CS 3630!

Lecture 3:
Probability
and Bayes Nets

Topics

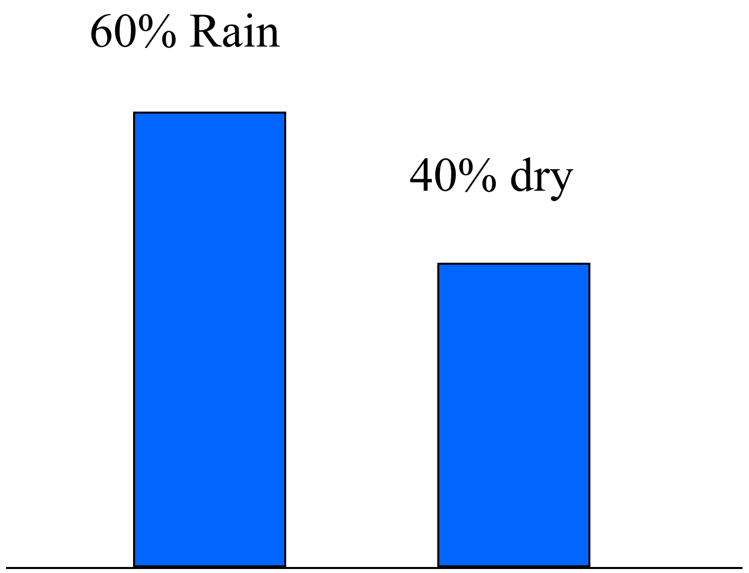
- **1. Discrete distributions**
- **2. Inverse transform sampling**
- **3. Conditional probability**
- **4. Models for sensing and acting**
- **5. Joint probability**
- **6. Bayes' rule**
- **7. Bayes nets**
- **8. Ancestral sampling**
- **9. Dynamic Bayes**
- **10. Inference in Bayes nets**

Motivation

- Probability -> simulate robots !
- Our example: grid world
- Probabilistic statements about state: Bayesian inference

The Bayesian Paradigm

- Knowledge as a probability distribution



1. Discrete Distributions

Figure 2.1: Graphical representation of a single random variable.

- Probability Mass Function or PMF
- Properties:

$$p_i \geq 0$$

$$\sum p_i = 1$$

a_i	$P(A = a_i)$
Left	0.2
Right	0.6
Up	0.1
Down	0.1

Table 2.1: Probability mass function.

