# Confounding and Directed Acyclic Graphs (DAGs)

## Confounding

* Often defined as variables that affect both the treatment and the outcome (but the latter NOT through the treatment)
  + For example, if treatment is assigned based on a coin flip, then that shouldn’t affect the outcome i.e. the coin flip isn’t a confounder.
  + For example, if people with a family history of cancer are more likely to develop cancer, but family history was not a factor in the treatment decision, then family history isn’t a confounder.
* For example, if older people are at higher risk of cardiovascular disease are more likely to receive medical attention (the treatment), age IS a confounder.

### Confounder Control

* Interested in:
  1. Identifying a set of variables X that will make the ignorability assumption (recall this assumption states that the treatment assignment is random given a set of covariates) hold, which will hence be sufficient to control for confounding.
  2. Using statistical method to control for these variables and estimate causal effects.

### Causal Graphs

* Which variables need to control for isn’t a simple question.
* Causal graphs help to answer that question.

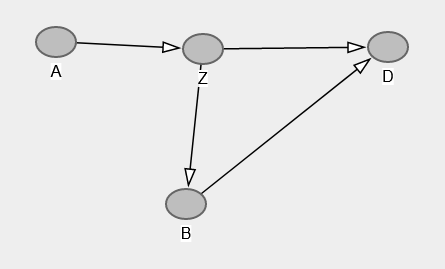
## Causal Graphs

### Overview

* Helpful in identifying variables to control for.
* Makes assumptions in the model/relationships between variables explicit and permits a peer-review of those assumptions.
* The direction of the arrow in a directed graph indicates the direction of the hypothesised relationship.
* If there is no arrow, it means there is a relationship, but the causal direction is unknown.
* Graphical models:
  1. Encode assumptions about the relationships between variables e.g. which are independent, dependent, conditionally independent etc.
  2. Can be used to derive non-parametric causal effect estimators.

### Terminology

* Nodes/vertices can be single or a collection of variables.
* Variables connected by an edge are adjacent.
* A path is a way to get from one node to another along edges.



* A is Z’s parent.
* B is a child of Z.
* D is a descendant of A.
* Z is an ancestor of D.
* D has 2 parents, B and Z. (A node can have multiple parents.)

### DAGs

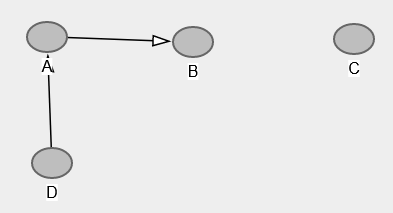
* Can’t have any undirected paths.
* No cycles e.g. A B Z A … in a never-ending loop.
* For example, Z affects A & B and A affects B but nothing cycles back to Z.
* Help us to determine the set of variables we need to control for to achieve ignorability.

## Relationship between DAGs and Probability Distributions

### DAGs and Probability Distributions

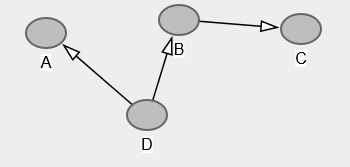
* DAGs encode assumptions about dependencies between nodes/vertices.
  1. Which variables are independent from each other.
  2. Which variables are conditionally independent from each other.
  3. Ways that we can factor and simplify the joint distribution.

### Example 1



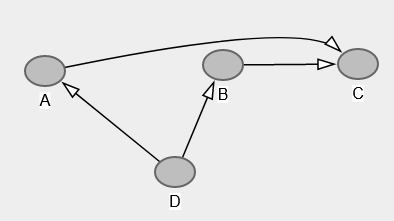
* i.e. C is independent of all variables.
  1. C is sitting off by itself and independent so conditioning on A, B and D won’t tell us anything about the probability of C.
* i.e.
  1. B is affected directly by A, and indirectly by D.
  2. All that really matters is A, because if we know A, then we know everything about the probability of B. D doesn’t give us any additional information, because it affects A, but we’re already conditioning on A.
* 1. B and D are marginally dependent because D affects A which affects B.
  2. For the same sorts of reasons that B is related to D, but only through A.

### Example 2



* i.e.
  1. The only thing affecting A is D, so as long as we condition on D, we’ll know everything we need to about A.
* i.e.
  1. C doesn’t tell us anything new about D, unlike A and B

### Example 3

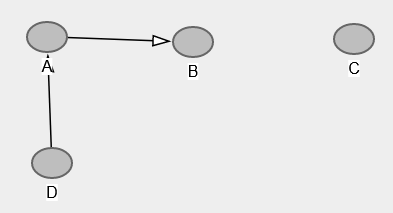


* i.e.
  1. A is directly affecting C, but B is only indirectly related to A through C and/or D so we can drop the conditioning on B.
* i.e.
  1. The same kind of idea applies here with respect to C.

