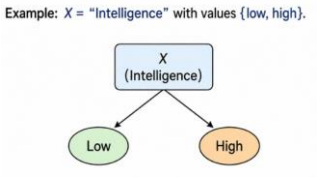
THE BAYESIAN NETWORK REPRESENTATION

Complete Tutorial with Verified Numericals — based on Koller & Friedman, Ch. 3

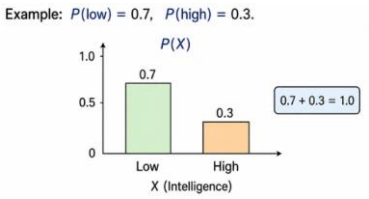
0. Preliminary Definitions

Before building compact representations, a few basic terms need to be pinned down precisely.

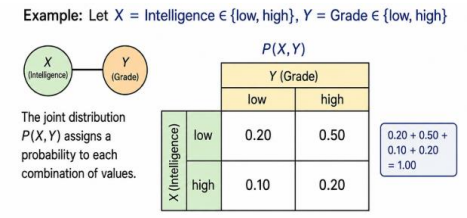
Random variable. A variable X that takes one of a set of possible values, each with some probability. Example: X = "Intelligence" with values {low, high}.



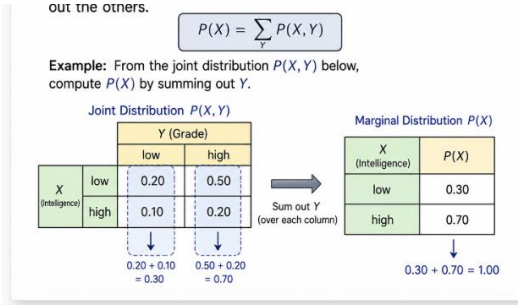
Probability distribution. A function P(X) assigning a probability to each possible value of X, with all probabilities summing to 1. Example: P(low) = 0.7, P(high) = 0.3.



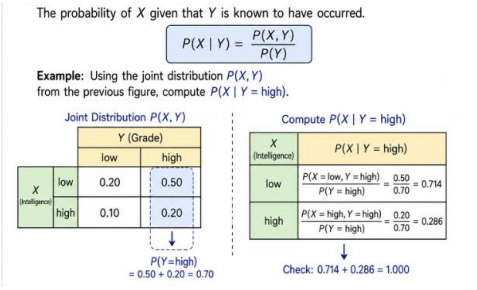
Joint distribution. A distribution $P(X_1, X_2, \ldots, X_n)$ over several variables simultaneously, assigning a probability to every combination of their values. This is the object that becomes unmanageable as n grows — it's the "2^n-1 numbers" to specify in full..



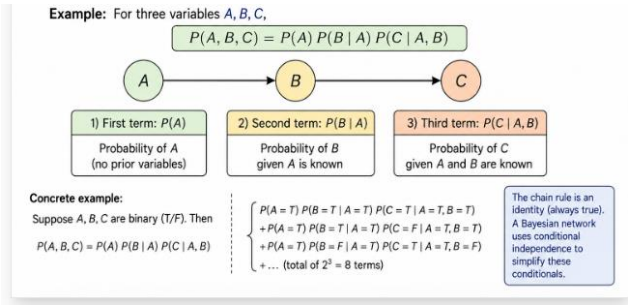
Marginalization. Recovering the distribution over a subset of variables by summing out the rest:
P(X) = Σ_Y P(X, Y)



Conditional probability. The probability of X given that Y is known to have occurred:
P(X | Y) = P(X, Y) / P(Y)



