From a Constraint Satisfaction Problem $[\text{Var,Dom,Con}]$ to random variables with probabilities constrained by a graph

- The **domain** (range) of a variable $X$, written $\text{Dom}(X)$, is the set of (possible) values $X$ can take.
From a Constraint Satisfaction Problem [Var, Dom, Con] to random variables with probabilities constrained by a graph

- The domain (range) of a variable $X$, written $\text{Dom}(X)$, is the set of (possible) values $X$ can take.

- A proposition $\alpha$ is an equation $X = x$ between a variable $X$ and a value $x \in \text{Dom}(X)$, or a Boolean combination of such.
From a Constraint Satisfaction Problem [Var,Dom,Con] to random variables with probabilities constrained by a graph

- The domain (range) of a variable $X$, written $\text{Dom}(X)$, is the set of (possible) values $X$ can take.

- A proposition $\alpha$ is an equation $X = x$ between a variable $X$ and a value $x \in \text{Dom}(X)$, or a Boolean combination of such.

- A proposition $\alpha$ is assigned a probability through
  - a notion $\models$ of a possible world $\omega$ satisfying $\alpha$, and
  - a measure $\mu$ for weighing a set of possible worlds.
Satisfaction, measure and probability

Fix a set $\Omega$ of possible worlds $\omega$ that assign a value to each random variable, and interpret a proposition via $\models$

$$\omega \models X = x \iff \omega \text{ assigns } X \text{ the value } x$$

$$\omega \models \alpha \land \beta \iff \omega \models \alpha \text{ and } \omega \models \beta$$

$$\omega \models \alpha \lor \beta \iff \omega \models \alpha \text{ or } \omega \models \beta$$

$$\omega \models \neg \alpha \iff \omega \nmodels \alpha.$$
Satisfaction, measure and probability

Fix a set \( \Omega \) of possible worlds \( \omega \) that assign a value to each random variable, and interpret a proposition via \( \models \):

\[
\omega \models X = x \iff \omega \text{ assigns } X \text{ the value } x
\]
\[
\omega \models \alpha \land \beta \iff \omega \models \alpha \text{ and } \omega \models \beta
\]
\[
\omega \models \alpha \lor \beta \iff \omega \models \alpha \text{ or } \omega \models \beta
\]
\[
\omega \models \neg \alpha \iff \omega \not\models \alpha.
\]

For finite \( \Omega \), a probability measure is a function

\[
\mu : \text{Pow}(\Omega) \to [0, 1]
\]

such that \( \mu(\Omega) = 1 \) and for any subset \( S \) of \( \Omega \),

\[
\mu(S) = \sum_{\omega \in S} \mu(\{\omega\}).
\]
Satisfaction, measure and probability

Fix a set $\Omega$ of possible worlds $\omega$ that assign a value to each random variable, and interpret a proposition via $\models$

$$\omega \models X = x \iff \omega \text{ assigns } X \text{ the value } x$$

$$\omega \models \alpha \land \beta \iff \omega \models \alpha \text{ and } \omega \models \beta$$

$$\omega \models \alpha \lor \beta \iff \omega \models \alpha \text{ or } \omega \models \beta$$

$$\omega \models \neg \alpha \iff \omega \not\models \alpha.$$ 

For finite $\Omega$, a probability measure is a function

$$\mu : \text{Pow}(\Omega) \to [0, 1]$$

such that $\mu(\Omega) = 1$ and for any subset $S$ of $\Omega$,

$$\mu(S) = \sum_{\omega \in S} \mu(\{\omega\}).$$

Given $\mu$, a proposition $\alpha$ has probability

$$P(\alpha) = \mu(\{\omega \mid \omega \models \alpha\}).$$
Tuples, distributions and the sum rule

A tuple \( X_1, \ldots, X_n \) of random variables is a random variable with domain

\[ \text{Dom}(X_1) \times \cdots \times \text{Dom}(X_n). \]
Tuples, distributions and the sum rule

A tuple $X_1, \ldots, X_n$ of random variables is a random variable with domain

$$\text{Dom}(X_1) \times \cdots \times \text{Dom}(X_n).$$

A probability distribution on a random variable $X$ is a function $P_X : \text{Dom}(X) \rightarrow [0, 1]$ s.t.

$$P_X(x) = P(X = x).$$

$P_X$ is often written as $P(X)$, and $P_X(x)$ as $P(x)$. 
A tuple $X_1, \ldots, X_n$ of random variables is a random variable with domain

$$\text{Dom}(X_1) \times \cdots \times \text{Dom}(X_n).$$

A probability distribution on a random variable $X$ is a function $P_X : \text{Dom}(X) \to [0, 1]$ s.t.

$$P_X(x) = P(X = x).$$

$P_X$ is often written as $P(X)$, and $P_X(x)$ as $P(x)$.