Discrete Distributions in Python

a_i	$P(A = a_i)$
Left	0.2
Right	0.6
Up	0.1
Down	0.1

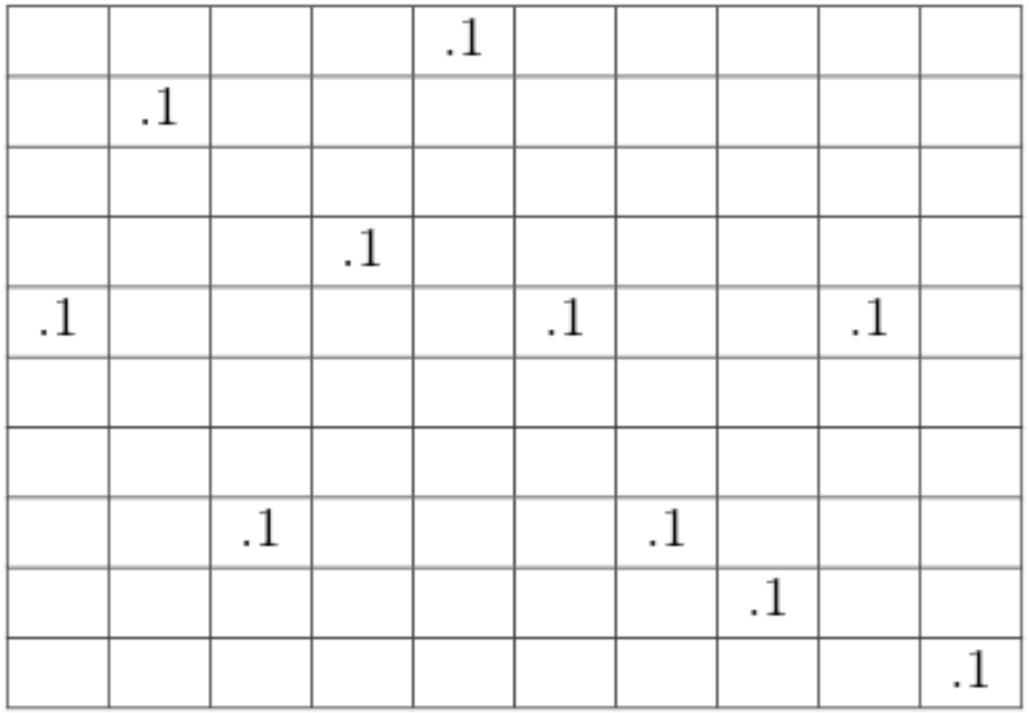
Table 2.1: Probability mass function.

```python
ACTIONS = ["Left", "Right", "Up", "Down"]
```

```python
def test_action_prior(self):
    """Check the prior on actions."""
    sum = 0.0
    for action in ACTIONS:
        p = action_prior(action)
        self.assertGreaterEqual(p, 0)
        self.assertLessEqual(p, 1.0)
        sum += p
    self.assertEqual(sum, 1.0)
```

```python
def action_prior(action):
    """Return prior probability distribution on an action.
      arguments:
        action (string): "Left", "Right", "Up", "Down"
      returns:
        probability of given action (float).
    """
    p = {"Left": 0.2, "Right": 0.6, "Up": 0.1, "Down": 0.1}
    return p[action]
```

Grid world example



				.1					
	.1								
			.1						
.1					.1			.1	
		.1				.1			
							.1		
									.1

- 100 outcomes!

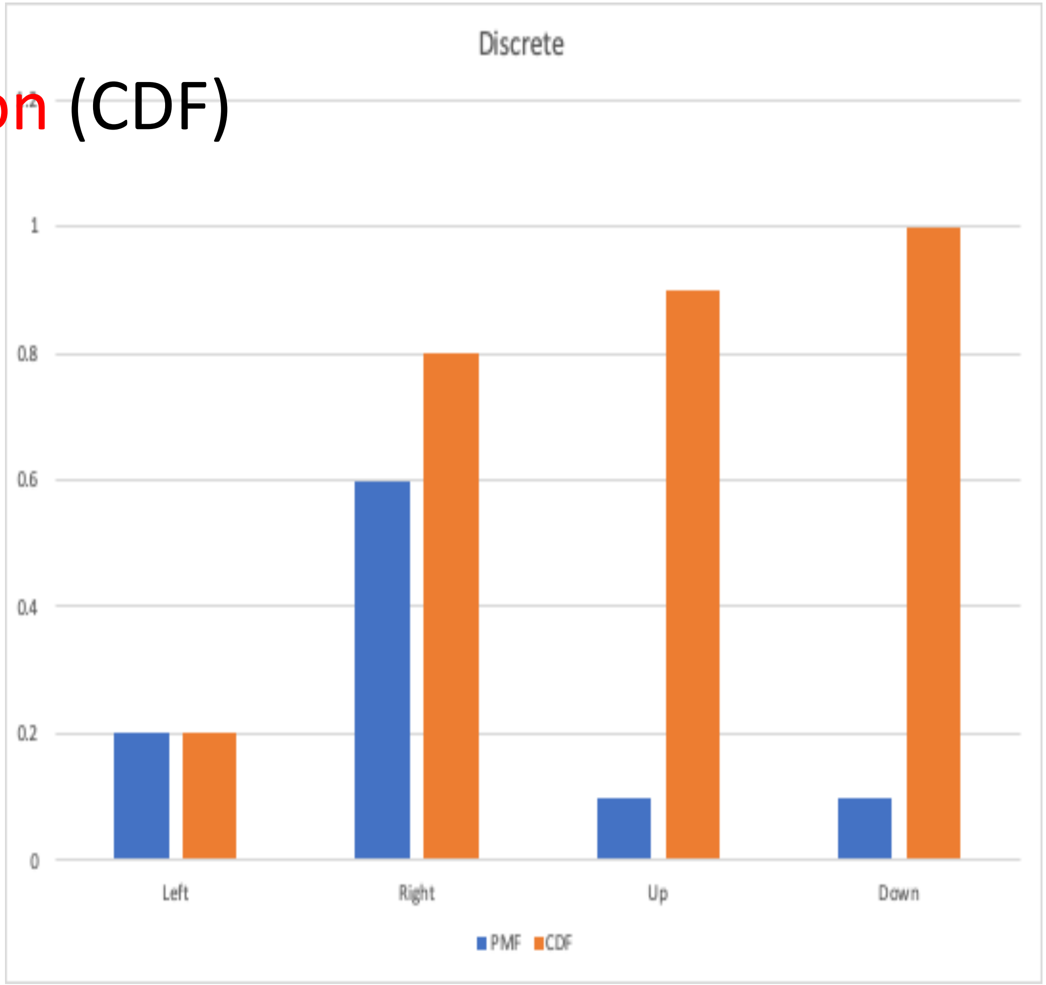
Table 2.2: A PMF describing where a robot might be in a grid.

