Plan of attack
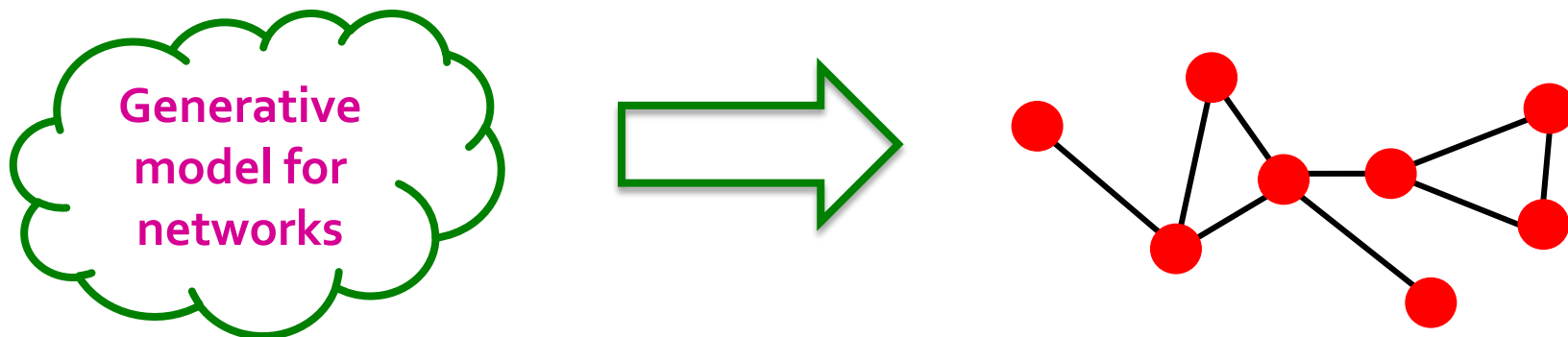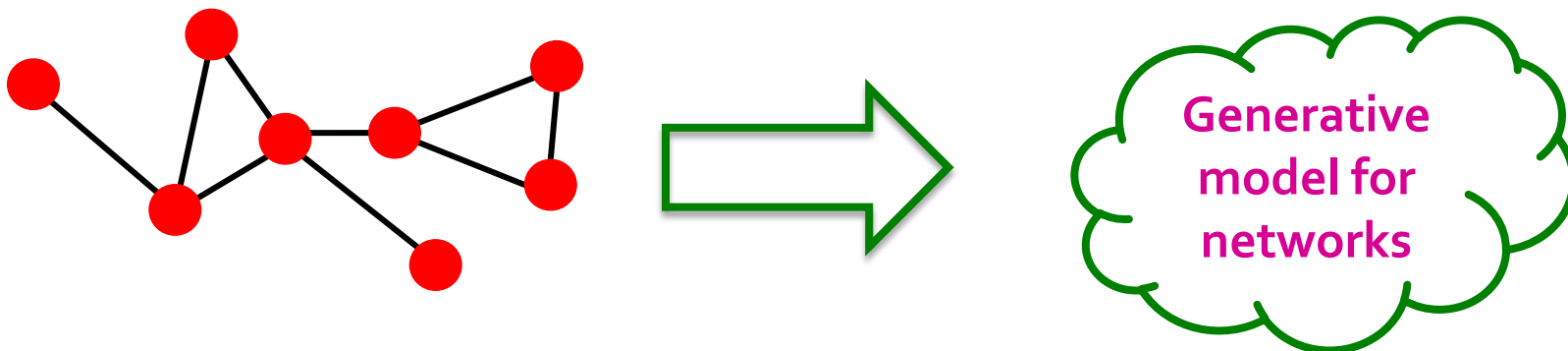
- **1) Given a model, we generate the network:**
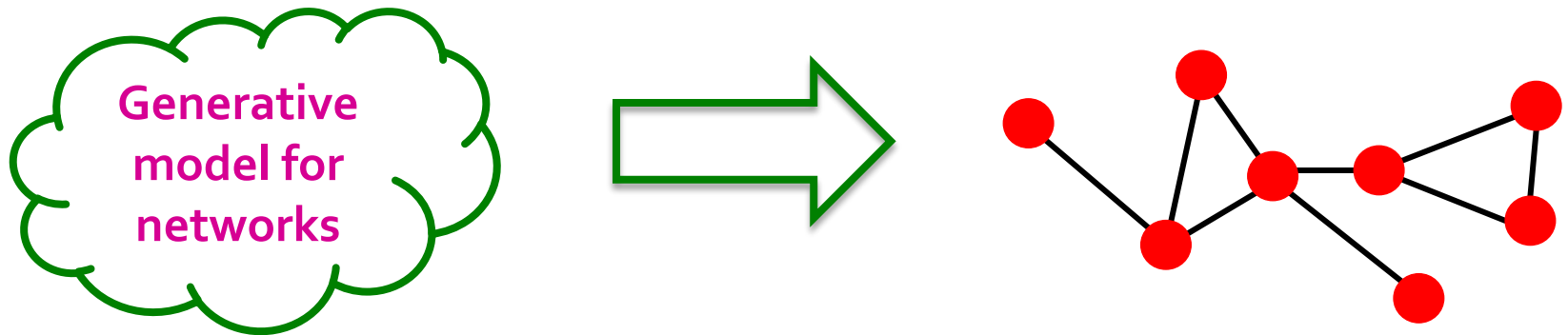