**Sum rule**

$$P(X) = \sum_Y P(X, Y)$$

$$P_X(x) = \sum_{y \in \text{Dom}(Y)} P_{X,Y}(x, y) \quad \text{for} \ x \in \text{Dom}(X).$$
Conditional probability

To incorporate a proposition $\alpha$ into the background assumptions, we restrict the set $\Omega$ of possible worlds to

$$\Omega \upharpoonright \alpha := \{\omega \in \Omega \mid \omega \models \alpha\}$$

and assuming $\mu(\Omega \upharpoonright \alpha) \neq 0$, map a subset $S \subseteq \Omega \upharpoonright \alpha$ to

$$\mu^\alpha(S) := \frac{\mu(S)}{\mu(\Omega \upharpoonright \alpha)}$$

for a probability measure $\mu^\alpha : \text{Pow}(\Omega \upharpoonright \alpha) \to [0, 1]$ on $\Omega \upharpoonright \alpha$. 
Conditional probability

To incorporate a proposition \( \alpha \) into the background assumptions, we restrict the set \( \Omega \) of possible worlds to

\[
\Omega \upharpoonright \alpha := \{ \omega \in \Omega \mid \omega \models \alpha \}
\]

and assuming \( \mu(\Omega \upharpoonright \alpha) \neq 0 \), map a subset \( S \subseteq \Omega \upharpoonright \alpha \) to

\[
\mu^\alpha(S) := \frac{\mu(S)}{\mu(\Omega \upharpoonright \alpha)}
\]

for a probability measure \( \mu^\alpha : \text{Pow}(\Omega \upharpoonright \alpha) \to [0, 1] \) on \( \Omega \upharpoonright \alpha \).

The conditional probability of \( \alpha' \) given \( \alpha \) is

\[
P(\alpha' \mid \alpha) := \mu^\alpha(\Omega \upharpoonright \alpha' \land \alpha) = \frac{P(\alpha' \land \alpha)}{P(\alpha)}
\]
The product rule and Bayes’ theorem

product rule

\[ P(X, Y) = P(X|Y)P(Y) \]
\[ P_{X,Y}(x, y) = P_X(x|Y = y)P_Y(y) \]

for \( x \in \text{Dom}(X), \ y \in \text{Dom}(Y) \)
The product rule and Bayes’ theorem

**product rule**  
\[ P(X, Y) = P(X|Y)P(Y) \]  
\[ P_{X,Y}(x, y) = P_X(x|Y = y)P_Y(y) \]  
for \( x \in \text{Dom}(X), \ y \in \text{Dom}(Y) \)

As conjunction is commutative (\( \Omega \upharpoonright \alpha' \land \alpha = \Omega \upharpoonright \alpha \land \alpha' \)),  
\[ P(X, Y) = P(Y, X) \]
and so the product rule yields

**Bayes’ theorem**  
\[ P(X|Y) = \frac{P(Y|X)P(X)}{P(Y)} \]  
if \( P(Y) \neq 0 \)
The product rule and Bayes’ theorem

**Product rule**

\[ P(X, Y) = P(X|Y)P(Y) \]

\[ P_{X,Y}(x,y) = P_X(x|Y=y)P_Y(y) \]

for \( x \in \text{Dom}(X), \ y \in \text{Dom}(Y) \)

As conjunction is commutative \((\Omega \upharpoonright \alpha' \land \alpha = \Omega \upharpoonright \alpha \land \alpha')\),

\[ P(X, Y) = P(Y, X) \]

and so the product rule yields

**Bayes’ theorem**

\[ P(X|Y) = \frac{P(Y|X)P(X)}{P(Y)} \text{ if } P(Y) \neq 0 \]

The prior probability of \( \alpha \)

\[ P(\alpha) = \mu(\Omega \upharpoonright \alpha) \]

is updated by \( \alpha_\circ \) to the posterior probability given \( \alpha_\circ \)

\[ P(\alpha | \alpha_\circ) = \mu^{\alpha_\circ}(\Omega \upharpoonright (\alpha \land \alpha_\circ)) \]
Why is Bayes’ theorem interesting?