### Decomposition of Joint Distribution

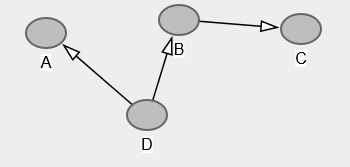
* Can decompose the joint distribution by sequential conditioning only on sets of parents.
  1. Start with roots (nodes with no parents).
  2. Proceed down the descendent line, where we always condition on the parents.

### Example 1



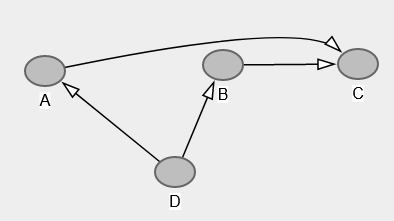
* The roots (independent random variables) are C and D.
* C doesn’t have any children so we’re done.
* D has a child, A and A has a child B so we multiply those conditional probabilities.
  1. Decomposition is implied by the DAG and doesn’t hold in general, but does hold if the DAG is correct.

### Example 2



* Only D doesn’t have a parent, so we start there.
* D has two children, A and B.
* A doesn’t have a child, so we’re done there but C does.

### Example 3



* This is the same as the previous example except that C has two parents, which only changes the last term in the probability distribution.

### Compatibility between DAGs and Distributions

* The DAG in the last example admits the following factorisation:
* … which means that the probability distribution and DAG are compatible.
* But particular probability functions don’t necessarily imply a unique DAG.
* For example, in one DAG but in another DAG , so they convey different information.
  1. But they both convey that A and B are dependent i.e.
  2. If you start with the latter, either DAG could be correct. i.e. both are compatible.

## Paths and Associations

### Types of Paths

* Fork:  i.e. E affects D and F. The middle part affects the outer parts.
* Chain: . Everything flows in one direction.
* Inverted fork: . The outer parts both affect the inner part.

### When Do Paths Induce Associations?

* If nodes A and B are on the ends of a path, they’re associated via this path if:
  1. Some information flows to both of them.
  2. Information from one makes it to the other.

### Information Flows to Both Nodes

* For example, nodes at the end of a path:
* This implies that A and B are not independent (because E affected both of them).
* A more complex example is:
  1. E is the node triggering the chain reaction that ends up affecting both A and B.
  2. And hence A and B are still dependent on each other through this path.
* For example, the chain:
* Or a longer chain:
  1. A is getting information to B (albeit via G, D and F) so A and B are dependent.

### Paths That do not Induce Associations

* Inverted forks:
  1. Both A and B affect G, so G is known as a collider (two arrows going into the same node).
  2. But there is no information flowing from G to either A or B.
  3. A and B are independent (if this was the only path between them).
* If there is a collider anywhere on the path from A to B, then no association between A and B comes from this path:
  1. G is a collider in this example too.
  2. There is no free flow of information between A and B at any point.

## Conditional Independence (d-separation)

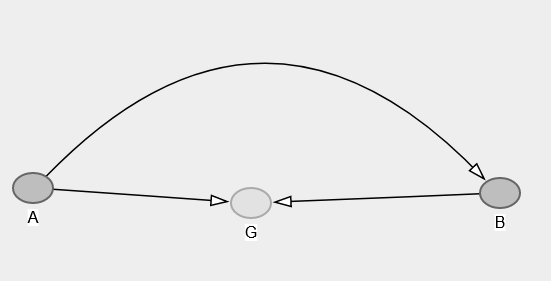
### Blocking

* Paths can be blocked by conditioning on nodes in the path.
* Consider
  1. If we condition on G (control for G), we block the path from A to B, because G is the variable permitting information to flow from A to B i.e. A and B become independent.
* Suppose A = temperature, G = whether footpaths are icy, B = whether a person falls.
  1. No arrow between A and B so no direct relationship between temperature and the chance of a person falling.
  2. A and B are marginally associated e.g. temperature and the chance of a fall are related.
  3. If we block/condition on G and restrict to situations in which the footpaths are icy/not ice, then temperature and the chance of a fall are should no longer related via this path.
     1. Another way of saying this is that we’re making the situation the same for everyone.
* Associations on a fork can also be blocked.
* Consider
  1. If we condition on G, the path from A to B is blocked because G is now being held constant, which means that A and B are now independent.

