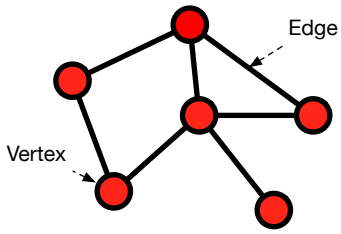
1 What are networks?

A **network** or a **graph** is a collection of discrete entities and the set of interactions among them. We call the entities **vertices** or **nodes** (or sometimes sites or actors), and we call the interactions **edges** or **links** (or sometimes bonds or ties). Any system that we can describe as being composed of identifiable nodes and definable links can be modeled and analyzed as a network.[1]



When using networks, the two most fundamental questions to answer are:

1. *What is a vertex?*
   The answer defines the set V of discrete entities or objects, among which edges exist.

2. *What is an edge?*
   The answer defines the set E of *pairwise* interactions[2] among the vertices, i.e., $E \subseteq V \times V$.

But, for any particular system—say, social interactions among people—there are often multiple ways of answering these two questions. In a social network where vertices are people, we can define multiple *types* of edges, each denoting a different kind of social interaction, such as trust, or friendship, or intimacy, or even just appearing together in a photograph. In a biological network where nodes are genes, an edge might denote a regulatory interaction, or a binding affinity between the corresponding proteins, or even a similarity in terms of evolutionary history. How we answer the two fundamental questions shapes the *kind* of descriptive, predictive, or causal questions we can ask about the underlying system.

[1] Historically, the study of graphs stretches back at least as far as Euler and his 1736 solution to the famous Königsberg Bridge puzzle. Prior to the 20th century, graphs were mainly the domain of mathematicians, and thus the term "graph" has a somewhat mathematical connotation to it. *Graph theory*, for instance, is a branch of mathematics concerned with the mathematical properties of different mathematical families of graphs. During most of the 20th century, sociologists were the main developers of *social network analysis*, which has a more empirical connotation. In the very late 20th century, in part because the computer revolution made it easier to measure, store, and analyze large network data sets, *network science* emerged as an interdisciplinary field, drawing on sociology, computer science, statistics, machine learning, and statistical physics for methods, and with applications in nearly every imaginable field, from science to the humanities.

[2] An "edge" can also be defined as a k-wise interaction, for $k > 2$. Such a network is called a *hypergraph*, denoting these "higher-order" interactions. Examples of hypergraphs include networks of actors and the films they appear in, and scientists and the papers they coauthor. In these notes, edges are typically only pairwise.

Networks appear in nearly every domain of natural, social, and artificial phenomena. The table below illustrates a small portion of that diversity, and the variety of answers to the two fundamental questions. Note that in several cases, for the same underlying system, we can answer the questions differently, producing different kinds of network representations of a single system.

domain	*network*	*vertex*	*edge*
biological	metabolic network	metabolite	metabolic reaction
	protein-interaction network	protein	bonding
	gene regulatory network	gene	regulatory effect
	drug interactions	drug	*in vivo* health interaction
	connectome	neuron	synapse
	physiology	muscles and bones	physical attachment
	pollination network	plants and pollinators	pollination
	food web	species	predation or resource transfer
social	friendship network (offline)	person	friendship, trust, etc.
	friendship network (online)	account	"friendship," follow, etc.
	proximity network	person	physical proximity
	sexual network	person	intercourse
	coauthorships	authors	collaboration
	fictional	character	co-appearance
	animal behavior	animals	interaction
economic	hiring network	workers and jobs	hired into
	international trade	country	trade flow
	purchasing	users and items	purchased
	board of directors	directors and boards	sits on
	inventions	inventors and patents	authored
information	software	function	function call
	World Wide Web	web page	hyperlink
	documents	article, patent, legal case	citation
	artifacts	item, document, concept	relatedness or similarity
	language	word	adjacency in text
technological	Internet (1)	computer	IP network adjacency
	Internet (2)	autonomous system (AS)	GBP connection
	digital circuits	logic gates	wire
	power grid	generating or relay station	transmission line
transportation	rail system	rail station	railroad tracks
	road network (1)	intersection	pavement
	road network (2)	named road	intersection
	airport network	airport	non-stop flight