Form a hypothesis $h$ given evidence $e$ with $P(e) \neq 0$ via Bayes

$$P(h|e) = \frac{P(e|h)P(h)}{P(e)}.$$
Why is Bayes’ theorem interesting?

Form a hypothesis $h$ given evidence $e$ with $P(e) \neq 0$ via Bayes

$$P(h|e) = \frac{P(e|h)P(h)}{P(e)}.$$ 

We often have causal knowledge

$$P(\text{symptom} \mid \text{disease}), \quad P(\text{alarm} \mid \text{fire})$$

$$P(\text{image} = \text{tree} \mid \text{a tree is in front of a car})$$

but want to do evidential reasoning

$$P(\text{disease} \mid \text{symptom}), \quad P(\text{fire} \mid \text{alarm})$$

$$P(\text{a tree is in front of a car} \mid \text{image} = \text{tree})$$
Tuples and the chain rule

Recall: a tuple $X_1, \ldots, X_n$ of random variables is a random variable.
Let us write

$$X_{1:n} \text{ for } X_1, \ldots, X_n$$
Recall: a tuple $X_1, \ldots, X_n$ of random variables is a random variable.

Let us write $X_{1:n}$ for $X_1, \ldots, X_n$

and apply the product rule repeatedly for

$$P(X_{1:n}) = P(X_n \mid X_{1:n-1})P(X_{1:n-1})$$
$$= P(X_n \mid X_{1:n-1})P(X_{n-1} \mid X_{1:n-2})P(X_{1:n-2})$$
$$= \ldots$$
$$= \prod_{i=1}^{n} P(X_i \mid X_{1:i-1}) \quad \text{chain rule}$$

with $X_{1:0}$ as the empty tuple and $P(X_1 \mid X_{1:0}) = P(X_1)$.
Simplifying the chain rule via conditional independence

Choose a sub-tuple $\text{parents}(X_i)$ of $X_{1:i-1}$ such that

$$P(X_i \mid X_{1:i-1}) = P(X_i \mid \text{parents}(X_i)) .$$
Simplifying the chain rule via conditional independence

Choose a sub-tuple $\text{parents}(X_i)$ of $X_{1:i-1}$ such that

$$P(X_i \mid X_{1:i-1}) = P(X_i \mid \text{parents}(X_i)) .$$

$X$ is independent of $Y$ given $Z$, written $X \perp \perp Y \mid Z$,

$$P(X \mid Y, Z) = P(X \mid Z)$$
Simplifying the chain rule via conditional independence

Choose a sub-tuple $\text{parents}(X_i)$ of $X_{1:i-1}$ such that

$$P(X_i \mid X_{1:i-1}) = P(X_i \mid \text{parents}(X_i)) .$$

$X$ is independent of $Y$ given $Z$, written $X \perp \perp Y \mid Z$,

$$P(X \mid Y, Z) = P(X \mid Z)$$

i.e. for all $x \in \text{Dom}(X)$, $y \in \text{Dom}(Y)$, and $z \in \text{Dom}(Z)$,

$$P(X = x \mid Y = y \land Z = z) = P(X = x \mid Z = z)$$

— knowing $Y$’s value says nothing about $X$’s value, given $Z$’s value.
Simplifying the chain rule via conditional independence

Choose a sub-tuple $\text{parents}(X_i)$ of $X_{1:i-1}$ such that

$$P(X_i \mid X_{1:i-1}) = P(X_i \mid \text{parents}(X_i))$$

$X$ is independent of $Y$ given $Z$, written $X \perp \perp Y \mid Z$,

$$P(X \mid Y, Z) = P(X \mid Z)$$

i.e. for all $x \in \text{Dom}(X)$, $y \in \text{Dom}(Y)$, and $z \in \text{Dom}(Z)$,

$$P(X = x \mid Y = y \land Z = z) = P(X = x \mid Z = z)$$

— knowing $Y$’s value says nothing about $X$’s value, given $Z$’s value.