### Colliders

* The opposite situation occurs if a collider is conditioned on.
* Consider an inverted fork:
  1. A and B are not associated via this path, but conditioning on G induces an association between A and B.

### Conditioning on Colliders

* Suppose A is the state of an off/on switch, B is the state of a second off/on switch, G is whether the lightbulb is lit up.
* A is determined by a coin flip, B is determined by a separate independent coin flip, and G is lit up iff A and B are both “on”.
* A and B are clearly independent from each other, but both affect G.
* BUT A and B are now dependent given G: if G is off, then A must be off if B is on, or vice versa.
  1. 
  2. When you condition on G, you open up a path between A and B, so now A and B are conditionally dependent given G.

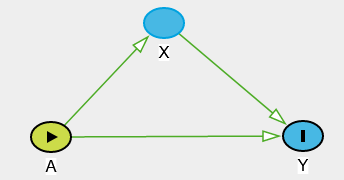
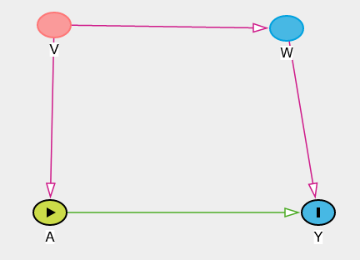
### Rules for d-separation

* A path is d-separated (whether a set of nodes creates independence between variables on a given path) by a set of nodes, C if:
  1. D = dependence
  2. Suppose we have a path that has a dependency between nodes that we want to know if a set of variables, C, removes that dependency (d-separates them).
  3. It contains a chain and the middle part is in C. e.g. if the middle of a chain is E, then E needs to be a member of C.
  4. It contains a fork and the middle part is in C.
  5. It contains an inverted fork and the middle part is NOT in C, not are any descendants of it (the collider) are in C.

### d-separation

* Two nodes, A and B, are d-separated by a set of nodes C if C blocks every path from A to B.
  1. If we successfully condition on C, then A and B become independent i.e.

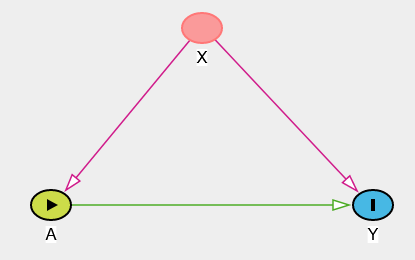
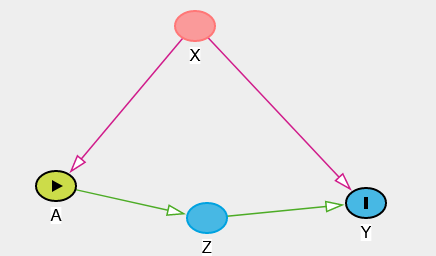
## Confounding Revisited

* Recall that the informal definition of a confounder is a variable affecting both the treatment and the outcome.
* A simple DAG where X is a confounder between treatment A and outcome Y:
  1. 
* Consider a more complicated example:
  1. 
  2. V directly affects A
  3. V indirectly affects Y through its affect in W.
  4. It’s reasonable to argue that V is a confounder.

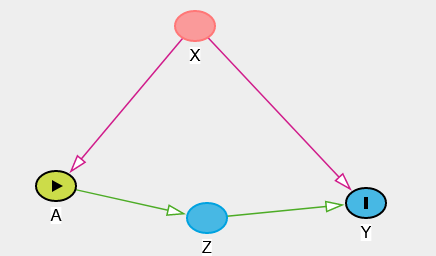
### Controlling for Confounding

* What matters isn’t identifying specific confounders (i.e. calling a specific variable a confounder) but the set of variables sufficient to control for confounding.
* Need to block backdoor paths from treatment to outcome.

### Front-door Paths

* A front-door path from A to Y is one that begins with an arrow emanating out of A (going in the direction of the arrows).
* 
* is a front-door path from A to Y.
* 
* is a front-door path from A to Y via Z.
* Don’t worry about front-door paths because they capture treatment effects (i.e. things we’re interested in).
* In the second example, wouldn’t want to block or control for Z because it’s part of the treatment effect.
  1. Only care about the effect on Y if A is manipulated.
* Causal mediation analysis – understanding front-door paths from A to Y.
  1. Care about front-door paths incl. quantifying it but still wouldn’t want to block Z.

