Decisions and Bayes Ball

Inference gives probabilities.

Loss determines the decision.

A DAG reveals independence assumptions.

A posterior probability and a decision

For one input x, our model gives

$$p(C = 1 \mid x) = 0.4, \qquad p(C = 0 \mid x) = 0.6.$$

Quick vote

Should we predict class 0 or class 1?

What else would you need to know?

Here $C = 1$ denotes the positive class. Class labels are arbitrary.

Conditional risk

L_{kj} is the loss when the true class is k and we choose j.

$$\underbrace{R(j \mid x)}_{\text{expected loss of action } j} = \sum_k L_{kj}\, p(C = k \mid x)$$

$$j^*(x) \in \arg\min_j R(j \mid x).$$

Under zero-one loss, $L_{kj} = \mathbf{1}\{k \neq j\}$:

$$R(j \mid x) = 1 - p(C = j \mid x).$$

The most probable class minimizes the chance of error.

Unequal error costs

Pair exercise **90 seconds**

Keep $p(C = 1 \mid x) = 0.4$. Use this illustrative loss matrix:

	Choose 0	Choose 1
True 0	0	1
True 1	4	0

1. Compute $R(0 \mid x)$ and $R(1 \mid x)$.
2. Which action minimizes expected loss?
3. At what posterior would the two actions tie?

The loss changes the threshold

$$R(0 \mid x) = 4(0.4) = 1.6, \qquad R(1 \mid x) = 1(0.6) = 0.6.$$

Choose class 1, even though class 0 is more probable.

Write $q = p(C = 1 \mid x)$ and let correct decisions cost zero.

c_{FP}: false-positive cost. c_{FN}: false-negative cost.

$$R(1 \mid x) = c_{\mathrm{FP}}(1 - q), \qquad R(0 \mid x) = c_{\mathrm{FN}}\, q.$$

For positive error costs, choose 1 when

$$\boxed{q > \frac{c_{\mathrm{FP}}}{c_{\mathrm{FP}} + c_{\mathrm{FN}}}}.$$

Here the threshold is $1/5 = 0.2$. Either action is optimal at equality.

A reject action has a cost too

Now use **zero-one classification loss** and a fixed reject cost λ.

$$R(0 \mid x) = q, \qquad R(1 \mid x) = 1 - q, \qquad R(\text{reject} \mid x) = \lambda.$$

Reject when it has strictly lower risk:

$$\lambda < \min(q, 1 - q) \quad \iff \quad \max_c p(C = c \mid x) < 1 - \lambda.$$

The lecture threshold is $\theta = 1 - \lambda$.

For $\lambda = 0.2$, reject when $\boxed{0.2 < q < 0.8}$.

At either boundary, rejection and one class prediction tie.

Check: what happens at $q = 0.4$? At $q = 0.9$?

Squared loss selects the conditional mean

For a real-valued target T, fix x and write $\mu = \mathbb{E}[T \mid x]$.

$$\mathbb{E}[(y - T)^2 \mid x] = (y - \mu)^2 + \mathrm{Var}(T \mid x).$$

Only the first term changes with y. Therefore

$$\boxed{y^*(x) = \mathbb{E}[T \mid x]}.$$

Check

If $T \mid x$ equals 0 with probability 0.75 and 4 with probability 0.25, what prediction minimizes squared loss?

Posterior updating and decisions

Bayesian updating combines a prior with new evidence:

$$\underbrace{p(C = c \mid x)}_{\text{posterior}} = \frac{\overbrace{p(x \mid C = c)}^{\text{likelihood}}\ \overbrace{p(C = c)}^{\text{prior}}}{\sum_k p(x \mid C = k)p(C = k)}.$$

Our opening example supplied $p(C = 1 \mid x) = 0.4$.

Decisions use these probabilities, however they were obtained.

Changing the loss can change the preferred action.

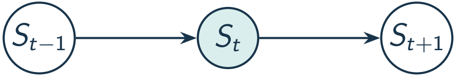
New evidence can change the posterior.

Why graphical models?

A compact model: build a joint distribution from smaller factors.

Relevant evidence: identify information a prediction can safely ignore.

Suppose a robot's state follows a first-order Markov model:

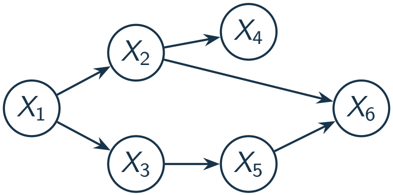


Knowing the current state makes the earlier state unnecessary for this prediction:

$$p(s_{t+1} \mid s_t, s_{t-1}) = p(s_{t+1} \mid s_t).$$

This is a model assumption. The state must contain enough information.