2. Inverse transform sampling

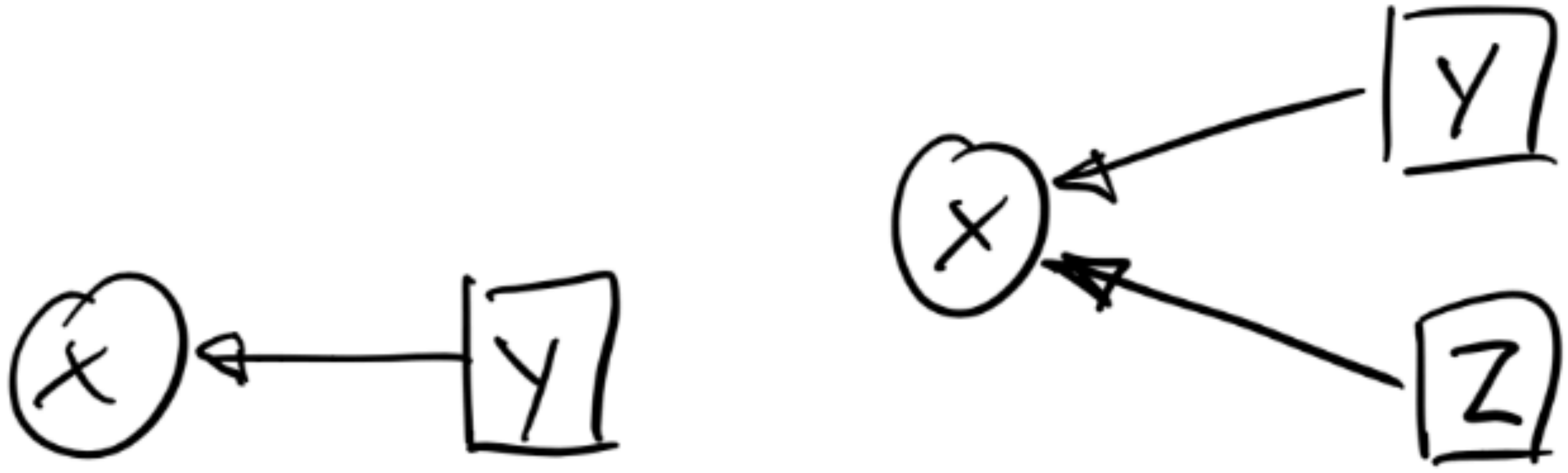
- Order outcomes
- Form cumulative distribution function (CDF)
- Sample random 0.0 <= u <= 1.0

a_i	$P(A = a_i)$	$F(a_i)$
Left	0.2	0.2
Right	0.6	0.8
Up	0.1	0.9
Down	0.1	1.0

Table 2.3: Cumulative distribution function.