Note

$$X \perp \perp Y \mid Z \iff P(X, Y \mid Z) = P(X \mid Z)P(Y \mid Z)$$

$$\iff Y \perp \perp X \mid Z$$
Totally order the variables of interest

\[ X_1 < X_2 < \cdots < X_n \]

and for each \( i \) from 1 to \( n \), choose \( \text{parents}(X_i) \) from \( X_{1:i-1} \) s.t.

\[
P(X_i \mid X_{1:i-1}) = P(X_i \mid \text{parents}(X_i)) \quad (\dagger)
\]
Belief networks

Totally order the variables of interest

\[ X_1 < X_2 < \cdots < X_n \]

and for each \( i \) from 1 to \( n \), choose \( \text{parents}(X_i) \) from \( X_{1:i-1} \) s.t.

\[ P(X_i \mid X_{1:i-1}) = P(X_i \mid \text{parents}(X_i)) \quad (\dagger) \]

A belief network consists of:
- a directed acyclic graph with nodes = random variables, and an arc from the parents of each node into that node
- a domain for each random variable
- conditional probability tables for each variable given its parents (for a probability distribution respecting \((\dagger)\))
Example: Markov chain

A Markov chain is a special sort of belief network:

What probabilities need to be specified?
Example: Markov chain

A Markov chain is a special sort of belief network:

![Diagram of Markov chain]

What probabilities need to be specified?

- $P(S_0)$ specifies initial conditions
- $P(S_{t+1}|S_t)$ specifies the dynamics
Example: Markov chain

A Markov chain is a special sort of belief network:

What probabilities need to be specified?
- $P(S_0)$ specifies initial conditions
- $P(S_{t+1}|S_t)$ specifies the dynamics

What independence assumptions are made?

$$P(S_{t+1}|S_{0:t}) = P(S_{t+1}|S_t)$$

$S_t$ represents the state at time $t$, capturing everything about the past ($< t$) that can affect the future ($> t$)

The future is independent of the past given the present.
Two elaborations

In a stationary Markov chain,

\[ \text{Dom}(S_i) = \text{Dom}(S_0) \text{ and } P(S_{i+1}|S_i) = P(S_1|S_0) \text{ for all } i \geq 0 \]

so it is enough to specify \( P(S_0) \) and \( P(S_1|S_0) \).

- Simple model, easy to specify
- The network can extend indefinitely
Two elaborations

In a stationary Markov chain,

\[ \text{Dom}(S_i) = \text{Dom}(S_0) \text{ and } P(S_{i+1}|S_i) = P(S_1|S_0) \text{ for all } i \geq 0 \]

so it is enough to specify \( P(S_0) \) and \( P(S_1|S_0) \).

- Simple model, easy to specify
- The network can extend indefinitely

A Hidden Markov Model (HMM) is a belief network of the form

\[ S_0 \rightarrow S_1 \rightarrow S_2 \rightarrow S_3 \rightarrow S_4 \]

\[ O_0 \rightarrow O_1 \rightarrow O_2 \rightarrow O_3 \rightarrow O_4 \]

- \( P(S_0) \) specifies initial conditions
- \( P(S_{i+1}|S_i) \) specifies the dynamics
- \( P(O_i|S_i) \) specifies the sensor model
Naive Bayes Classifier

Problem: classify on the basis of features $F_i$

$$P(\text{Class}|F_{1:n}) = \frac{P(F_{1:n}|\text{Class})P(\text{Class})}{P(F_{1:n})}$$

Assume $F_i$ are independent of each other given Class

Assume the values of features $F_i$ are predictable given a class.

Requires $P(\text{Class})$ and $P(F_i|\text{Class})$ for each $F_i$
Naive Bayes Classifier

Problem: classify on the basis of features $F_i$

$$P(Class|F_{1:n}) = \frac{P(F_{1:n}|Class)P(Class)}{P(F_{1:n})}$$

Assume $F_i$ are independent of each other given $Class$

$$P(F_{1:n}|Class) = \prod_{i} P(F_i|Class)$$
**Naive Bayes Classifier**

Problem: classify on the basis of features $F_i$

\[
P(\text{Class}|F_{1:n}) = \frac{P(F_{1:n}|\text{Class})P(\text{Class})}{P(F_{1:n})}
\]

Assume $F_i$ are independent of each other given Class

\[
P(F_{1:n}|\text{Class}) = \prod_i P(F_i|\text{Class})
\]

Assume the values of features $F_i$ are predictable given a class.

