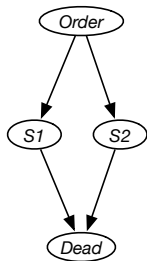


# Counterfactual Reasoning

- The do calculus is for intervening before observing.  
 $P(x \mid y, do(z))$  means the probability of  $x$  after doing  $z$  then observing  $y$ .
- The other case is observing then intervening.
- When the intervention is different from what actually happened, this is **counterfactual reasoning**, which is asking “what if something else were true” .
- Let's use a more general notion of counterfactual, where you can ask “what if  $x$  were true” without knowing whether  $x$  were true.

## Example: firing squad



- A captain can give an order to a number of shooters who can each shoot to kill a prisoner condemned to death.
- Each shooter can think “I wasn’t responsible for killing the prisoner, because the prisoner would be dead even if I didn’t shoot”.
- The captain some probability of issuing order.
- Each shooter obeys order with high probability.
- The prisoner is dead if any of the shooters shoot.
- One counterfactual is “if the second shooter shot, what would have happened if the second shooter had not shot?”
- Another counterfactual query is “if the prisoner died; what would have happened if shooter 2 had not shot”.

$E=e$  is observed, the query is “what if  $C=c$  happened?”

1. Determine what must be true for  $E=e$  to be observed. This is an instance of **abduction**.
2. Intervene to make  $C=c$  true.
3. Query the resulting model, using the posterior probabilities from the first step as the prior for the intervened model.

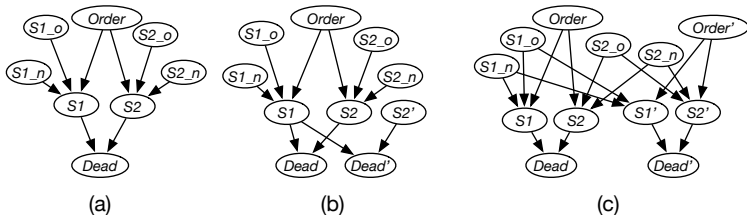
This can be implemented by constructing a causal network, from which queries from the counterfactual situation can be made.

# Counterfactual Causal Network

To model observing  $E=e$ , and asking “what if  $C=c$  happened”:

- represent the problem using a causal network where conditional probabilities are in terms of a **deterministic system with stochastic inputs**, such as a **probabilistic logic program** or a **probabilistic program**
- create a node  $C'$  (a primed variable ) with the same domain as  $C$  but with no parents
- for each descendant  $D$  of  $C$  in the original model, create a node  $D'$
- The conditional probability for  $D'$  is the same as for  $D$ , but using primed parents that exist.
- Condition on  $C'=c$
- Condition on the observations of the initial situation using unprimed variables.

# Example



- (a) original network, e.g.,  $s1 \leftrightarrow (order \wedge s1_o) \vee (\neg order \wedge s1_n)$
- (b) “what if shooter 2 shot” or “what if shooter 2 didn’t shoot”.  
“the prisoner is dead; what is the probability that the prisoner would be dead if the second shooter did not shoot?”:

$$P(dead' \mid dead \wedge \neg s2')$$

- (c) “what if the order was not given” “shooter 1 didn’t shoot and the prisoner was dead; what is the probability the prisoner is dead if the order was not given”:

$$P(dead' \mid \neg s1 \wedge dead \wedge \neg order')$$