3. Conditional Distributions

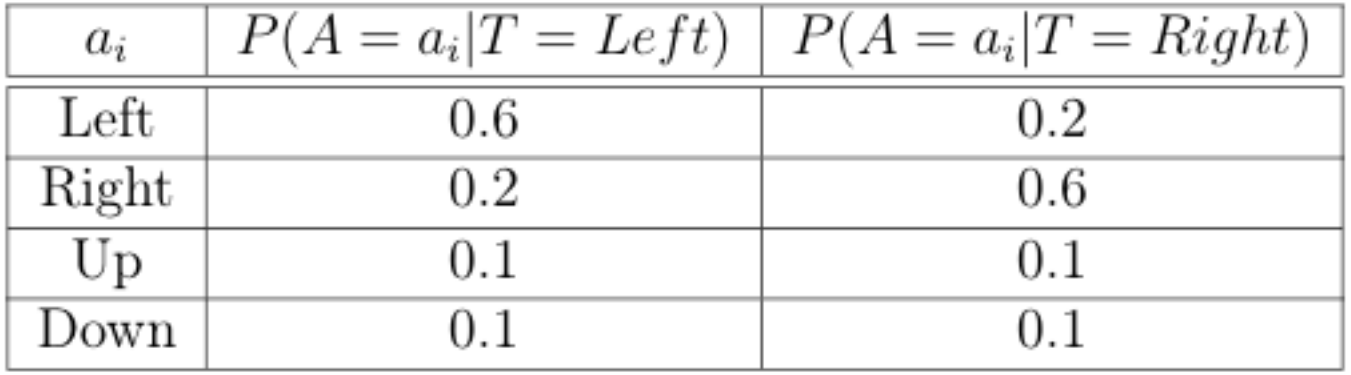


(a) $P(X|Y)$ (b) $P(X|Y,Z)$

Figure 2.2: Conditional probability distributions.

- A distribution that depends on a (known) parameter or parameters.

Example

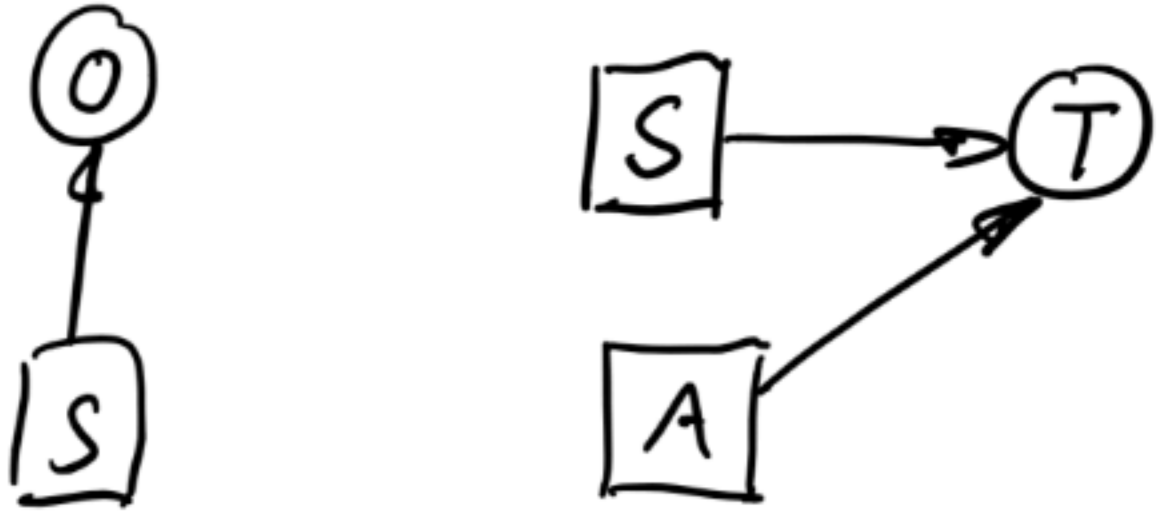


a_i	$P(A = a_i \mid T = Left)$	$P(A = a_i \mid T = Right)$
Left	0.6	0.2
Right	0.2	0.6
Up	0.1	0.1
Down	0.1	0.1

Table 2.4: Conditional probability table (CPT).