Networks are models

When answering the two fundamental questions, it's important to remember that a network is a **representation** or a **description** of an underlying system. Sometimes, a network representation is a better approximation than in others, e.g., a network can be a fairly good description of both a system of roads and a system of power transmission lines. But, a network is probably a poor representation of the stars in a galaxy, and captures only some aspects of friendships among people. Similarly, in molecular signaling networks, some signals are mediated by conglomerations of

several proteins, each of which can have its own independent signaling role. A network representation might be a poor model of the underlying signaling system because proteins can interact with other proteins either individually or in groups, and that behavior is difficult to represent as simple pairwise interactions. Throughout the use and study of networks, it is important to keep this fundamental point in mind: networks are models.

Network domains
In the table above, each example network is also tagged by one of six scientific **domains**: biological, social, economic, technological, information, or transportation. These are not mathematical categories, but are rather a rough taxonomy of the kind of underlying system the network models. The domain labels answer the question of what kind of phenomenon are the nodes and edges modeling? The six-domain taxonomy used here originates from the *Index of Complex Networks* (`icon.colorado.edu`), which is a large index of network datasets, organized by domain and **subdomain**, e.g., online vs. offline for social networks.

Biological networks, for instance, include networks of molecules, genes, cells, tissues, and entire species, and are studied across nearly all life-science fields, e.g., molecular biology, microbiology, developmental biology, physiology, neuroscience, ecology, and evolutionary biology.

Social networks include all different kinds of social interactions among people or organizations, except for those that are explicitly economic in nature. Networks of economic interactions, e.g., economic transactions, preferences, and relationships, get their own economic networks category.

Information networks is a broad category, including both web graphs, software graphs, and document networks, all of which are defined by citation-like interactions, as well as semantic networks, where edges denote abstract or ontological relationships. This category also includes networks based on pairwise similarity or relatedness scores that do not obviously fall into some other category.

Technological networks capture systems fundamentally grounded in technology, and especially computer technology, such as the Internet or various other kinds of electronic communication networks. Finally, transportation networks capture systems of physical movement, such as roads, railroads, airplanes, ships, etc., but they can also represent animal transportation systems, e.g., ant trails.

1.1 Graph properties of networks

Because networks are a way of representing an underlying complex system, there are many variations on the basic idea of a set of V nodes and E pairwise interactions. A given network will thus have a particular set of **graph properties**, which help define the specific aspects of the underlying system that the network captures.

For concreteness, we define a graph or network as $G = (V, E)$, where V is the set of vertices, and E is the set of edges. Each edge is a pair of nodes $i, j \in V$ such that $(i, j) \in E$.

1.1.1 Simple graphs

The most basic kind of network is called a *simple* graph, which has the following properties:

1. edges are **undirected**: a connection $(i, j) \in E$ implies a connection $(j, i) \in E$
2. edges are **unweighted**: edges are either present or absent, only (a "binary" relation)
3. there are no self-loops: no edge connects a vertex to itself $(i, i) \notin E$
4. there are no annotations on the nodes, except that nodes are uniquely indexed.

The following figure shows an example of a simple graph, on the left, and a more exotic (non-simple) graph on the right, which provides some examples of how additional information can be stored in a graph representation.

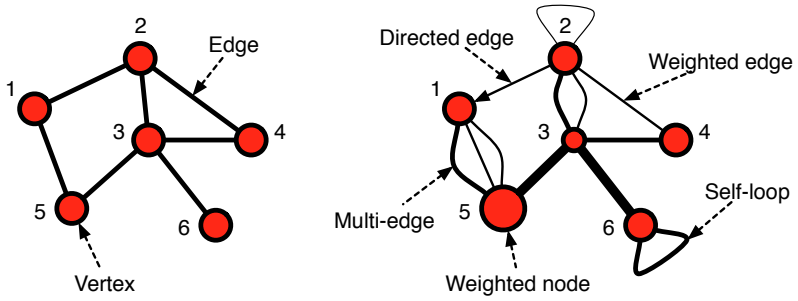


Figure 1: A simple graph (unweighted, undirected, no self-loops), and a more exotic network.

