D-separation

Contents

---

1. History and Motivation

In the early 1930s, a biologist named Sewall Wright figured out a way to statistically model the causal structure of biological systems. He did so by combining directed graphs , which naturally represent causal hypotheses, and linear statistical models, which are systems of linear regression equations and statistical constraints, into a unified representation he called *path analysis*.

Wright, and others after him, realized that the *causal structure* of his models (the directed graph) determined statistical predictions we could test without doing experiments. For example, consider a model in which blood sugar causes hunger, but only indirectly.

blood sugar --> stomach acidity --> hunger

The model asserts that blood sugar causes stomach acidity directly, and that stomach acidity causes hunger directly. It turns out that no matter what the strength (as long as its not zero) of these causal connections, which are called "parameters," the model implies that blood sugar and hunger are correlated, but that the *partial correlation* of blood sugar and hunger *controlling* for stomach acity does vanish.

This means that if we could measure blood sugar, stomach acidity and hunger, then we could also test the causal claims of this theory without doing a controlled experiment. We could invite people off the street to come into our office, take measurements of their blood sugar, stomach acidity and hunger levels, and examine the data to see if blood sugar and hunger are significantly correlated, and not significantly correlated when we control for stomach acidity. If these predictions don't hold, then the *causal* claims of our model are suspect.

Although it is easy to derive the two statistical consequences of this path analytic causal model, in general it is quite hard. In the 1950s and 60s, Herbert Simon (1954) and Hubert Blalock (1961) worked on the problem, but only solved it for a number of particular causal structures (directed graphs). The problem that Wright, Simon, and Blalock were trying to tackle can be put very generally: what are the testable statistical consequences of causal structure. This question is central to the epistemology and methodology of behavioral science, but put this way is still too vague to answer mathematically.

By assuming that the causal structure of a model is captured entirely by the directed graph part of the statistical model, we move a step closer towards framing the question in a clear mathematical form. By clarifying what we mean by "testable statistical consequences" we take one more step in this direction.

Although Wright, Blalock and Simon considered vanishing correlations and vanishing partial correlations, we will be a little more general and consider independence and conditional independence , which include vanishing correlation and partial correlation as special cases, as one class of "testable statistical constraints." These are not the only statistical consequences of causal structure. For example, Spearman (1904), Costner (1971), and Glymour, Scheines, Spirtes, and Kelly (1987) used the vanishing tetrad difference to probe the causal structure of models with variables that cannot be directly meausured (called latent variables) like general intelligence. But clearly conditional independence constraints are central, and here we restrict ourselves to them.

So here is a general question that is precise enough to answer mathematically: Can we specify an algorithm that will compute, for any directed graph interpreted as a linear statistical model, all and only those independence and conditional independence relations that hold for all values of the parameters (causal strengths).

Judea Pearl, Dan Geiger, and Thomas Verma, computer scientists at UCLA working on the problem of storing and processing uncertain information efficiently in artificially intelligent agents, solved this mathematical problem in the mid 1980s. Pearl and his colleagues realized that uncertain information could be stored much more efficiently by taking advantage of conditional independence, and they used directed acyclic graphs (graphs with no loops from a variable back to itself) to encode probabilities *and* the conditional independence relations among them. D-separation was the algorithm they invented to compute all the conditional independence relations entailed by their graphs (see Pearl, 1988). Peter Spirtes, Clark Glymour, and Richard Scheines, working on the problem of causal inference at the Philosopy Department at Carnegie Mellon University in the late 1980s and early 1990s, connected the artificial intelligence work of Pearl and his colleagues to the problem of testing and discovering causal structure in behavioral sciences (see Spirtes, Glymour, and Scheines, 1993). The work didn't stop there, however. Pearl and his colleagues proved many more interesting results about graphical models, what they entail, and algorithms to discover them (see Judea Pearl's home page). In 1994, Spirtes proved that d-separation correctly computes the conditional independence relations entailed by cyclic directed graphs interepred as linear statistical models (Spirtes, 1994), and in the same year Richardson (1994) developed an efficient procedure to determine when two linear models, cyclic or not, are d-separation equivalent. In 1996, Pearl proved that d-separation correctly encodes the independencies entailed by directed graphs with or without cycles in a special class of discrete causal models (Pearl, 1996). Also in 1996, Spirtes Richardson, Meek, Scheines, and Glymour (1996) proved that d-separation works for linear statistical models with correlated errors. So it should be obvious that d-separation is a central idea in the theory of graphical causal models. In the rest of this module, we try to explain the ideas behind the definition and then give the definition formally. At the end of the module you can run a few Java applets which provide interactive tutorials for these ideas.

Return to Top of Page

---

2. D-separation Explained

In this section we explain the ideas that underly the definition of d-separation. If you want to go to the section in which we give a formal definition of d-separation, click here.