- CPT: conditional probability table
- Example: robot “tending” to the left or the right

4. Modeling the World



(a) Sensor Model $P(O|S)$

(b) State transition model $P(T|S, A)$

Figure 2.3: Conditional distributions to model sensing and acting.

- Conditionals are cool: we can use them to model sensing *and* acting.

Parametric descriptions

$$\begin{cases} P(O=k|S=i,j)=1 & \text{iff } k=j \\ P(O=k|S=i,j)=0 & \text{otherwise} \end{cases}$$

$$\begin{cases} P(O=k|S=i,j)=0.91 & \text{iff } k=i \\ P(O=k|S=i,j)=0.01 & \text{otherwise} \end{cases}$$

- Sensor model P(O|S)
- CPTs can become very big
- Uniformity: implement as a function!
- Exercise: what are the above models? *S = (i,j)* is robot location in grid.

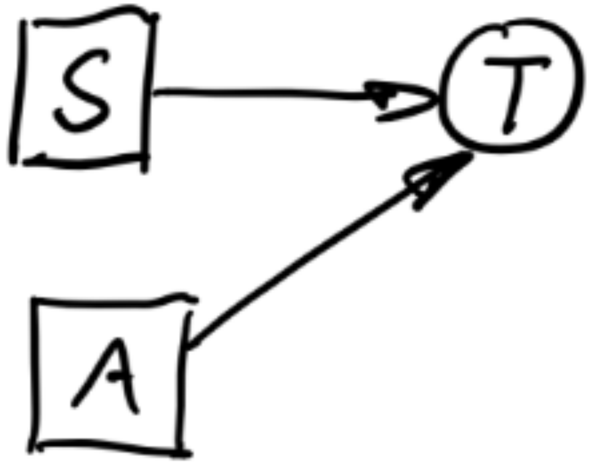
Parametric descriptions (cont’d)

$$\begin{cases} P(O=k|S=i,j)=1 & \text{iff } k=j \\ P(O=k|S=i,j)=0 & \text{otherwise} \end{cases}$$

$$\begin{cases} P(O=k|S=i,j)=0.91 & \text{iff } k=i \\ P(O=k|S=i,j)=0.01 & \text{otherwise} \end{cases}$$

- Uniformity: implement as a function!
- Left: report the horizontal coordinate *j* of the robot faithfully
- Right: reports the vertical coordinate *i* of the robot, but with 9% probability gives a random faulty reading

Modeling action



- We can do the same for modeling action:
- State transition model *P(T|S,A)*
- Exercise: come up with a fairly realistic model for grid world.

5. Joint Distributions

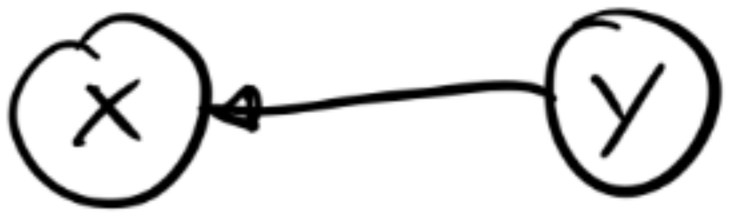
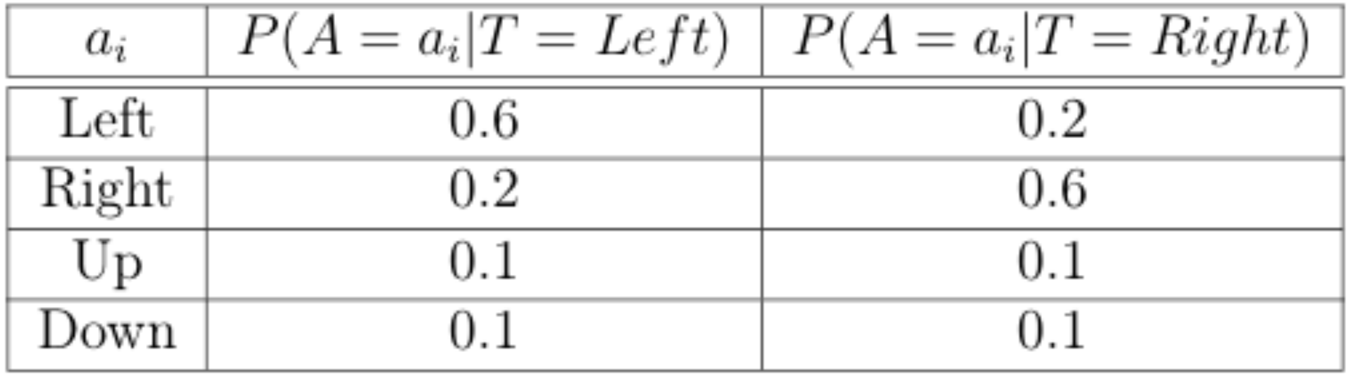