1.1.2 Non-simple graphs

When we relax one of the graph properties of a simple graph, we get a network representation that can capture additional kinds of information about the underlying system. A simple graph is called "simple" because it is the closest to the basic definition of a set of discrete entities V and their pairwise interactions.

The table below lists common graph properties, arranged by whether the property is a function of an *edge*, a *node*, or the whole *network*. Properties with a symbol next to them ($\circ$, $\bullet$, $\star$, $\diamond$, $\dagger$) represent grouped properties, such that a network will have exactly one property from that group.

edge	*node*	*network*
○ **unweighted**	metadata or attributes	⋆ sparse
○ weighted	locations or coordinates	⋆ dense
○ signed	state variables	◇ bipartite
• **undirected**		◇ projection
• directed		† connected
multigraph		† disconnected
timestamps		acyclic
		temporal
		multiplex
		hypergraph

Networks with edge attributes

Many networks have auxiliary data associated with their edges. For example, an edge can be **weighted**, meaning that each edge (i, j) has an associated scalar value or edge weight w_{ij}, which might represent the frequency of interaction (as a natural number $w_{ij} \in \mathbb{Z}$) or the interaction's strength (as a real-valued number $w_{ij} \in \mathbb{R}$). Edges may also be **signed** $w_{ij} \in \{-1, +1\}$, which is a simple way to represent inhibition or activation in a biological system, or distrust and trust in a social system. In fact, edges annotations can be arbitrarily complicated, extending to a whole vector of attributes, a list of "tags," or just a "color" or other kind of categorical variable. When an edge attribute denotes a discrete point in time $t \in \mathbb{N}$ at which that edge exists, we say the network is a **temporal** network, and each group of co-occurring edges (same t) is a network "snapshot." If, on the other hand, edges are annotated by a starting and stopping time, as in a network of phone calls, or a network of physical proximities, then edges have a continuous duration, and we instead say the network has **timestamps**.[3] We elaborate on this distinction a little more below.

A **multigraph** relaxes the prohibition against repeated connections (and generally also the constraint on self-loops), meaning that for at least one pair $i, j \in V$, there exists a multiplicity of edges $(i, j) \in E$. If vertices represent cities, and edges represent driving paths between a pair of cities, then a multigraph will be a reasonable representation because there can be several distinct such paths between a pair of cities. Similarly, in a network of neuron cells, two neurons can have multiple synapses and we might wish to represent each such connection as a distinct edge.

Networks with node attributes

Nodes can also have **attributes** or **metadata** attached to them, denoted as x_i, and these can be categorical variables (sometimes called "labels"), single scalars or vectors representing **state variables**, or even spatial coordinates or **locations** in some metric space.[4] For example, if nodes are cities, node attributes might include the city's population and GPS coordinates. In a social

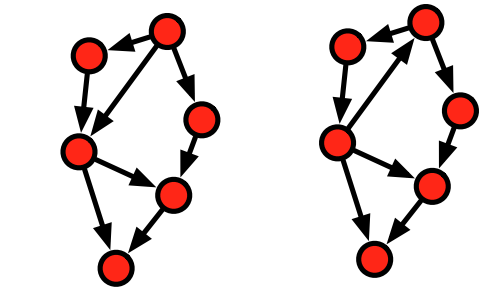
[3] Temporal networks and timestamped networks become indistinguishable when each snapshot spans only a small interval of time, e.g., a few seconds. In this limit, most snapshots will be nearly empty graphs. More commonly, a snapshot represents the accumulation of all timestamped interactions over some period, e.g., a day, a week, or a year.

[4] It has become trendy to refer to such information as "ground truth" when one is trying to predict missing values

network, node metadata may include age, sex, and location. In a protein-interaction network, node attributes might include the molecular weight or Gene Ontology functional labels.

Directed networks

If edges can be asymmetric, we call them **directed**, meaning that the edge (i, j) can occur independently of (j, i). Such directed edges are sometimes called *arcs*, but not always. The World Wide Web is a familiar directed network: webpages are nodes, and hyperlinks are the directed edges. Many biological networks are directed, including gene regulation and neural activation.