Although there are many ways to understand d-separation, we prefer using the ideas of **active path** and **active vertex** on a path (see the active path applet).

Recall the motivation for d-separation. The "d" in d-separation and d-connection stands for dependence. Thus if two variables are d-separated relative to a set of variables Z in a directed graph, then they are independent conditional on Z in all probability distributions such a graph can represent. Roughly, two variables X and Y are independent conditional on Z if knowledge about X gives you no extra information about Y once you have knowledge of Z. In other words, once you know Z, X adds nothing to what you know about Y.

Intuitively, a path is *active* if it carries information, or dependence. Two variables X and Y might be connected by lots of paths in a graph, where all, some, or none of the paths are active. X and Y are d-connected, however, if there is *any* active path between them. So X and Y are d-separated if *all* the paths that connect them are *inactive*, or, equivalently, if no path between them is active.

So now we need to focus on what makes a path active or inactive. A path is active when *every* vertex on the path is active. Paths, and vertices on these paths, are active or inactive *relative* to a set of other vertices Z. First lets examine when things are active or inactive relative to an empty Z. To make matters concrete, consider all of the possible undirected paths between a pair of variables A and B that go through a third variable C:

1) A --> C --> B
2) A <-- C <-- B
3) A <-- C --> B
4) A --> C <-- B

The first is a directed path from A to B through C, the second a directed path from B to A through C, and the third a pair of directed paths from C to A and from C to B. If we interpret these paths causally, in the first case A is an indirect cause of B, in the second B is an indirect cause of A, and in the third C is a common cause of A and B. All three of these causal situations give rise to association, or dependence, between A and B, and all three of these undirected paths are *active* in the theory of d-separation. If we interpret the fourth case causally, then A and B have a common effect in C, but no causal connection between them. In the theory of d-separation, the fourth path is *inactive*. Thus, when the conditioning set is empty, only paths that correspond to causal connection are active.

We said before that a path is active in the theory of d-separation just in case *all* the vertices on the path are active. Since C is the only vertex on all four paths between A and B above, it must be active in the first three paths and inactive in the fourth.

What is common to the way C occurs on the first three paths but different in how it occurs on the fourth? In the first three, C is a *non-collider* on the path, and in the fourth C is a *collider* (See the module on directed graphs for an explanation of colliders and non-colliders). When the conditioning set is empty, non-colliders are *active*. Intuitively, non-colliders transmit information (dependence). When the conditioning set is empty, colliders are *inactive*. Intuitively, colliders don't transmit information (dependence). So when Z is empty, the question of whether X and Y are d-separated by Z in a graph G is very simple: Are there any paths between X and Y that have no colliders?

Now consider what happens when the conditioning set is not empty. When a vertex is in the conditioning set, its status with respect to being active or inactive flip-flops. Consider the four paths above again, but now lets consider the question of whether the variables A and B are d-separated by C (in boldface).

1) A --> **C** --> B

2) A <-- **C** <-- B
3) A <-- **C** --> B
4) A --> **C** <-- B

In the first three paths, C was active when the conditioning set was empty, so now C is inactive on these paths. To fix intuitions, again interpret the paths causally. In the first case the path from A to B is blocked by conditioning on the intermediary C, similarly in case 2, and in case 3 you are conditioning on a common cause, which makes the effects independent. Philosophers like Reichenbach, Suppes, and Salmon, as well as mathematicians like Markov, worked out this part of the story. Reichenbach called it the "Principle of the Common Cause," and Markov expressed it as the claim that the present makes the past and future independent, but all were aware that conditioning on a causal intermediary or common cause, which are non-colliders in directed graphs interpreted causally, cuts off dependence that would otherwise have existed.

In the fourth case, C is a collider and thus inactive when the conditioning set is empty, so is now active. This can also be made intuitive by considering what happens when you look at the relationship between two independent causes after you condition on a common effect. Consider an example given by Pearl (1988) in which there are two independent causes of your car refusing to start: having no gas and having a dead battery.

dead battery --> **car won't start** <-- no gas

Telling you that the battery is charged tells you nothing about whether there is gas, but telling you that the battery is charged *after* I have told you that the car won't start tells me that the gas tank must be empty. So independent causes are made dependent by conditioning on a common effect, which in the directed graph representing the causal structure is the same as conditioning on a collider. David Papineau (1985) was the first to understand this case, but never looked at the general connection between directed graphs interpreted causally and conditional independence.

The final piece of the story involves the descendants of a collider. Whereas conditioning on a collider activates it, so does conditioning on any of its descendants. No one understood this case before Pearl and his colleagues.

We built a Java applet to help you understand active paths and active vertices on the path. You can draw a graph, pick vertices and a conditioning set, and then pick a path between the vertices you have selected. You then must decide which vertices are active or inactive on the path. The applet will give you feedback, and, if you like, explanations. Run the applet on active paths and active vertices.