The graph specifies the model



One factor per variable:

$$p(\mathbf{x}) = \prod_i p(x_i \mid x_{\mathrm{pa}(i)}).$$

For X_6, identify its parents.

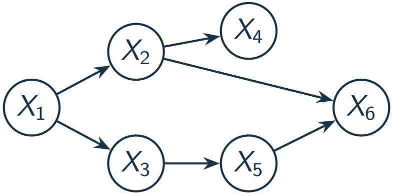
Quick check

What is its factor?

A **conditional probability table (CPT)** lists probabilities for each configuration of a node's parents.

Adapted from the current Week 2 lecture, pp. 22–23.

The graph specifies the model



One factor per variable:

$$p(\mathbf{x}) = \prod_i p(x_i \mid x_{\mathrm{pa}(i)}).$$

For X_6, identify its parents.

Quick check

What is its factor?

$$p(\mathbf{x}) = p(x_1)p(x_2 \mid x_1)p(x_3 \mid x_1)p(x_4 \mid x_2)p(x_5 \mid x_3)p(x_6 \mid x_2, x_5).$$

A **conditional probability table (CPT)** lists probabilities for each configuration of a node's parents.

Adapted from the current Week 2 lecture, pp. 22–23.

Independence, d-separation, and Bayes Ball

Conditional independence is a property of a distribution:

$$A \perp\!\!\!\perp B \mid Z \quad \iff \quad p(a, b \mid z) = p(a \mid z)p(b \mid z).$$

Discrete case: for all a, b and z with $p(z) > 0$.

D-separation is the graph criterion: the evidence blocks every path between the endpoints.

Bayes Ball is an algorithm for checking that criterion.

If a distribution factorizes over the DAG, d-separation guarantees independence for every valid choice of the CPTs.

D-connection means the graph does not guarantee independence.

The local path rules

Pattern	Path	Blocked when
Chain	$A \to M \to B$	M is observed
Fork	$A \leftarrow M \to B$	M is observed
Collider	$A \to M \leftarrow B$	Neither M nor any descendant of M is observed

A path is active when no middle node blocks it.

D-separation requires every path between the endpoints to be blocked.

Arrival direction is relative to the current node

At node V:

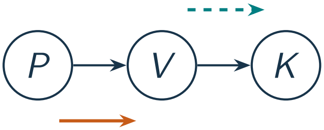
- Arrival from P is **from a parent**.
- Arrival from K is **from a child**.

The ball can travel against a DAG arrow.

The DAG arrows stay fixed throughout the algorithm.

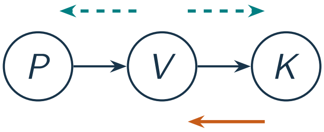
An unobserved node

Arrival from a parent



Send to all children.
No direct return to parents.

Arrival from a child



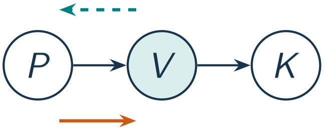
Send to all parents and all children.
This permits travel through a fork.

Solid orange: arrival. Dashed teal: allowed departures.

Solid navy arrows describe the DAG. Each sketch shows one parent and one child.

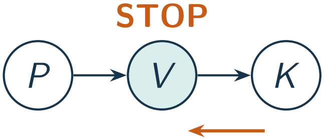
An observed node

Arrival from a parent



Send to all parents.
No departure to children.

Arrival from a child



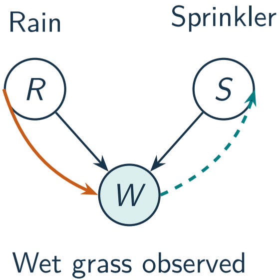
Stop this branch.
No departure to parents or children.

Shaded nodes are observed.

Entering from one parent and leaving toward another crosses an observed collider.

Why an observed effect connects its parents

The ball tracks whether information could change a belief elsewhere.



This toy model treats rain and sprinkling as independent before observing the grass.
Given wet grass, learning that it rained can make sprinkling less likely.
Rain provides an explanation: explaining away.

At observed W: $R \rightsquigarrow W \rightsquigarrow S$ links its parents.

An observed W blocks a chain $R \to W \to D$: its value is already known.

Returning to the same parent can open routes through a new arrival direction.

The four Bayes Ball movement rules

Current node	Ball arrives from	Send the ball to
Unobserved	A parent	All children
Unobserved	A child	All parents and children
Observed	A parent	All parents
Observed	A child	Nowhere

Apply the row to the current node and arrival direction.

Sending to several neighbors creates several branches to explore.