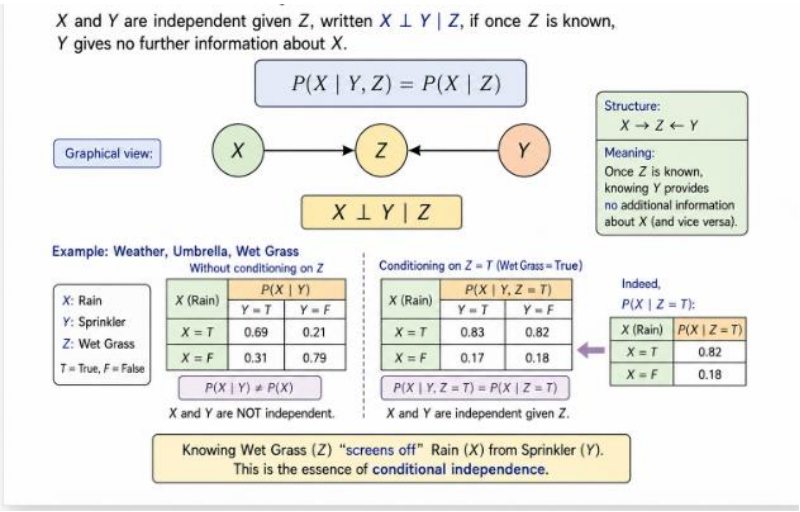
Chain rule of probability. Any joint distribution can always be decomposed into a product of conditionals, in any order — this holds unconditionally, with no assumptions:
$P(X_1, \ldots, X_n) = P(X_1)\, P(X_2|X_1)\, P(X_3|X_1,X_2) \cdots P(X_n|X_1,\ldots,X_{n-1})$



This is the identity Bayesian networks exploit — the *representation* problem is that, written this way, the last few conditionals still need exponentially many parameters. The network's job is to prove that many of these conditionals collapse to a much smaller set of variables (the parents), not all preceding variables.

Independence. X and Y are independent, written ($X \perp Y$), if knowing Y tells you nothing about X: $P(X|Y) = P(X)$, equivalently $P(X,Y) = P(X)P(Y)$.

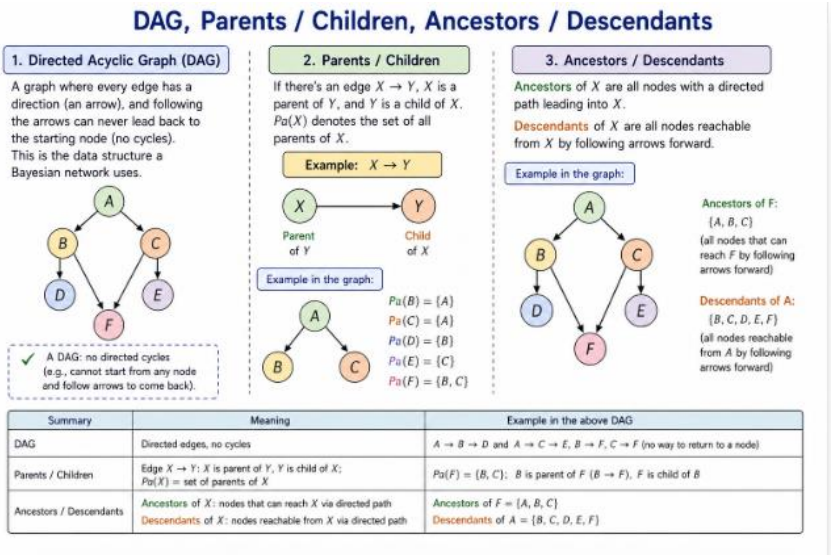
Conditional independence. X and Y are independent *given* Z, written ($X \perp Y \mid Z$), if once Z is known, Y gives no further information about X: $P(X|Y,Z) = P(X|Z)$. This is the central concept the whole chapter is built on — plain independence is rare in real data, but conditional independence is common and is what a Bayesian network's graph structure encodes.



Directed Acyclic Graph (DAG). A graph where every edge has a direction (an arrow), and following the arrows can never lead back to the starting node (no cycles). This is the data structure a Bayesian network uses.

Parents / children. If there's an edge $X \to Y$, X is a parent of Y, and Y is a child of X. Pa(X) denotes the set of all parents of X.

Ancestors / descendants. Ancestors of X are all nodes with a directed path leading into X; descendants are all nodes reachable from X by following arrows forward.