Requires $P(\text{Class})$ and $P(F_i|\text{Class})$ for each $F_i$
Learning Probabilities

<table>
<thead>
<tr>
<th>$F_1$</th>
<th>$F_2$</th>
<th>$F_3$</th>
<th>$F_4$</th>
<th>$C$</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>:</td>
<td>:</td>
<td>:</td>
<td>:</td>
<td>:</td>
<td>:</td>
</tr>
<tr>
<td>$t$</td>
<td>$f$</td>
<td>$t$</td>
<td>$t$</td>
<td>1</td>
<td>40</td>
</tr>
<tr>
<td>$t$</td>
<td>$f$</td>
<td>$t$</td>
<td>$t$</td>
<td>2</td>
<td>10</td>
</tr>
<tr>
<td>$t$</td>
<td>$f$</td>
<td>$t$</td>
<td>$t$</td>
<td>3</td>
<td>50</td>
</tr>
<tr>
<td>:</td>
<td>:</td>
<td>:</td>
<td>:</td>
<td>:</td>
<td>:</td>
</tr>
</tbody>
</table>
Learning Probabilities

<table>
<thead>
<tr>
<th>$F_1$</th>
<th>$F_2$</th>
<th>$F_3$</th>
<th>$F_4$</th>
<th>$C$</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>:</td>
<td>:</td>
<td>:</td>
<td>:</td>
<td>:</td>
<td>:</td>
</tr>
<tr>
<td>$t$</td>
<td>$f$</td>
<td>$t$</td>
<td>$t$</td>
<td>$t$</td>
<td>1</td>
</tr>
<tr>
<td>$t$</td>
<td>$f$</td>
<td>$t$</td>
<td>$t$</td>
<td>$t$</td>
<td>2</td>
</tr>
<tr>
<td>$t$</td>
<td>$f$</td>
<td>$t$</td>
<td>$t$</td>
<td>$t$</td>
<td>3</td>
</tr>
<tr>
<td>:</td>
<td>:</td>
<td>:</td>
<td>:</td>
<td>:</td>
<td>:</td>
</tr>
</tbody>
</table>

$P(C=c) = \frac{\sum_{\omega \models C=c} \text{Count}(\omega)}{\sum_{\omega} \text{Count}(\omega)}$

$P(F_k = b | C=c) = \frac{\sum_{\omega \models C=c \land F_k=b} \text{Count}(\omega)}{\sum_{\omega \models C=c} \text{Count}(\omega)}$

with pseudo-counts (Cromwell’s rule)
The domain of $H$ is the set of all help pages. The observations are the words in the query.

What probabilities are needed? What pseudo-counts and counts are used? What data can be used to learn from?
Constructing a belief network

To represent a domain in a belief network, we need to consider:

- What are the relevant variables?
  - What will you observe?
  - What would you like to find out (query)?
  - What other features make the model simpler?

- What values should these variables take?

- What is the relationship between them?
  Express this in terms of a directed graph, representing how each variable $X_i$ is generated from its predecessors $X_{1:i−1}$.
  
  The parents of $X$ are variables on which $X$ directly depends
  - $X$ is independent of its non-descendants given its parents.

- How does the value of each variable depend on its parents?
  This is expressed in terms of the conditional probabilities.
Example: fire alarm belief network

Variables:

- **Fire**: there is a fire in the building
- **Tampering**: someone has been tampering with the fire alarm
- **Smoke**: what appears to be smoke is coming from an upstairs window
- **Alarm**: the fire alarm goes off
- **Leaving**: people are leaving the building *en masse*.
- **Report**: a colleague says that people are leaving the building *en masse*. (A noisy sensor for leaving.)
Head-to-tail: Chain