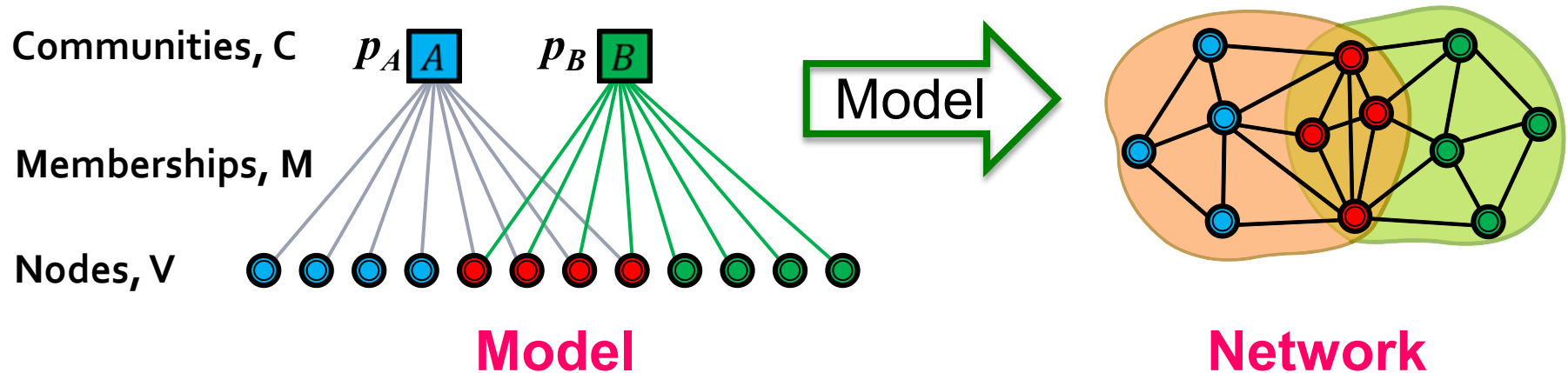- **2) Given a network, find the "best" model**

Model of networks

- **Goal:** **Define a model that can generate networks**
   - The model will have a set of "parameters" that we will later want to estimate (and detect communities)

Generative model for networks

- **Q: Given a set of nodes, how do communities "generate" edges of the network?**

Community-Affiliation Graph

Model   **Network**

- **AMG: Affiliation Graph Model: a generative model B($V, C, M, \{p_c\}$) for graphs:**
  - Nodes **V**, Communities **C**, Memberships **M**
  - Each community **c** has a single probability p_c

  - Later we fit the model to networks to detect communities

AGM: Generative Process

Community Affiliations   **Network**
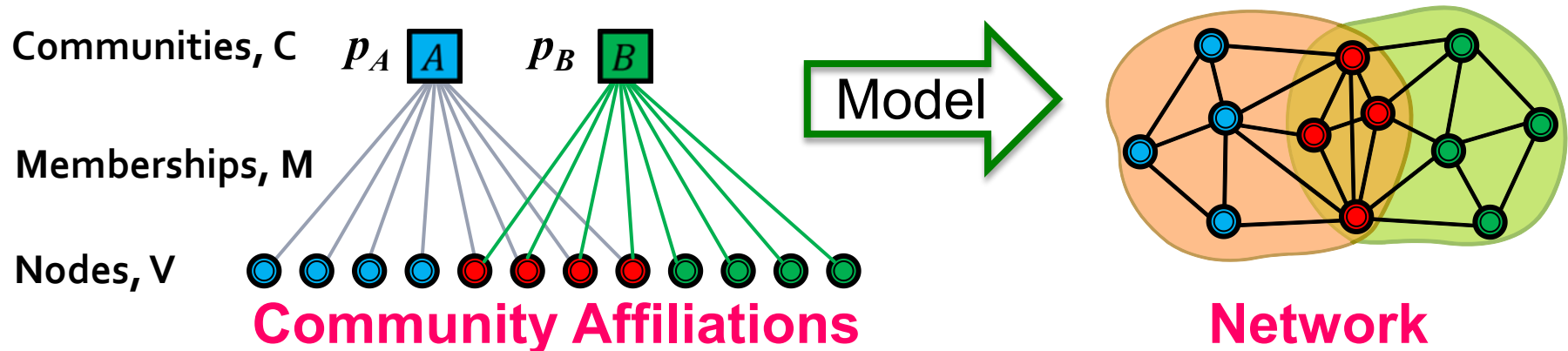
- ## AGM generates the links: For each