1. The Problem: Why We Need a Compact Representation

A joint distribution over n binary random variables needs 2^n-1 numbers to specify fully — every possible combination of values. For even modest n, this is computationally unmanageable, cognitively impossible to elicit from an expert, and statistically hopeless to learn from limited data. Bayesian networks solve this by exploiting **independence structure** in the distribution, using a directed acyclic graph (DAG) as the data structure that captures which variables directly affect which.

2. Exploiting Independence — From Simple to Naive Bayes

2.1 Fully independent variables

If $X_1,\ldots,X_n$ are marginally independent (e.g. n separate coin tosses), the joint factorizes trivially:

$$P(X_1,\ldots,X_n) = P(X_1)P(X_2)\cdots P(X_n)$$

With $\theta_i = P(X_i = \text{heads})$, only **n independent parameters** are needed instead of 2^n-1 — an exponential saving. But most real variables are not marginally independent, so this simple case rarely applies directly.

2.2 The conditional parameterization — Intelligence and SAT

Consider two variables: Intelligence (I: i_0 low, i_1 high) and SAT score (S: s_0 low, s_1 high). Using the chain rule of probability:

$$P(I,S) = P(I)\,P(S \mid I)$$

This is represented as a tiny Bayesian network I → S, with a prior P(I) and a conditional probability distribution (CPD) P(S|I):

P(I)	**i_0**	**i_1**
	0.7	0.3

P(S|I)	**s_0**	**s_1**
i_0	0.95	0.05
i_1	0.2	0.8

Reading the CPD: a low-intelligence student is very unlikely to score high on the SAT ($P(s_1|i_0)=0.05$); a high-intelligence student is likely, but not certain, to score high ($P(s_1|i_1)=0.8$).

2.3 The Naïve Bayes model — adding Grade

Add a third variable, Grade ($G \in \{g_1,g_2,g_3\}$ = A,B,C). No marginal independencies hold here — I, S, G are all pairwise correlated. But a **conditional** independence is plausible: once we know a student's intelligence, the SAT score gives no further information about the grade:

$P(S, G \mid I) = P(S|I)\,P(G|I)$ — i.e. $(S \perp G \mid I)$

This factorizes the joint distribution:

$$P(I,S,G) = P(S \mid I)\,P(G \mid I)\,P(I)$$

Using P(I) and P(S|I) as before, plus a new CPD P(G|I):

P(G|I)	**g_1**	**g_2**	**g_3**
i_0	0.20	0.34	0.46

P(G|I)	g_1	g_2	g_3
i_1	0.74	0.17	0.09

Worked numerical: $P(i_1, s_1, g_2) = P(i_1) \times P(s_1 \mid i_1) \times P(g_2 \mid i_1)$

$$P(i_1, s_1, g_2) = P(i_1)P(s_1 \mid i_1)P(g_2 \mid i_1) = 0.3 \times 0.8 \times 0.17 = 0.0408$$

Verified independently: 0.3 × 0.8 × 0.17 = 0.0408. ✓

This example generalizes to the **Naïve Bayes model**: a class variable C with features $X_1,\ldots,X_n$, each conditionally independent given C:

$$P(C, X_1, \ldots, X_n) = P(C)\prod_{i=1}^{n} P(X_i \mid C)$$

This needs only 2n+1 parameters for binary features (linear in n) versus exponential for the full joint. It is used for **classification**: decide the class by comparing posterior ratios,

$$\frac{P(C{=}c_1 \mid x_1, \ldots, x_n)}{P(C{=}c_2 \mid x_1, \ldots, x_n)} = \frac{P(C{=}c_1)}{P(C{=}c_2)}\prod_{i=1}^{n}\frac{P(x_i \mid C{=}c_1)}{P(x_i \mid C{=}c_2)}$$

Historical note (interview-relevant): Naïve Bayes was used in early medical-diagnosis systems (Pathfinder). It works well because classification only needs correct **ranking** of classes, not calibrated probabilities — but it can "double-count" correlated evidence (e.g. hypertension and obesity both indicating heart disease), degrading accuracy as the number of correlated features grows.

3. Bayesian Networks — The General Model

A Bayesian network generalizes Naïve Bayes: instead of one class variable with all features as its children, **any** DAG structure is allowed, tailored to the actual independencies of the domain.