We call a directed network **acyclic** if it contains no cycles, i.e., for every possible $i, j \in V$, if there exists a path $i \to \cdots \to j$ then no path exists in the reverse direction $j \to \cdots \to i$. For instance, a citation network is composed of the set of published scientific papers, and an edge $(i, j) \in E$ if paper i cites paper j in its bibliography. Citation networks should be acyclic because of time: a newly published paper can only cite previously published papers, meaning each of its bibliographic links points to older papers. For a cycle to exist, some previously published paper would need to cite a non-yet-published paper, i.e., a paper in the future. (In practice, this kind of oddity does happen, as a result of preprints, and simultaneous publications.) Similarly, in food webs, predation is typically a directed and acyclic activity, with edges pointing "up" from basal species like plants or algae toward, eventually, apex predators like wolves and sharks. However, some food webs are not acyclic because species can predate themselves (cannibalism), and some pairs of species predate each other (a 2-cycle).

Bipartite networks and one-mode projections

To be a k-partite graph, where k is an integer, the following must be true. The set of vertices V is composed of k distinct classes of nodes (e.g., producers and consumers) *and* only nodes of different classes interact (consumers only interact with producers, and vice versa).[5] The simplest and most

on the nodes. However, this is wrong—node annotations are just more data, and thus should not be treated as absolute truth under any circumstances. We'll revisit this idea when we talk about community detection.

[5]Mathematically, a bipartite network can be defined in this way: $V = A \cup B$ where $A \cap B = \emptyset$, and

common form of such graph is the **bipartite** network, where $k = 2$. A popular type of bipartite graph is the actor-film network, in which actors and films represent the two classes, and actors connect to the films in which they play a part.[6]

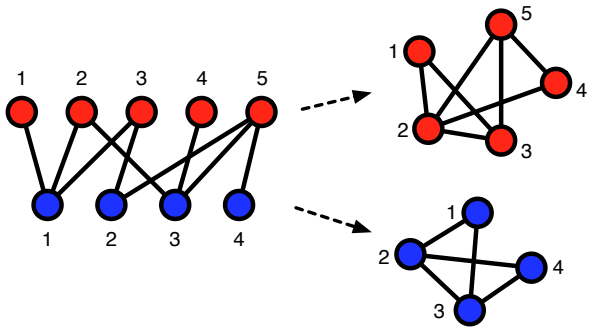


Figure 2: Example of a bipartite graph and its two one-mode projections.

Sometimes, we prefer not to work with a k-partite graph and would instead like to work with a network in which all the nodes are of the same class. This conversion is called a **one-mode projection**. In every k-partite graph, there are k one-mode projections. And, in a one-mode projection, two vertices are connected if and only if they share at least one neighbor in k-partite graph. For instance, to derive the actor-collaboration network from the actor-film network, we add an edge between a pair of actors i, j if they ever appeared in a film together. This procedure is equivalent to saying i, j are connected in the projection if there exists a path of length 2 in the actor-film. Projections can also be weighted, so that (i, j) in the projection is given a weight w_{ij} that corresponds to its multiplicity as a result of the projection procedure, e.g., w_{ij} would count the number of movies in which the actors i, j appeared together.

Warning 1: an important consequence of the one-mode projection procedure is the construction of cliques, i.e., a subgraph of size ℓ in which every pair of nodes is connected. Every node i that is being "projected through," meaning i will not be present in the projection, is represented in the projected graph as a clique of size ℓ, because all pairs of its neighbors are exactly distance two away from each other. For instance, all actors in a particular film will be joined in a clique in the one-mode actor projection.

Warning 2: another important consequence of the one-mode projection procedure is that the same one-mode projection may result from multiple different bipartite networks. In this way, the pro-

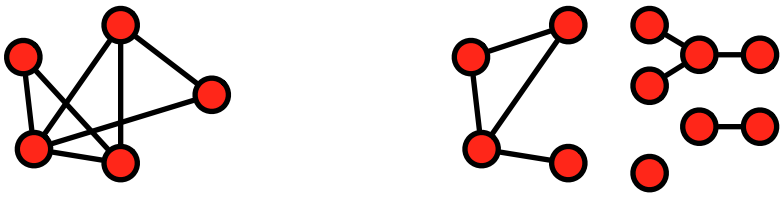
$\forall_{(i,j)\in E}\,((i \in A) \wedge (j \in B)) \vee ((i \in B) \wedge (j \in A))$.