- alarm and report are dependent

\[
P(\text{report, alarm} | \text{leaving}) = P(\text{report, alarm, leaving}) \frac{P(\text{leaving})}{P(\text{leaving})} = P(\text{alarm}) P(\text{leaving} | \text{alarm}) P(\text{report} | \text{leaving});
\]

\[
P(\text{alarm, leaving}) P(\text{leaving}) P(\text{report} | \text{leaving}) \text{ for } \perp \perp
\]
Head-to-tail: Chain

- \( \text{alarm} \) and \( \text{report} \) are dependent

\[
P(\text{report, alarm} | \text{leaving}) = P(\text{report, alarm, leaving}) P(\text{leaving}) = P(\text{alarm}) P(\text{leaving} | \text{alarm}) P(\text{report} | \text{leaving}) P(\text{leaving}) 
\]

\[
= P(\text{alarm, leaving}) P(\text{leaving}) P(\text{report} | \text{leaving}) \text{ product for } \perp \perp
\]
Head-to-tail: Chain

- alarm and report are dependent
- alarm and report are given

\[ P(\text{report}, \text{alarm} | \text{leaving}) = P(\text{report}, \text{alarm}, \text{leaving}) \]
\[ = P(\text{alarm}) \cdot P(\text{leaving} | \text{alarm}) \cdot P(\text{report} | \text{leaving}) \]
\[ = P(\text{alarm}, \text{leaving}) \cdot P(\text{leaving}) \cdot P(\text{report} | \text{leaving}) \]
Head-to-tail: Chain

- alarm and report are dependent
- alarm and report are independent given leaving

Intuitively, the only way that the alarm affects report is by affecting leaving.

\[
P(\text{report, alarm} | \text{leaving}) = P(\text{report, alarm, leaving}) \frac{P(\text{leaving})}{P(\text{leaving})} = P(\text{alarm, leaving}) \frac{P(\text{report} | \text{leaving}) P(\text{leaving})}{P(\text{leaving})} = P(\text{alarm} | \text{leaving}) P(\text{report} | \text{leaving})
\]
Head-to-tail: Chain

- *alarm* and *report* are dependent
- *alarm* and *report* are independent given *leaving*
- Intuitively, the only way that the *alarm* affects *report* is by affecting *leaving.*
Head-to-tail: Chain

- *alarm* and *report* are dependent
- *alarm* and *report* are independent given *leaving*
- Intuitively, the only way that the *alarm* affects *report* is by affecting *leaving*.

\[
P(\text{report, alarm} \mid \text{leaving}) = \frac{P(\text{report, alarm, leaving})}{P(\text{leaving})}
\]

\[
= \frac{P(\text{alarm})P(\text{leaving} \mid \text{alarm})P(\text{report} \mid \text{leaving})}{P(\text{leaving})} \quad \text{net}
\]

\[
= \frac{P(\text{alarm, leaving})}{P(\text{leaving})}P(\text{report} \mid \text{leaving}) \quad \text{product}
\]

\[
= P(\text{alarm} \mid \text{leaving})P(\text{report} \mid \text{leaving}) \quad \text{for } \bot
\]
Tail-to-tail: Common ancestors

- alarm and smoke are dependent given fire. Intuitively, fire can explain alarm and smoke; learning one can affect the other by changing your belief in fire.

\[
P(smoke, alarm \mid fire) = \frac{P(smoke, alarm, fire)}{P(fire)} = P(alarm \mid fire)P(smoke \mid fire)P(fire) \text{ net}
\]
Tail-to-tail: Common ancestors

- *alarm* and *smoke* are dependent

\[
P(smoke, alarm | fire) = P(smoke, alarm, fire)
\]
\[
P(smoke, alarm, fire) = P(fire) P(alarm | fire) P(smoke | fire) P(fire)
\]
\[
P(smoke, alarm, fire) = P(alarm | fire) P(smoke | fire) P(fire)
\]
Tail-to-tail: Common ancestors

- *alarm* and *smoke* are dependent
- *alarm* and *smoke* are independent given *fire*