3.1 The Student example (full version)

Five variables: Difficulty (D), Intelligence (I), Grade (G), SAT (S), Letter (L). All binary except G (ternary). The DAG:

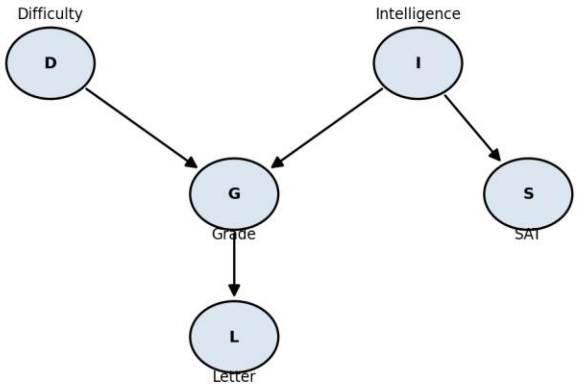


Figure 1. The Student Bayesian network. Each edge represents direct probabilistic influence; a node depends directly only on its parents.

The CPDs (verified, matching the book exactly):

P(I)	i_0	i_1
	0.7	0.3

P(D)	d_0	d_1
	0.6	0.4

P(G|I,D)	g_1	g_2	g_3
i_0,d_0	0.30	0.40	0.30
i_0,d_1	0.05	0.25	0.70
i_1,d_0	0.90	0.08	0.02
i_1,d_1	0.50	0.30	0.20

P(L|G)	l_0	l_1
g_1	0.10	0.90
g_2	0.40	0.60

P(I)	i_0	i_1
g_3	0.99	0.01

The joint distribution is given by the **chain rule for Bayesian networks**:

$$P(I, D, G, S, L) = P(I)P(D)P(G \mid I, D)P(S \mid I)P(L \mid G)$$

Worked numerical (verified against the full 48-entry joint distribution): Probability that a student is intelligent, in an easy class, gets a B, scores high on SAT, and gets a weak letter:

$$P(i_1, d_0, g_2, s_1, l_0) = P(i_1)P(d_0)P(g_2 \mid i_1, d_0)P(s_1 \mid i_1)P(l_0 \mid g_2) = 0.3 \times 0.6 \times 0.08 \times 0.8 \times 0.4 = 0.004608$$

Independently recomputed by summing the full joint: exact match, 0.004608. ✓

3.2 Parameter savings

$$\text{Network: } 1+1+8+2+3 = 15 \quad \text{vs.} \quad \text{Full joint: } 2^1 \times 2^1 \times 3 \times 2^1 \times 2^1 - 1 = 47$$

The general rule: with n binary variables and at most k parents per node, the network needs only about $n \times 2^k$ parameters, versus $2^n - 1$ for the explicit joint — exponentially fewer whenever $k \ll n$.

3.3 Reasoning patterns (verified against the full joint)

All values below were independently recomputed from the CPDs above and match the book exactly:

Query	Value	Reasoning type
$P(l_1)$ — strong letter, no evidence	0.502	baseline
$P(l_1 \mid i_0)$ — given low intelligence	0.389	causal (downstream)
$P(l_1 \mid i_0, d_0)$ — given low intelligence, easy class	0.513	causal
$P(i_1 \mid g_3)$ — intelligence given a C grade	0.079	evidential (upstream)
$P(i_1 \mid g_3, s_1)$ — same, but SAT also high	0.578	intercausal / explaining away
$P(i_1 \mid g_2, d_1)$ vs $P(i_1 \mid g_2)$	0.340 vs 0.175	explaining away

- **Causal reasoning:** predicting downstream effects (knowing I changes belief about L).
- **Evidential reasoning:** reasoning from effect back to cause (a poor grade lowers belief in intelligence).
- **Intercausal reasoning / "explaining away":** two causes of the same effect interact — a hard class "explains away" a poor grade, so belief in low intelligence partly recovers ($P(i_1|g_2)$=0.175 rises to 0.340 once we also learn the class was hard).