[6] If there are multiple classes of vertices, but edges can exist within each class, then we do not call it a k-partite graph. Instead, it is simply an annotated network with mixed node types.

jection operation is not *one-to-one* i.e., it is not a *bijection*.[7] The projection operation is, however, *surjective*, meaning that any projected network P has at least one bipartite network B such that the projection of B results in P. Can you think of how to prove this statement?

Connected networks and components

A **path** on a network is a sequence of edges $(i, j), (j, k), \ldots, (y, z)$ in which no vertex is repeated,[8] and the *length* of the path is the number of edges in the sequence. A simple network is **connected** if for every pair of nodes $i, j \in V$, there exists a path $i \to \cdots \to j$ (which implies the existence of a path in the reverse direction, $j \to \cdots \to i$). In this case, we say that j is *reachable* from i. If there is some set of nodes $T \subset V$ that is not reachable from some node i, then the network is **disconnected**, meaning that it's composed of at least two *components*. The minimum number of edges for a network to be connected is $m = n - 1$, i.e., a tree.



In a directed network, a set of vertices $T \subseteq V$ that are all pairwise reachable from each other is a *strongly connected* component, while a set of vertices $S \subseteq V$ in which either i is reachable from j *or* j is reachable from i (but not necessarily both), is called a *weakly connected* component. Because of the "or" in the definition, weakly connected components tend to be supersets of strongly connected components.

Temporal, dynamic, and evolving networks

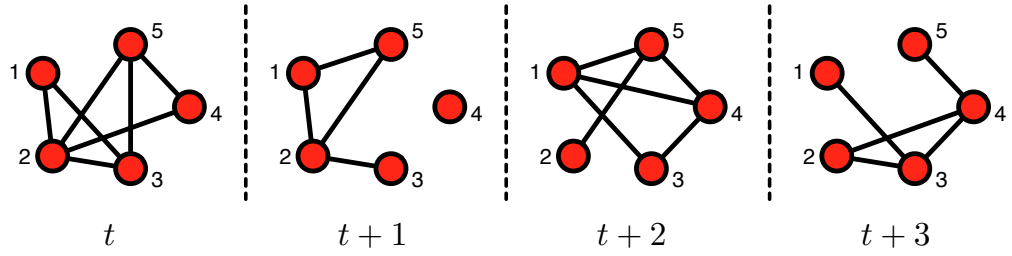
The networks described so far are *static*, meaning that the vertices and edges do not change over time. Graphs that do change over time are an important class of networks. For example, in citation networks, new vertices join the network continuously and each time a new vertex joins, it creates new edges representing citations to older papers. If a network varies over time, we call it a *temporal* or *dynamic* or *evolving* network.

A **temporal** network typically refers to a kind of edge-annotated network in which the annotations represent points in discrete time. We often talk about temporal networks as being a sequence of network "snapshots" $A^{(t_1)}, A^{(t_2)}, \ldots$, where the superscript t_i indexes the passage of discrete time. For instance, all the interactions observed among friends on Monday, and then Tuesday, etc. It

[7] The namespace here is a little cluttered, but we can, with a grin, say that *a bipartite projection is not a bijection.*

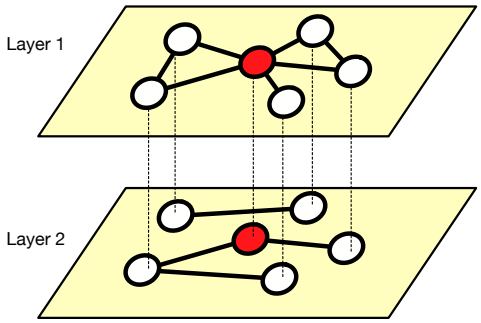
[8] A *trail* is a sequence in which vertices may be repeated, but not edges.

may not always be obvious in the data, but a crucial question is whether A^t represents only the interactions present at time t or the aggregation of all interactions between t and $t+1$. Often, it is the latter.



