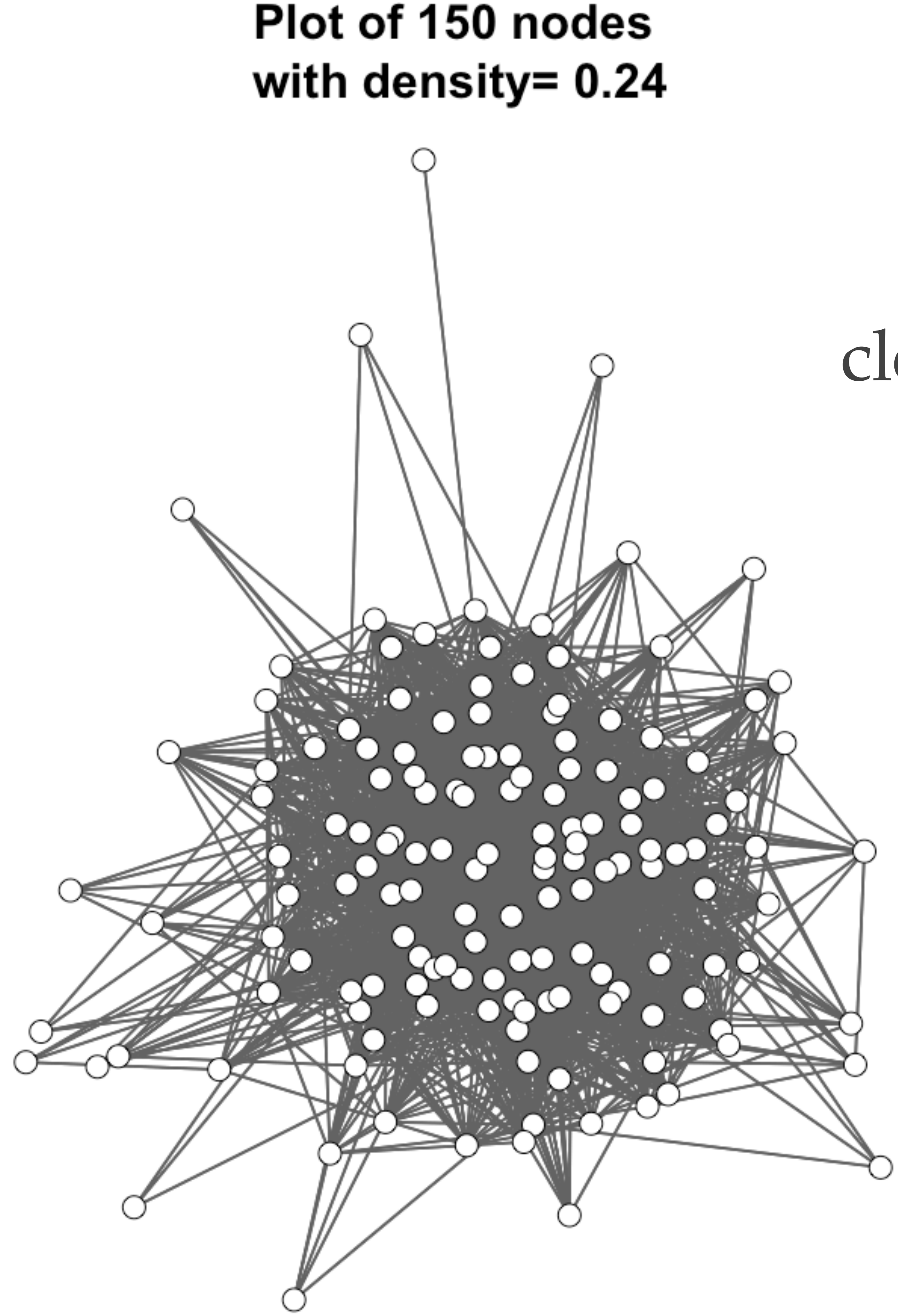


k-core

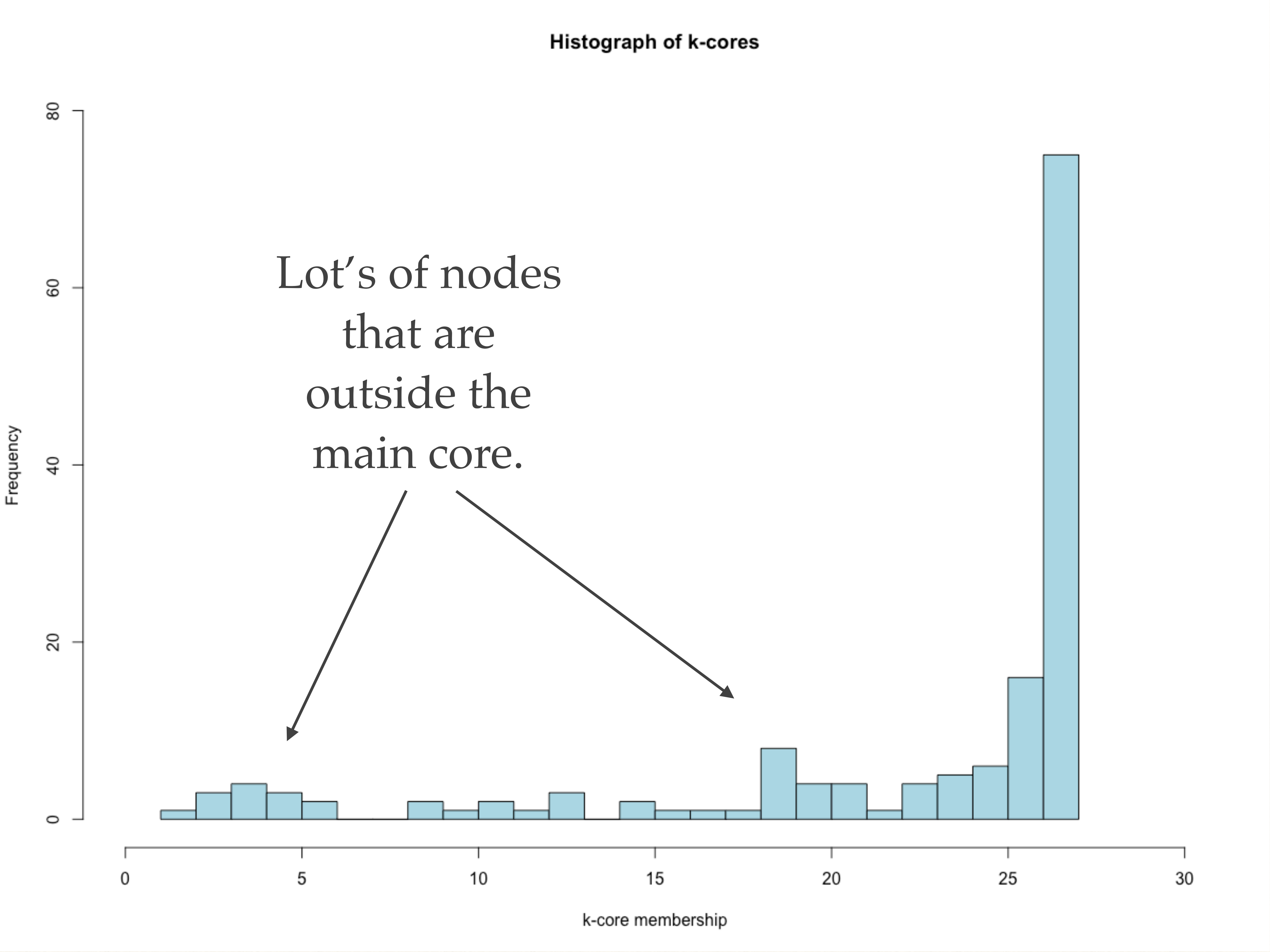
- The cohesive group is harder to see in more complex structures.
  - Examining the distribution of the *k*-cores gives a sense of <u>what</u> group is the most cohesive as well as how <u>embedded</u> that group is in the graph.