Running a complete search

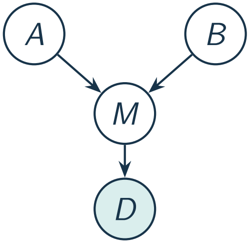
To check whether the graph implies $A \perp\!\!\!\perp B \mid Z$:

1. Shade the evidence Z.
2. Start at A, **as if arriving from a child**.
3. Apply the movement rules and explore every resulting branch.
4. Record $(\text{node}, \text{arrival direction})$. Skip a state already processed.

Target reached: an active connection exists.

Search exhausted without reaching it: the nodes are d-separated.

Why an observed descendant opens a collider



1. M: from a parent, unobserved. Send to D.
2. D: from a parent, observed. Return to M.
3. M: now from a child. Send to its parents, including B.

Ball visits: $A \rightsquigarrow M \rightsquigarrow D \rightsquigarrow M \rightsquigarrow B$.

M stays unobserved. Its arrival direction changes on the return visit.

Allowed or blocked?

Individual check **60 seconds**

At the current node V, is each proposed move allowed?

	V	Arrival	Proposed departure
1	Unobserved	From a parent	To another parent
2	Observed	From a parent	To another parent
3	Unobserved	From a child	To another child
4	Observed	From a child	To a parent

Which two facts determine the answer?

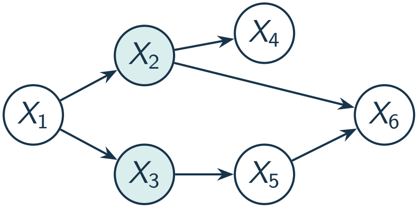
Allowed or blocked: answers

Move	Allowed?	Reason
1	No	Unobserved, from parent: children only
2	Yes	Observed, from parent: parents only
3	Yes	Unobserved, from child: both directions
4	No	Observed, from child: stop

An ancestor of an observed node is **not itself observed**.

A return visit from a different direction can change the legal departures.

Practice 1: two blocked routes



Pair exercise **90 seconds**

Does the graph imply

$$X_1 \perp\!\!\!\perp X_6 \mid \{X_2, X_3\}?$$

Start at X_1.
Where can each ball go after it reaches X_2 or X_3?

Record the node, arrival direction, and allowed departures.

Adapted from the current Week 2 lecture, pp. 33–34.

Practice 1: complete ball trace

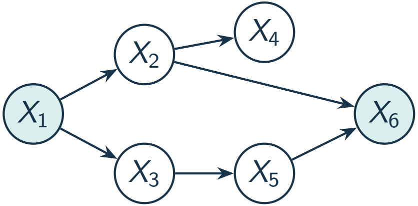
State processed	Status	Departures
(X_1, child)	Unobserved	X_2, X_3
(X_2, parent)	Observed	X_1
(X_3, parent)	Observed	X_1
(X_1, child) again	Already processed	Skip

The search ends without reaching X_6. The graph therefore implies

$$\boxed{X_1 \perp\!\!\!\perp X_6 \mid \{X_2, X_3\}}.$$

Both routes out of X_1 are blocked by observed non-colliders.

Practice 2: changed evidence



Pair exercise **60 seconds**

Does the graph imply

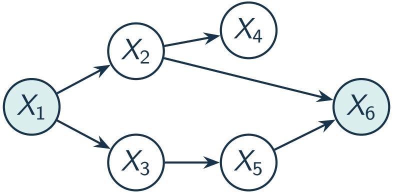
$$X_2 \perp\!\!\!\perp X_3 \mid \{X_1, X_6\}?$$

Start at X_2.
Does reaching observed X_6 stop this branch?

If you find an active connection, give one legal ball sequence.

Adapted from the current Week 2 lecture, pp. 35–36.

Practice 2: an observed collider allows passage



Successful branch

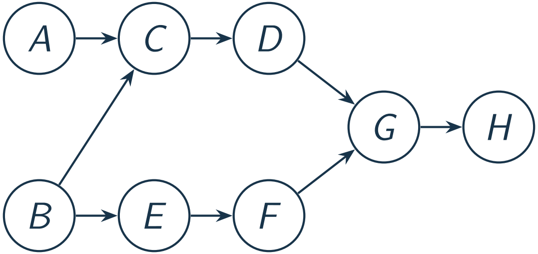
$X_2 \rightsquigarrow X_6 \rightsquigarrow X_5 \rightsquigarrow X_3$.

X_6: observed, entered from a parent. Send to parents.
X_5: unobserved, entered from a child. Send to parents and children.

The active path is $X_2 \to X_6 \leftarrow X_5 \leftarrow X_3$.