A **time-stamped** network is one in which edges are annotated with the duration of continuous time in which they existed, e.g., (i, j, t_1, t_2) might represent a phone call or face-to-face interaction that began at time t_1 and ended at time t_2.

Multiplex networks, spatial networks, and hypergraphs
A network in which edges are marked by which "layer" they exist in is called a **multiplex** or **multilayer** network. These networks are used to represent a system in which there are multiple types of interactions, and we store the connectivity of each type in a different "layer" of the multiplex network. A temporal network is a special kind of multiplex network, where these layers form a temporal (ordered) sequence. Crucially, there can dynamics on each vertex that govern which layer some kind of interaction occurs on, so multiplex networks are not merely a special kind of graph in which edges are annotated by different colors or layer numbers.



Spatial networks are a special kind of node-annotated network, in which the annotations represent the node's location in some d-dimensional space. This graph property is most common in

transportation networks, e.g., as road and city networks, airport transportation networks, oil and gas distribution networks, shipping networks, etc., but can also appear in social networks. *Planar* graphs are a special case of spatial networks, in which the nodes are embedded on a 2-dimensional surface and edges do not cross.

Hypergraphs are another type of network, in which edges denote the interaction of more than two vertices, e.g, $E \subseteq V \times V \times V$. Scientific collaboration graphs can be represented as a hypergraph, in which each "edge" is the set of coauthors on a scientific article. However, collaboration networks are more commonly represented as bipartite graphs, in which scientists and papers form two sets of vertices, and scientist-nodes are connected to all the paper-nodes on which they are authors.

2 Four flavors of network analysis and modeling

There are four general approaches in network analysis and modeling: exploratory, explanatory, predictive, and causal.

Exploratory analysis is typically descriptive in nature, and its central goal is to produce a clear view of the kinds of statistical patterns that exist in a network. This approach is largely unsupervised, in the sense that we may not know exactly what we are looking for, or what is interesting about a network's structure. We often use statistical summaries of the network's structure, such as the degree distribution, its community structure, node-level measures like centrality scores or measures of degree assortativity, and more. With a new data set, we often start with exploratory analysis, even if our ultimate interest is in understand what degrees of freedom underlie and explain whatever patterns we may find. That is, the first step is often to identify what patterns are worth explaining in the first place.

The outcome of good exploratory analyses is typically one or more hypotheses about potential causal effects or underlying mechanisms that relate to a network's structure. Exploratory analysis cannot itself test those hypotheses, but it can use *null models* as a way of deciding if some pattern is interesting, e.g., by asking whether a pattern observed in a real-world network is distinguishable from "noise," which is typically operationalized through some kind of *random graph model*. We will explore this topic more in Lectures 2, 3, and 5.

Good exploratory analysis requires *creativity* (to imagine what shape the network might have, and why), *mathematical intuition* (to know what kinds of shapes are possible, and even plausible), *algorithmic tools* (to know how to see that shape and to extract it from the data), and *statistical rigor* (to show that the shape is real and not a clever illusion). Good exploratory analysis finds new and interesting patterns within empirical data, and generates questions to be addressed through a more hypothesis-driven approach.

Many exploratory network analysis tasks can be reduced to the following kind of model. We imagine that an edge (i, j) exists with probability

$$p_{ij} = \Pr(i \to j \,|\, x_i, x_j, \gamma_i, \gamma_j, \theta_{ij}) \ , \tag{1}$$

where each x represent a set of vertex-level observed attributes, each γ represents a set of vertex-level latent (unobserved) attributes, and θ is some latent attributes of the pairing of i and j. For instance, consider a pair of individuals i and j on Facebook. Each person's x contains the attributes they disclose about themselves on Facebook (age, sex, location, etc.). Their γ represents all attributes not disclosed on Facebook (including attributes that Facebook does not ask about), and θ represents latent attributes of the pair (family relationship, work relationship, etc.).