  - For each pair of nodes in community A, we connect them with prob. p_A

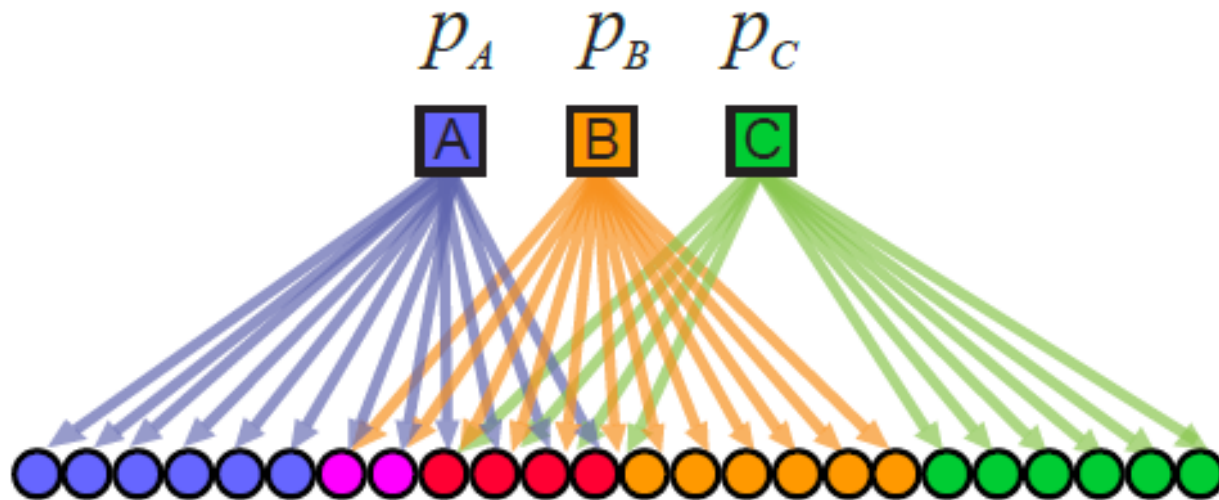 - **The overall edge probability is:**

$$P(u,v) = 1 - \prod_{c \in M_u \cap M_v}(1 - p_c)$$

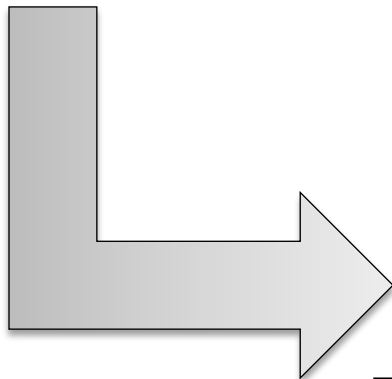If u, v share no communities: $P(u,v) = \varepsilon$

M_u set of communities node u belongs to

Think of this as an "OR" function: If at least 1 community says "YES" we create an edge

Recap: AGM networks

p_A p_B p_C

A   B   C

Model

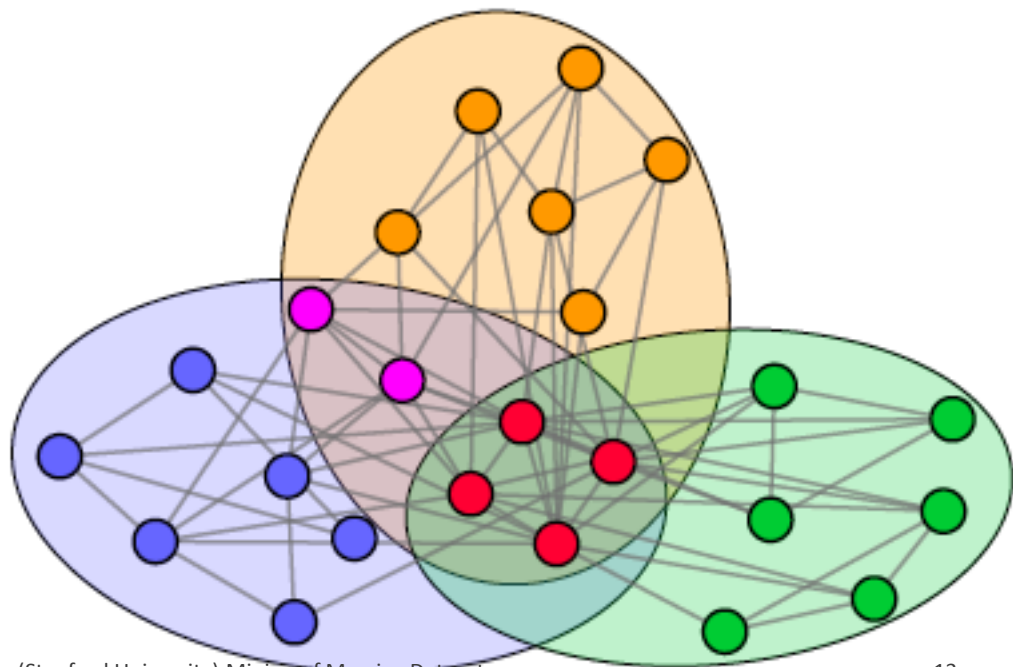**Network**

AGM: Flexibility

- **AGM can express a variety of community structures:** Non-overlapping, Overlapping, Nested