![Diagram](image-url)
Tail-to-tail: Common ancestors

- \( \text{alarm} \) and \( \text{smoke} \) are dependent
- \( \text{alarm} \) and \( \text{smoke} \) are independent given \( \text{fire} \)

\[
\Pr(\text{smoke, alarm} | \text{fire}) = \Pr(\text{smoke, alarm}, \text{fire}) \frac{\Pr(\text{fire})}{\Pr(\text{fire})} = \Pr(\text{alarm} | \text{fire}) \Pr(\text{smoke} | \text{fire}) \]

for \( \text{alarm} \) and \( \text{smoke} \) are independent given \( \text{fire} \).
Tail-to-tail: Common ancestors

- alarm and smoke are dependent
- alarm and smoke are independent given fire
- Intuitively, fire can explain alarm and smoke; learning one can affect the other by changing your belief in fire.
Tail-to-tail: Common ancestors

- \textit{alarm} and \textit{smoke} are dependent
- \textit{alarm} and \textit{smoke} are independent given \textit{fire}
- Intuitively, \textit{fire} can explain \textit{alarm} and \textit{smoke}; learning one can affect the other by changing your belief in \textit{fire}.

\begin{align*}
\text{smoke} \perp \perp \text{alarm} \mid \text{fire} \\

P(\text{smoke, alarm} \mid \text{fire}) &= \frac{P(\text{smoke, alarm, fire})}{P(\text{fire})} \\
&= \frac{P(\text{fire})P(\text{alarm} \mid \text{fire})P(\text{smoke} \mid \text{fire})}{P(\text{fire})} \\
&= P(\text{alarm} \mid \text{fire})P(\text{smoke} \mid \text{fire}) \quad \text{for} \quad \perp \perp
\end{align*}
Head-to-head: Common descendants

- tampering
- fire
- alarm

*tampering and fire are*

\[
P(f_i = 1 | a_m = 1) > P(f_i = 1 | a_m = 1 \land t_g = 1)
\]

\[
P(t_g = 0) = 0.9
\]

\[
P(f_i = 0) = 0.95
\]

\[
P(a_m = 1 | t_g = 1 \land f_i = 1) = 0.9
\]

\[
P(a_m = 1 | t_g = 0 \land f_i = 1) = 0.1
\]
Head-to-head: Common descendants

- tampering
- fire
- alarm

- tampering and fire are independent

Intuitively, tampering can explain away fire.

\[
P(f_i = 1 | a_m = 1) > P(f_i = 1 | a_m = 1 \land t_g = 1)
\] for 

\[
P(t_g = 0) = 0.
\]

\[
P(f_i = 0) = 0.
\]

\[
P(a_m = 1 | t_g = 1 \land f_i = 1) = 0.
\]

\[
P(a_m = 1 | t_g = 1 \land f_i = 0) = 0.
\]

\[
P(a_m = 1 | t_g = 0 \land f_i = 1) = 0.
\]

\[
P(a_m = 1 | t_g = 0 \land f_i = 0) = 0.
\]
Head-to-head: Common descendants

- tampering and fire are independent
- tampering and fire are given alarm

\[ P(\text{fi} = 1 | \text{am} = 1) > P(\text{fi} = 1 | \text{am} = 1 \land \text{tg} = 1) \]

for \( P(\text{tg} = 0) = 0 \).

\[ P(\text{am} = 1 | \text{tg} = 1 \land \text{fi} = 1) = 0 \]

\[ P(\text{am} = 1 | \text{tg} = 1 \land \text{fi} = 0) = 0 \]

\[ P(\text{am} = 1 | \text{tg} = 0 \land \text{fi} = 1) = 0 \]

\[ P(\text{am} = 1 | \text{tg} = 0 \land \text{fi} = 0) = 0 \]
Head-to-head: Common descendants

- tampering and fire are independent
- tampering and fire are dependent given alarm
Head-to-head: Common descendants

- tampering and fire are independent
- tampering and fire are dependent given alarm
- Intuitively, tampering can explain away fire
Head-to-head: Common descendants