Facebook has many reasons for wanting to know p_{ij}, but they may not care about why it takes that value or how that value changes over time. But, if they knew a functional representation of Eq. (1) for their network, they could do many powerful things, including inferring missing attributes and predicting missing links. The goal of exploratory analysis is, to a large extent, estimating a low-dimensional form for Eq. (1), i.e., a form that depends on many fewer variables than the number of vertices or edges in the network. The more compact the form, the simpler the shape of the network.

Explanatory analysis typically seeks to "explain" some observed pattern as being driven by some other, hopefully more fundamental variable. In social network analysis in sociology, this often takes the form of explaining how some attribute of a node correlates with the network structural patterns that surround that node. For instance, explaining a node's wealth as a function of its central position in the network, or explaining a node's large influence or special behavior via its network connections.

The simplest version of explanatory analysis is to convert the network itself into additional node-level features that encode a node's network characteristics, and then carry out traditional explanatory modeling, i.e., regression, between the "independent" variables and whatever dependent variable we are trying to explain. There are, however, many technical details that make this task non-trivial, most of which stem from the fact that each node's network characteristics are not independent—they are all derived from the same underlying network. In the social sciences, tools like exponential random graph models or stochastic actor-oriented models are network-based methods for explanatory modeling (but, beware [9]).

Predictive modeling aims to construct a predictive model of either node attributes (including future state variables) or structural features, using other network information as the input. Predic-

[9] Under common specifications, these "explanatory" modeling approaches can have significant pathologies that can make their outputs scientifically useless, e.g., see Shalizi and Rinaldo, "Consistency under sampling of exponential random graph models." *Annals Statistics* **41**, 508-535 (2013).

tive modeling often uses machine learning tools to do its work, and these tools can be classification, regression, or probabilistic (often Bayesian) models.

For instance, recommendation algorithms, like on Netflix or Amazon, are really a kind of link prediction algorithm: given a set of nodes attribute representing user preferences and product characteristics, and a set of past connections between users and products, predict which connections are missing; these missing connections are the new product recommendations. If we're making recommendations only among the users (i.e., predicting missing links on the network of users), then it's like the "People You May Know" feature on Facebook, the "Suggestions for you" feature on Instagram, and the "Who to follow" feature on Twitter.

Causal modeling aims to identify cause-and-effect relations that involve networks, such as asking whether being more centrally located in a network *causes* better access to information, or whether a particular social behavior, e.g., buying something or clicking on an ad, is caused by influence from friends' behavior or not.

Generally, causal modeling comes in several flavors. Statisticians and machine learners favor causal inference models that can be applied to an observed network and its dynamics. These techniques can be very complicated in part because they need to isolate and model the many different paths along which influence can travel from one node to another. Nevertheless, these techniques are essential when the goal is making causal claims, but the data are a network, and it is either impractical or unethical to conduct controlled experiments.

In contrast, biologists and some social scientists often favor *network experiments* to tease out causality, e.g., by knocking out an edge or otherwise inducing some change in it and then observing the subsequent effects. Network experiments can be very expensive, because the unit of replication across experiments is the entire network.

Finally, mathematicians and physicists tend to favor mathematical models and *network simulations*, where causal behavior is expressed via a mathematical mechanism, e.g., differential equations or stochastic processes, and then the predictions of the mathematical model are compared with empirical data. While these models can establish sufficiency, i.e., they assume, if the world works like this model, then its consequences are such and such, but they cannot establish necessity, i.e., they say little about alternative explanations.

Good causal modeling (and good explanatory modeling) is often hypothesis driven, meaning that we already have in mind some notion of why and how an effect of interest comes about. But genuinely establishing causality often requires *creativity* (to imagine how a network's shape could lead to the behavior of interest), *mathematical intuition and rigor* (to know what kinds of mechanisms are possible and to show their consequences), *numerical tools* (to simulate the mechanism and

analyze its results), and *statistical rigor* (to show that the hypothesis is supported or not). Good hypothesis-driven analysis identifies and demonstrates believable causes for real effects, and shows that these explanations are better than simple alternatives.

3 Supplemental readings

1. Peruse or skim Chapters 1–5 in *Networks* (background material)
2. Read Chapter 6.1–6.10 in *Networks*