Return to Top of Page

---

2. D-separation formally defined

In this section we define d-separation formally. If you want to go to the section that explains the ideas that underly the definition of d-separation, then click here.

The following terms occur in the definition of d-separation:

- undirected path,
- collider
- non-collider
- descendant

Each of them is defined and explained in the module on directed graphs. It is easier to define d-connection, and then define d-separation as the negation of d-connection.

D-connection:
If G is a directed graph in which X, Y and Z are disjoint sets of vertices, then X and Y are d-connected by Z in G if and only if there exists an undirected path U between some vertex in X and some vertex in Y such that for every collider C on U, either C or a descendent of C is in Z, and no non-collider on U is in Z.

X and Y are d-separated by Z in G if and only if they are not d-connected by Z in G.

Since you can't really learn a definition unless you try to apply it, we built a Java aapplet that lets you experiment with this definition. The applet lets you draw any graph you like, pick vertices and a conditioning set, state your opinion about whether the vertices you have picked are d-separated or d-connected by the conditioning set you have chosen, and finally tells you whether you are right or wrong. Run the applet on the definition of d-separation

Return to Top of Page

References

Blalock, H. (Ed.) (1971). Causal Models in the Social Sciences. Aldine-Atherton, Chicago.

Blalock, H. (1961). Causal Inferences in Nonexperimental Research. University of North Carolina Press, Chapel Hill, NC.

Costner, H. (1971). Theory, deduction and rules of correspondence. Causal Models in the Social Sciences, Blalock, H. (ed.). Aldine, Chicago.

Geiger, D. and Pearl, J. (1989b). Axioms and Algorithms for Inferences Involving Conditional Independence. Report CSD 890031, R-119-I, Cognitive Systems Laboratory, University of California, Los Angeles.

Geiger, D., Verma, T., and Pearl, J. (1990) Identifying independence in Bayesian Networks. Networks 20, 507-533.

Glymour, C., Scheines, R., Spirtes, P., and Kelly, K. (1987). Discovering Causal Structure. Academic Press, San Diego, CA.

Kiiveri, H. and Speed, T. (1982). Structural analysis of multivariate data: A review. Sociological Methodology, Leinhardt, S. (ed.). Jossey-Bass, San Francisco.

Pearl, J. (1988). Probabilistic Reasoning in Intelligent Systems. Morgan and Kaufman, San Mateo.

Pearl, J. (1995). Causal diagrams for empirical research. Biometrika, 82, pp. 669-710.

Pearl, J. and Dechter, R. (1989). Learning structure from data: A survey. Proceedings COLT '89, 30-244.

Pearl, J., Geiger, D. and Verma, T. (1990). The logic of influence diagrams. Influence Diagrams, Belief Nets and Decision Analysis. R. Oliver and J. Smith, editors. John Wiley & Sons Ltd.

Pearl, J. and Verma, T. (1991). A theory of inferred causation. Principles of Knowledge Representation and Reasoning: Proceedings of the Second International Conference, Morgan Kaufmann, San Mateo, CA.

Pearl, J. and Verma, T. (1990). A Formal Theory of Inductive Causation. Technical Report R-155, Cognitive Systems Labratory, Computer Science Dept. UCLA.

Pearl, J. and Verma, T. (1987). The Logic of Representing Dependencies by Directed Graphs. Report CSD 870004, R-79-II, University of California at Los Angeles Cognitive Systems Laboratory.

Richardson, T. (1994). Properties of Cyclic Graphical Models. MS Thesis, Carnegie Mellon University.

Richardson (1995). A Polynomial-Time Algorithm for Deciding Markov Equivalence of Directed Cyclic Graphical Models, Technical Report PHIL-63, Philosophy Department, Carnegie Mellon University.

Reichenbach, H. (1956). The Direction of Time. Univ. of California Press, Berkeley, CA.

Salmon, W. (1980). Probabilistic causality. Pacific Philosophical Quarterly 61, 50-74.

Spirtes, P. (1994a). "Conditional Independence in Directed Cyclic Graphical Models for Feedback." Technical Report CMU-PHIL-54, Department of Philosophy, Carnegie Mellon University, Pittsburgh, PA.

Spirtes, P., Glymour, C., & Scheines, R. (1993). Causation, prediction, and search. Springer-Verlag Lecture Notes in Statistics 81,. Springer-Verlag, NY.

Spirtes, P., Richardson, T., Meek, C., Scheines, R., and Glymour, C. (1996). Using d-separation to calculate zero partial correlations in linear models with correlated errors. Technical Report CMU-PHIL-72, Dept. of Philosophy, Carnegie Mellon University, Pittsburgh, PA, 15213.

Suppes, P. (1970). A Probabilistic Theory of Causality. North-Holland, Amsterdam.

[Return to Top of Page]

Richard Scheines (R.Scheines@andrew.cmu.edu)/ Carnegie Mellon University