- **tampering** and **fire** are independent
- **tampering** and **fire** are dependent given **alarm**
- Intuitively, **tampering** can explain away **fire**

\[
P(fi = 1 \mid am = 1) > P(fi = 1 \mid am = 1 \land tg = 1)
\]

for

\[
\begin{align*}
P(tg = 0) &= 0.9 & P(fi = 0) &= 0.9 \\
P(am = 1 \mid tg = 1 \land fi = 1) &= 0.95 \\
P(am = 1 \mid tg = 1 \land fi = 0) &= 0.5 \\
P(am = 1 \mid tg = 0 \land fi = 1) &= 0.9 \\
P(am = 1 \mid tg = 0 \land fi = 0) &= 0.1 
\end{align*}
\]
$P(fi = 1|am = 1) \approx 0.418$

$$P(fi = 1|am = 1) = \frac{P(am = 1|fi = 1)P(fi = 1)}{P(am = 1)} \quad \text{Bayes}$$

$$P(am = 1|fi = 1) = \sum_{tg} P(am = 1, tg|fi = 1) \quad \text{sum}$$

$$P(am = 1|tg, fi = 1)P(tg|fi = 1) \quad \text{product}$$

$$P(tg) \quad \text{net}$$

$$P(am = 1) = \sum_{tg}\sum_{fi} P(am = 1, tg, fi) \quad \text{sum}$$

$$P(tg)P(fi)P(am = 1|tg, fi) \quad \text{net}$$
\[ P(fi = 1 | am = 1, tg = 1) \approx 0.174 \]

\[
P(fi = 1 | am = 1, tg = 1) = \frac{P(am = 1 | fi = 1, tg = 1) P(fi = 1 | tg = 1)}{P(am = 1 | tg = 1)}
\]

Bayes

\[
P(am = 1 | tg = 1) = \sum_{fi} P(am = 1, fi | tg = 1)
\]

sum

\[
P(am = 1 | fi, tg = 1) P(fi | tg = 1) \text{ product}
\]

\[
P(fi) \text{ net}
\]
Conditional independence via d-separation

Given disjoint sets $A, B, C$ of nodes (variables).

When are the variables in $A$ independent of those in $B$ given $C$?

When $A$ is \textit{d-separated} from $B$ by $C$ — i.e., all paths from $A$ to $B$ are $C$-blocked, where

a path from $A$ to $B$ is \textit{C-blocked} if it has a node $Z$ s.t.

(i) $Z$ is in $C$, and the arrows on the path meet head-to-tail or tail-to-tail at $Z$

or

(ii) neither $Z$ nor any of its descendants are in $C$, and the arrows on the path meet head-to-head at $Z$. 

Fact. If $A$ is d-separated from $B$ by $C$, then the variables in $A$ are independent of those in $B$ given $C$ (for all network probabilities).

We can make the net undirected with disconnected $\Rightarrow$ d-separated.
Conditional independence via d-separation

Given disjoint sets $A$, $B$, $C$ of nodes (variables).

When are the variables in $A$ independent of those in $B$ given $C$?

When $A$ is **d-separated** from $B$ by $C$ — i.e., all paths from $A$ to $B$ are $C$-blocked, where

a path from $A$ to $B$ is **$C$-blocked** if it has a node $Z$ s.t.

(i) $Z$ is in $C$, and the arrows on the path meet head-to-tail or tail-to-tail at $Z$

or

(ii) neither $Z$ nor any of its descendants are in $C$, and the arrows on the path meet head-to-head at $Z$.

**Fact.** If $A$ is d-separated from $B$ by $C$, then the variables in $A$ are independent of those in $B$ given $C$ (for all network probabilities).

We can make the net undirected with disconnected $\equiv$ d-separate.
Understanding conditional independence

From non-implications

\[ A \perp \perp B \iff A \perp \perp B \mid C \] (head-to-head)

\[ A \perp \perp B \mid C \iff A \perp \perp B \] (head-to-tail, tail-to-tail)

to

**Graphoid axioms** (Pearl & Paz)

\[ A \perp \perp B \mid C \] as “C intercepts all paths from A to B”

(i) \[ A \perp \perp B \mid C \text{ implies } B \perp \perp A \mid C \]

(ii) \[ A \perp \perp B, B' \mid C \text{ implies } A \perp \perp B \mid C \]

(iii) \[ A \perp \perp B, B' \mid C \text{ implies } A \perp \perp B \mid B', C \]

(iv) \[ A \perp \perp B \mid B', C \text{ and } A \perp \perp B' \mid C \text{ implies } A \perp \perp B, B' \mid C \]

(v) \[ A \perp \perp B \mid B', C \text{ and } A \perp \perp B' \mid B, C \text{ implies } A \perp \perp B, B' \mid C \]