4. Formal Semantics — Local Independencies and I-Maps

Every Bayesian network structure encodes a precise set of conditional independence assumptions, called the **local independencies**:

$$(X_i \perp \text{NonDescendants}_{X_i} \mid \text{Pa}^{\mathcal{G}}_{X_i})$$

In words: each variable is independent of its non-descendants, given its parents. For the Student network these are exactly: (D⫫), (S⊥D,G,L | I), (L⫫D,S | G), (G⊥S | I,D).

I-map (independency map): a graph G is an I-map of P if every independency G asserts actually holds in P: I_l (G) ⊆I(P).

$$\mathcal{I}_\ell(\mathcal{G}) \subseteq \mathcal{I}(P) \quad \Longrightarrow \quad \mathcal{G} \text{ is an I-map of } P$$

Two fundamental theorems connect independence and factorization:

- **Theorem 3.1:** if G is an I-map for P, then P factorizes according to G (chain rule holds).
- **Theorem 3.2:** conversely, if P factorizes according to G, then G is an I-map for P.

These two results together show the equivalence of the two views of a Bayesian network: as a scaffold for factorizing P, and as an encoding of independence assumptions about P.

5. D-Separation — Reading Independencies Off the Graph

Local independencies alone don't tell us everything the graph implies. **D-separation** ("directed separation") is the general algorithm for reading **any** independence directly from the graph's structure, via the notion of an **active trail**.

5.1 The four basic two-edge trail patterns

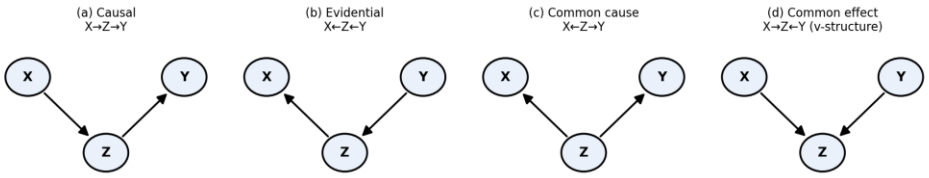


Figure 2. The four possible trails between X and Y through an intermediate node Z.

Trail	Active when Z is…	Example (Student net)
(a) Causal X→Z→Y	NOT observed	I→G→L: I affects L unless G is known
(b) Evidential X←Z←Y	NOT observed	same trail, opposite direction
(c) Common cause X←Z→Y	NOT observed	S←I→G: S,G correlated unless I known
(d) Common effect (v-structure) X→Z←Y	OBSERVED (or a descendant is)	I→G←D: I,D independent unless G (or L) known

The one surprising case: a v-structure (common effect) behaves **oppositely** to the other three — it is active only when the middle node (or a descendant of it) **is** observed. This is exactly the mechanism behind "explaining away" in Section 3.3: observing G activates the v-structure I→G←D, correlating I and D.

5.2 General definition and the algorithm

A trail is active given evidence set Z if every v-structure on it has its middle node (or a descendant) in Z, and no other node on the trail is in Z. X and Y are **d-separated** given Z if **no** active trail connects them:

$$\text{d-sep}_{\mathcal{G}}(X, Y \mid Z) \iff \text{no active trail between } X, Y \text{ given } Z$$

$$\mathcal{I}(\mathcal{G}) = \{(X \perp Y \mid Z) : \text{d-sep}_{\mathcal{G}}(X, Y \mid Z)\}$$

A linear-time algorithm (two-phase breadth-first search: mark ancestors of Z, then traverse active trails from X) computes d-separation efficiently, avoiding the exponential cost of enumerating every trail.

5.3 Worked d-separation example

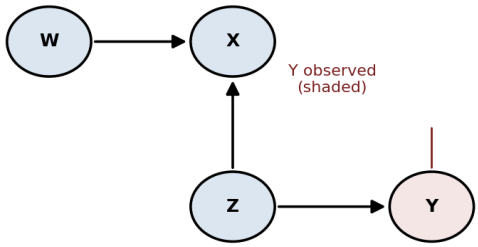


Figure 3. Is X d-separated from Y given {Y is observed}? Trail W→X←Z→Y: the segment X←Z→Y is a common-cause trail through Z (unobserved) — active. But X and Y share no direct trail once we check X←Z→Y specifically passes through Z, which is NOT in the evidence set — so this trail is active, and X, Y are NOT d-separated.