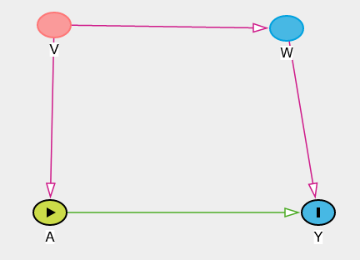
### Back-door Paths

* These are the ones we need to worry about when it comes to controlling for confounding.
* A back-door path from treatment A to outcome Y are paths from A to Y travelling through arrows going into A
* 
  1. is a back-door path from A to Y.
  2. This path has nothing to do with A causing Y i.e. no treatment effect but A and Y are still related.
  3. If we examine the marginal associations between A and Y, some of that association is due to the direct relationship via Z, but some will be due to the fact that X causes both A and Y.
  4. Need to separate out the treatment effect from the confounding effect occurring through this back-door path.
* Back-door paths confound the relationship between Z and Y.
  1. These need to be blocked.
* To sufficiently control for confounding, we need to identify a set of variables that block all back-door paths from treatment to outcome, which then means that ignorability holds.

## Backdoor Path Criterion

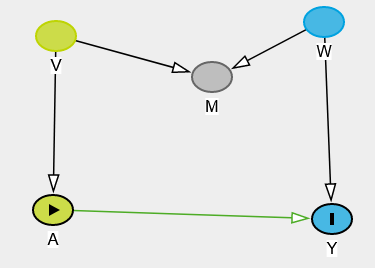
* A set of variables is sufficient to control for confounding if:
  1. It blocks all back-door paths from treatment to outcome
  2. It doesn’t include any descendants of treatment (i.e. part of the causal effect of treatment).
* The set of variables isn’t necessarily unique i.e. not necessarily strictly one set of variables that will satisfy this criterion.

### Example 1



* V in particular is a confounder as it affects A directly, and Y indirectly via W.
* There is just one back-door path from A to Y: .
  1. Not blocked by a collider.
* Sets of variables sufficient to control for confounding: , ,
  1. Can pick any one of these; typically choose the smallest reasonable set.

### Example 2

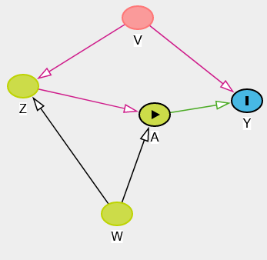


* There is just one back-door path from A to Y:
  1. It is blocked by a collider.
  2. There actually isn’t any confounding here, because V directly affects treatment, but the information from V can’t flow to Y. Similarly, W affects the outcome, but the information from W can’t flow back to A.
  3. That is, don’t have to control for anything and an unadjusted analysis is perfectly appropriate.
* If M is controlled for (even if unintentionally because you don’t realise it’s a collider), it opens up a path between V and W.
  1. V and W were independent marginally but are conditionally dependent.
  2. Sets of variables sufficient to control for confounding:

,

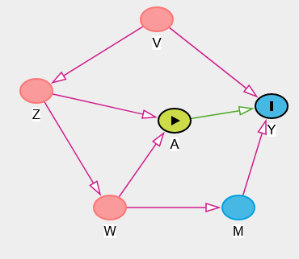
* 1. No harm done if control just for V or W, but if you control for M, you have to control for V, W or both.
  2. Controlling just for M introduces a confounder as it opens up a back-door path.

### Example 3



* There are 2 back-door paths from A to Y:
* First path:
  1. No colliders, so controlling for Z, V or both is sufficient (but do have to control for at least one of them to block the path).
* Second path:
  1. Collision at Z so it’s already blocked.
  2. Controlling for Z opens a path from W to V, which can be blocked with
* Overall, the following sets of variables are sufficient to control for confounding:
  1. Not as these would open up a path with W and V in the first instance, and A and Z in the second.
  2. The minimal set is , which is the ideal set if you know what it is.

### Example 4



* 3 back-door paths from A to Y:
* First path:
  1. Can block with Z or V or both.
* Second path:
  1. No colliders on this path.
  2. Can block with W, Z or V or any combination of these.
* Third path:
  1. Can block with W or M or both.
* Overall, the following sets of variables are sufficient to control for confounding:
  1. Many options here.

### Conclusion

* A DAG is an assumption and can be wrong, but could still be the case that the variables being controlled for are sufficient.
* Starts to get you thinking more formally about the variables in your model.
* Sensitivity analyses etc become more important if you’re not sure about the DAG.
* The back-door path criterion for variable selection requires knowing the DAG.
  1. But for many problems, it could be difficult to construct an accurate DAG.