The graph does not imply the proposed conditional independence.

Practice 3: a larger graph



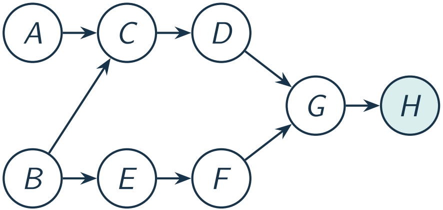
Pair exercise **90 seconds**

Does the graph imply $A \perp\!\!\!\perp E \mid Z$ when:

(a) $Z = \varnothing$ (b) $Z = \{H\}$?

For (b), trace a ball that returns from observed H.

Practice 3: the descendant creates a return route



(a) D-separated.
C blocks $A \to C \leftarrow B \to E$.
G blocks
$A \to C \to D \to G \leftarrow F \leftarrow E$.

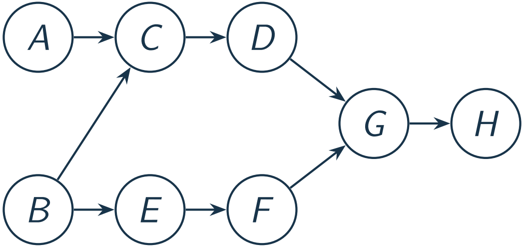
(b) Not d-separated.
H is an observed descendant of both colliders.

One legal ball sequence for (b):

$$A \rightsquigarrow C \rightsquigarrow D \rightsquigarrow G \rightsquigarrow H \rightsquigarrow G \rightsquigarrow F \rightsquigarrow E.$$

Another active path: $A \to C \leftarrow B \to E$, also opened by H.

Practice 4: evidence can open and block routes



Pair exercise **90 seconds**

Does the graph imply $A \perp\!\!\!\perp E \mid Z$ for each evidence set?

(a) $Z = \{B, H\}$ (b) $Z = \{B, D\}$.

Give a successful route, or explain why the whole search ends.

Practice 4: solutions

(a) $Z = \{B, H\}$**: not d-separated.**

The shorter route through B is blocked, but

$$A \to C \to D \to \underbrace{G}_{\text{opened by } H} \leftarrow F \leftarrow E$$

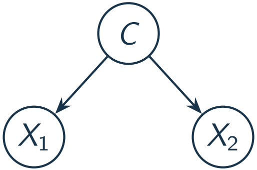
is active. The return through H still reaches E via G, F.

(b) $Z = \{B, D\}$**: d-separated.**

D returns the ball to C. From C, the branch toward observed B stops on arrival from a child. The branches toward A and D repeat processed states.

An open collider does not make every route active.

Putting it together: a spam filter



$C = 1$: spam. $C = 0$: legitimate email.
$X_1 = 1$: the word "free" appears.
$X_2 = 1$: the word "meeting" appears.

Naive Bayes assumption: $X_1 \perp\!\!\!\perp X_2 \mid C$.

$$p(c, x_1, x_2) = p(c)\, p(x_1 \mid c)\, p(x_2 \mid c).$$

Given C, the fork is blocked. Without C, the graph permits dependence.

Conditional independence is a modeling assumption about the words.

Spam filtering: the model and a new email

Use these illustrative probabilities, with $p(C = 1) = 0.2$:

Class	$p(X_1 = 1 \mid C)$	$p(X_2 = 1 \mid C)$
Spam ($C = 1$)	0.8	0.2
Legitimate ($C = 0$)	0.4	0.4

The new email contains “free” and does not contain “meeting”.

We observe $x = (1, 0)$.

Pair calculation **45 seconds**

Compute $p(C = 1 \mid X_1 = 1, X_2 = 0)$.

Include the absent word and normalize over both classes.

Bayes' rule produces the posterior

Conditional independence lets us multiply the feature likelihoods:

$$\begin{aligned} w_1 &= p(C = 1, x) = 0.2 \times 0.8 \times (1 - 0.2) = 0.128, \\ w_0 &= p(C = 0, x) = 0.8 \times 0.4 \times (1 - 0.4) = 0.192. \end{aligned}$$

Normalize over the possible classes:

$$p(C = 1 \mid x) = \frac{w_1}{w_1 + w_0} = \frac{0.128}{0.320} = \boxed{0.4}.$$

The evidence raised the spam probability from 0.2 to 0.4.

We have now calculated the posterior supplied in the opening example.

The posterior and the filtering decision

$p(\text{spam} \mid x) = 0.4$. Choose between the inbox and the spam folder.

Correct decisions cost zero in both cases below.