This mirrors the book's own worked example (Figure 3.6): careful bookkeeping of "visited from top" vs. "visited from bottom" is required at nodes that lie on multiple trails, which is exactly what the two-phase algorithm handles correctly.

5.4 Soundness and completeness

$$\text{If } P \text{ factorizes over } \mathcal{G}, \text{ then } \mathcal{I}(\mathcal{G}) \subseteq \mathcal{I}(P) \quad \text{(Soundness, Thm 3.3)}$$

Soundness (Theorem 3.3) guarantees: any independence reported by d-separation genuinely holds in every distribution that factorizes over G — d-separation never lies. Completeness is more subtle: d-separation does not catch **every** independence that might hold in a specific numerical distribution P (there can be accidental, parameter-specific independencies), but it catches every independence that holds for **almost all** distributions consistent with G (Theorem 3.5) — exceptions form a measure-zero set in parameter space.

6. I-Equivalence, Minimal I-Maps, and Perfect Maps

6.1 I-equivalence

$$\mathcal{G}_1 \text{ and } \mathcal{G}_2 \text{ are I-equivalent} \iff \mathcal{I}(\mathcal{G}_1) = \mathcal{I}(\mathcal{G}_2)$$

Different graphs can encode identical independence sets. Patterns (a), (b), (c) in Figure 2 are all I-equivalent to each other (all encode only (X⊥Y|Z)), but **not** equivalent to (d), the v-structure, which encodes marginal independence (X⊥Y) instead.

$$\text{Same skeleton} + \text{same immoralities} \iff \text{I-equivalent (Thm 3.8)}$$

An **immorality** is a v-structure X→Z←Y with **no** direct edge between X and Y. This same-skeleton-same-immoralities test is the practical way to check I-equivalence.

6.2 Minimal I-maps are not enough

A minimal I-map removes every redundant edge, but **different variable orderings give different minimal I-maps** — and most of them are needlessly dense, failing to reveal all true independencies. For the Student distribution, the natural order (D,I,S,G,L) recovers the original sparse network, but the order (L,D,S,I,G) produces an almost-complete graph — both are valid minimal I-maps of the same distribution, yet only one is useful.

6.3 Perfect maps

$$\mathcal{I}(\mathcal{G}) = \mathcal{I}(P) \quad \text{(Perfect map)}$$

A perfect map (P-map) captures **exactly** the independencies in P, no more, no less. Not every distribution has one — two classic counterexamples: (1) a distribution with a deterministic XOR relationship among three variables, where the resulting independence pattern cannot be represented by any DAG; (2) the “Misconception” example, where four students' pairwise study relationships (a cycle: A-B-C-D-A) imply symmetric independencies (A⊥C|B,D) and (B⊥D|A,C) that **no** DAG can represent simultaneously — this is precisely the case that motivates **undirected** graphical models (Markov networks), covered in Chapter 4.

When a P-map exists, it can be recovered algorithmically: build the undirected skeleton via conditional-independence tests, mark immoralities, then propagate three local orientation rules to a fixed point (Algorithm 3.5) — producing a **class PDAG** (partially directed graph) representing the entire I-equivalence class at once, rather than committing arbitrarily to one member of it.

7. Summary

- A Bayesian network is a DAG + CPDs; it factorizes the joint distribution via the chain rule and encodes conditional independence assumptions — two equivalent views (Theorems 3.1–3.2).
- Naïve Bayes is the special case where all features are children of a single class node.
- D-separation reads off any implied independence from the graph alone, using active-trail analysis of the four two-edge patterns — soundly (Thm 3.3) and completely for almost all parameterizations (Thm 3.5).
- I-equivalent graphs (same skeleton + same immoralities) encode identical independencies; the direction of an edge is not always identifiable from data alone.
- Minimal I-maps depend on variable ordering and can be misleadingly dense; perfect maps capture independencies exactly but do not always exist — motivating undirected models.