Plot of 150 nodes
with density= 0.24

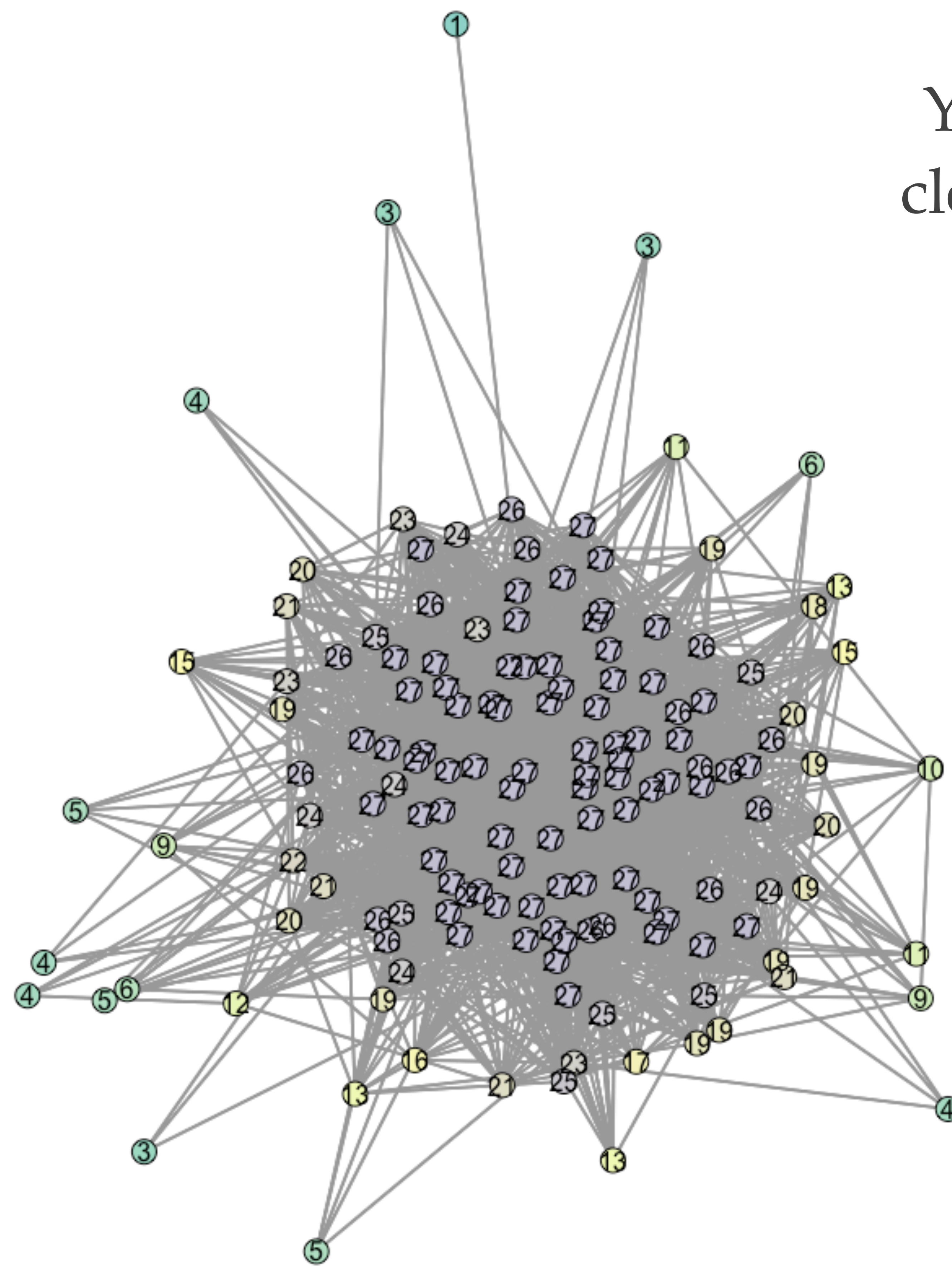




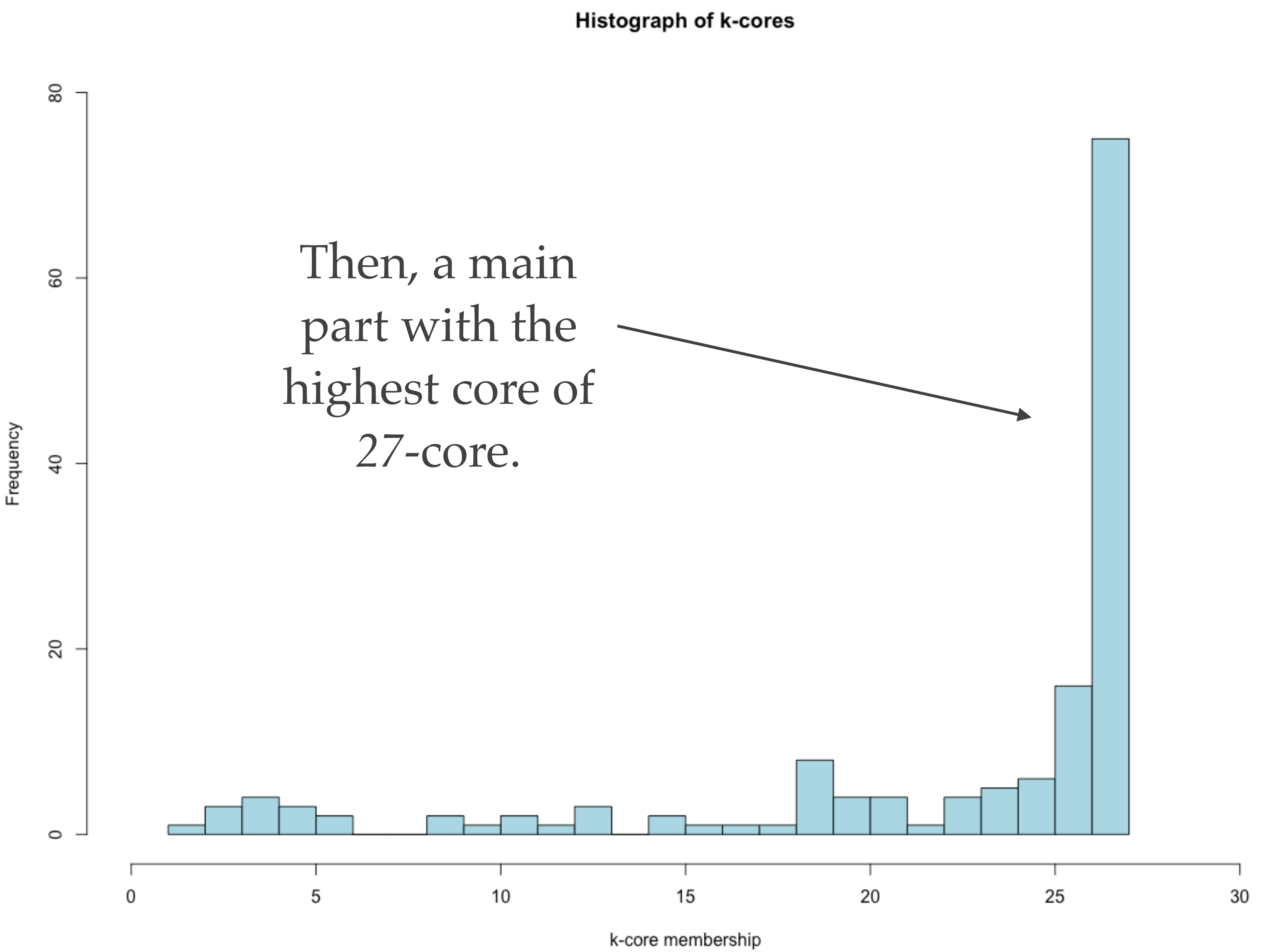
You can clearly see the cohesive groups…


Histograph of k-cores
Frequency
0
20
40
60
80
k-core membership
0
5
10
15
20
25
30
Lot’s of nodes
that are
outside the
main core.