## Disjunctive Cause Criterion

### Variable Selection

* An alternative method for choosing variables to control for.
* Control for all (observed) causes of the exposure, the outcome or both.
* Researchers don’t need to know the entire causal, but just the list of variables that affects the exposure or outcome.

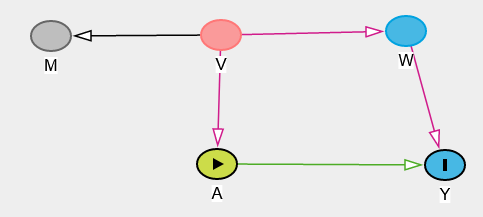
### Property

* If there’s a set of observed variables that satisfies the back-door path criterion, then the variables selected based on the disjunctive cause criterion will be sufficient to control for confounding.

### Example

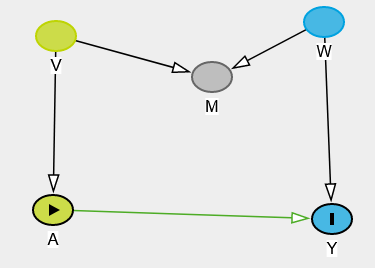
* Observed pre-treatment variables: {M, W, V}
* Unobserved pre-treatment variables:
* Suppose we know that W and V are causes of A, Y or both, and M is not a cause of either A or Y.
* Assuming that you don’t know the DAG, but do have some information about the variables. Then you can compare two methods for selecting variables:
  1. Use all pre-treatment covariates: {M, W, V}
  2. Use variables based on the disjunctive cause criterion: {W, V}

### Hypothetical DAG 1



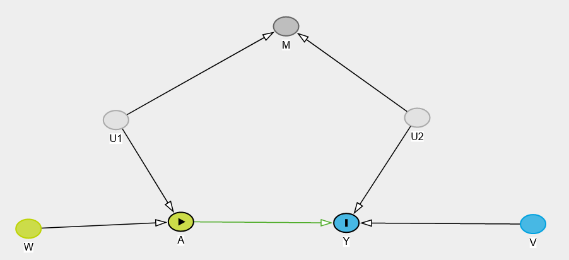
* Approach 1: satisfies back-door path criterion
  1. Only one back-door path from A to Y, which is through V and W
* Approach 2: satisfies back-door path criterion
  1. Again, blocking the one back-door path.

### Hypothetical DAG 2



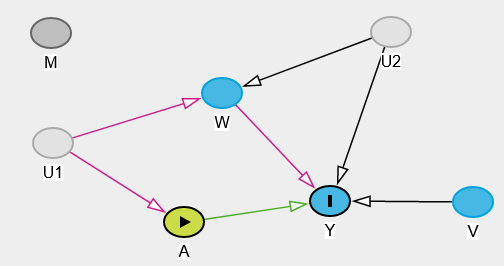
* Technically don’t need to control for anything as the back-door path is already blocked due to the collision at M.
* Approach 1: satisfies back-door path criterion
  1. Controlling for M opens up the path, but we’re also controlling for V and W, so this is fine.
* Approach 2: satisfies back-door path criterion
  1. Not conditioning on the collider so no new paths are being opened up that aren’t already blocked.

### Hypothetical DAG 3



* Can’t control for unobserved variables.
* Don’t need to control for anything in this DAG because the only back-door path includes a collider.
* Approach 1: doesn’t satisfy back-door path criterion
  1. Controlling for M introduces confounding when there previously was none, so a back-door path is opened up.
* Approach 2: satisfies back-door path criterion
  1. Not conditioning on the collider so no new paths are being opened up that aren’t already blocked.

### Hypothetical DAG 4



* There’s actually no way to satisfy the back-door path criterion in this example just by controlling for the observed variables.
* There’s a path from A to Y via W, but there’s a collision at W that opens up a path from U1 to U2, so controlling for W means you can get to Y via that back-door path.
* And if you control for all three, you again open up this back-door path from U1 to U2 that allows for A to be associated with Y in a non-causal way.
* Similarly, if you control for W and V using the disjunctive cause criterion, the same result occurs.

### Conclusion

* Doesn’t always select the smallest set of variables to control for, BUT it’s conceptually simpler.
* Guaranteed to select a set of variables that are sufficient to control for confounding IF:
  1. Such a set exists
  2. All the observed causes of A and Y are correctly identified.