Loss assumption	Risk: inbox	Risk: spam folder	Best action
Both errors cost 1	0.4	0.6	Inbox
Missed spam costs 4	$4(0.4) = 1.6$	$1(0.6) = 0.6$	Spam folder

In the second row, sending legitimate mail to spam costs 1.

The DAG and its probabilities give the posterior.

The loss function determines the preferred action.

Graph guarantees and AI applications

Extensions of this week's ideas

A guarantee for every factorizing distribution

For the fork $A \leftarrow Z \rightarrow B$, the graph specifies

$$p(a, b, z) = p(z)\, p(a \mid z)\, p(b \mid z).$$

Different valid probability tables give different joint distributions.

For any of those choices, whenever $p(z) > 0$,

$$p(a, b \mid z) = \frac{p(z)\, p(a \mid z)\, p(b \mid z)}{p(z)} = p(a \mid z)\, p(b \mid z).$$

Thus $A \perp\!\!\!\perp B \mid Z$, without needing specific numerical values.

The reverse guarantee also holds: an independence true in every distribution factorizing over the DAG must follow from d-separation.

Extra independence and faithfulness

For $A \to B$, take $p(A = 1) = 0.5$ and this CPT for B:

Parent A	$p(B = 0 \mid A)$	$p(B = 1 \mid A)$
$A = 0$	0.3	0.7
$A = 1$	0.3	0.7

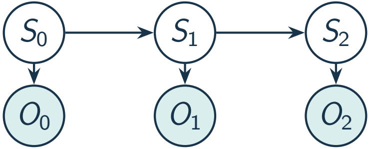
Identical rows give $A \perp\!\!\!\perp B$, despite the direct edge.

This distribution still factorizes as $p(a)p(b \mid a)$.

Faithfulness means the distribution has exactly the graph-implied independences. It rules out this extra independence.

Then the converse holds by definition. Faithfulness is an additional assumption.

Robot localization: a hidden state



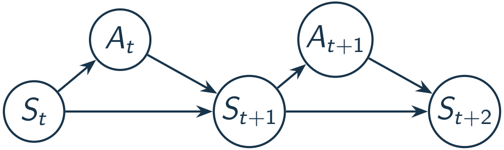
S_t: robot location. O_t: noisy sensor reading.

$$p(s_{0:T}, o_{0:T}) = p(s_0) \prod_{t=1}^{T} p(s_t \mid s_{t-1}) \prod_{t=0}^{T} p(o_t \mid s_t).$$

Past observations can help infer the current hidden state.

The Markov assumption applies to the state process.

MDPs: actions influence the next state



This trajectory graph assumes a fixed Markov policy $\pi(a_t \mid s_t)$.

$$p(s_{t+1} \mid s_{0:t}, a_{0:t}) = p(s_{t+1} \mid s_t, a_t).$$

An MDP adds actions, transitions, and rewards. Planning chooses a policy to maximize expected return:

$$\max_{\pi} \mathbb{E}_{\pi}\left[\sum_{t=0}^{T-1} \gamma^t R_t\right].$$

Bayes Ball checks independence. Planning chooses the actions.

An AI agent can pay for more information

Illustrative robot decision: $p(\text{obstacle}) = 0.4$.

Loss	Proceed	Wait
Clear route	0	1
Obstacle	10	1

Without another reading, risks are 4 and 1. The robot waits.

A hypothetical perfect detector allows the best action in each state:

$$\text{Expected loss after observing} = 0.6(0) + 0.4(1) = 0.4.$$

Perfect information has value $1 - 0.4 = \boxed{0.6}$ loss units.

Check: use a perfect detector costing 0.2?

An autoregressive LLM as a token DAG

Within a fixed context, each token can depend on all earlier tokens:

$$
\begin{tikzcd}
X_1 \arrow[r] \arrow[rr, bend left] \arrow[rrr, bend left] & X_2 \arrow[r] \arrow[rr, bend right] & X_3 \arrow[r] & X_4
\end{tikzcd}
$$

$$p_\theta(x_{1:T}) = \prod_{t=1}^{T} p_\theta(x_t \mid x_{<t}).$$

All edges point forward in token position, so this is a DAG.

This complete DAG implies no nontrivial conditional independences.

A neural model with shared parameters represents the conditional distributions.

References

Course material

Ohad Shamir, CSC412/2506, Week 2 lecture

Graphical models and Bayes Ball: pp. 19–36. Decision theory: pp. 3–17.

Bayes Ball

Ross Shachter, Bayes-Ball: The Rational Pastime, Section 3.

AI extensions

Berkeley CS188: Markov Decision Processes

Berkeley CS188: The Value of Perfect Information

Radford et al. (2019): autoregressive language modeling, Section 2