Plot of All k-cores

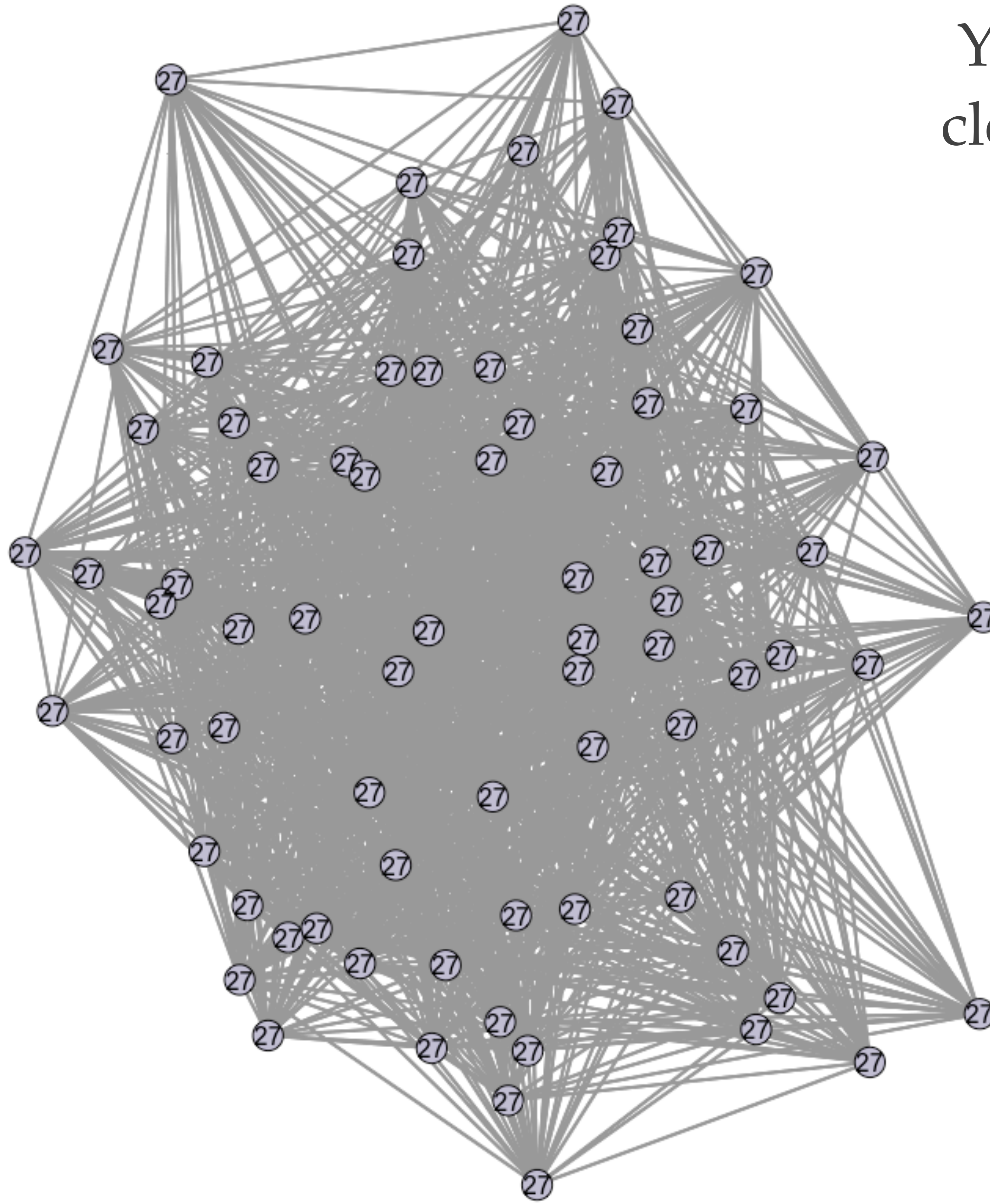


You can *now* clearly see the cohesive groups…



Then, a main part with the highest core of 27-core.

Plot of All k-cores



You can ***now*** clearly see the cohesive group…

What about external ties?

- So far, we have focused on cohesiveness by defining ties <u>within</u> the group.
  - Other approaches look at external ties comparisons.
    - **Community Detection**
      - Algorithms designed to maximize between group ties distance and minimize within group distance.
    - **Modularity**
      - Measure of how much clustering there is on some attribute.
        - Can be used to figure out how well a community detection algorithm is working.

Directed Networks

Directed Graphs

- Subgroup identification is based on **reciprocated** ties:
  - Requires that both dyad members report directed relations,
  - Then use same criteria for undirected graphs, since the symmetric directed graph is the same as an undirected graph.

Relaxing reciprocity constraint

- If we relax this constraint, we can identify four kinds of dyads of increasingly strict connectivity:
  - **Weakly n-connected**: *i* and *j* are joined by a semipath of length $\leq n$.
  - **Unilaterally n-connected**: a path of length $\leq n$ from *i* to *j* or from j to *i*.
  - **Strongly n-connected**: *i* and *j* are connected by two reciprocal paths of length $\leq n$, where the paths may have different intermediary nodes.
  - **Recursively n-connected**: *i* and *j* use same intermediary nodes and lines in reverse order as the path from j to *i*.
- The result is that we have 4 types of n-cliques: weakly, unilaterally, strongly, and recursively.

Learning Goals

- Examine conceptualization of *cohesion*.
- Understand conceptual definitions of cohesion.
- Understand approaches to operationalizing cohesion.

Questions?