Figure 2.4: Joint probability distribution on two variables X and Y.

- What if parameter in conditional is itself a random variable?
- Chain rule: $P(X,Y) = P(X|Y)\,P(Y)$
- Riddle: How do we sample?

Joint Distributions (cont'd)



a_i	$P(A = a_i \mid T = Left)$	$P(A = a_i \mid T = Right)$
Left	0.6	0.2
Right	0.2	0.6
Up	0.1	0.1
Down	0.1	0.1

Table 2.4: Conditional probability table (CPT).

- Marginals
- Conditionals
- Exercise: let's make some tables!
  - Joint P(A,T), assuming P(T=Right)=0.7
  - Both marginals, both conditionals

6. Bayes' Rule

- Inference:
  - probabilistic statements about what we *know*
- Given: we observe a sensor measurement $O=o$
- What can we say about the state S ?
- You need:
  - Sensor model $P(O|S)$
  - Prior probability distribution $P(S)$
- What we want:
  - Posterior probability distribution $P(S|O=o)$

Bayes' Rule (cont'd)

- *P(S|O) = P(S,O) / P(O) = P(O|S) P(S) / P(O)*
- Hence:

$$\boxed{P(S|O=o) = \frac{P(O=o|S)P(S)}{P(O=o)}}$$

- This is known as Bayes' rule (or: Bayes' law....)

In class exercise

				.1					
	.1								
			.1						
.1					.1			.1	
		.1				.1			
							.1		
									.1

$$\begin{cases} P(O=k|S=i,j)=0.91 & \text{iff } k=i \\ P(O=k|S=i,j)=0.01 & \text{otherwise} \end{cases}$$

- Apply Bayes' rule to calculate the posterior *P(S|O=5)*
- *First think about the representation of the result: what is it?*

Likelihood functions

- In Bayes' law, given *O=o*, all are functions of S

$$P(S|O=o) = \frac{P(O=o|S)P(S)}{P(O=o)}$$

- Introduce the likelihood function:

$$L(S;o) \triangleq P(O=o|S)$$

- Bayes' law:

$$\boxed{P(S|O=o) \propto L(S;o)P(S)}$$

The many ways of Bayes

- Classic:

$$\boxed{P(S|O=o) = \frac{P(O=o|S)P(S)}{P(O=o)}}$$

- Intuitive:

$$\boxed{P(S|O=o) \propto L(S;o)P(S)}$$

- Bare-bones:

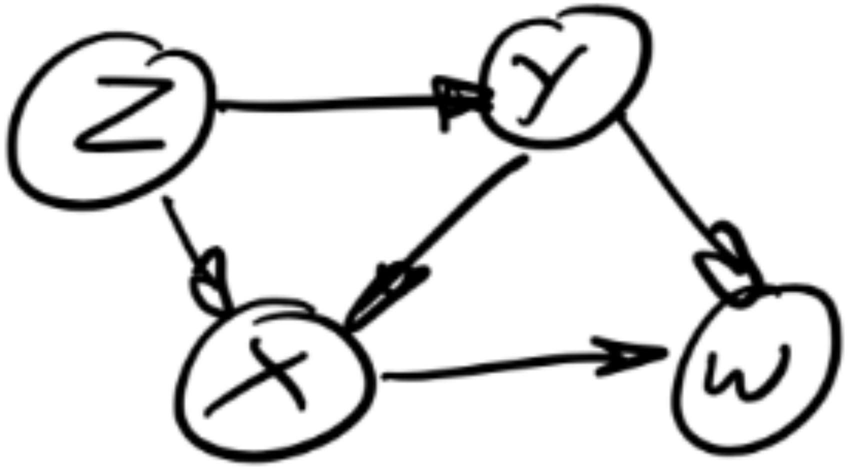
$$\boxed{P(S|O=o) \propto P(S,O=o)}$$

In-class exercise

Suppose we are interested in estimating the probability of an obstacle in front of the robot, given an "obstacle sensor" that is accurate 90% of the time. Apply Bayes' rule to this example by creating the sensor model, prior, and deriving the posterior.

For the previous example, what happens if the sensor is random, i.e., it just outputs "obstacle detected" with 50% probability?

7. Bayes Nets

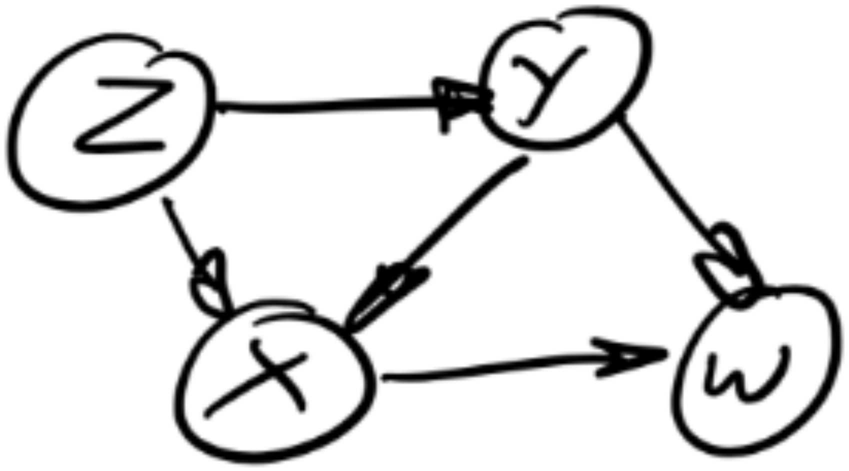


- A Bayes net is a directed acyclic graph (DAG) of conditionals.
- The joint is given by multiplying all conditionals:

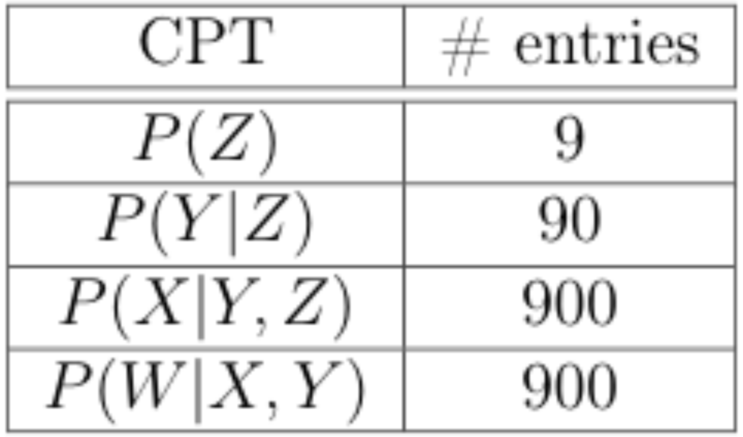
$$P(\{X_i\}) = \prod_{i=1}^{n} P(X_i|\Pi_i)$$

- Exercise: $P(W,X,Y,Z,$
- Note chain rule $P(X,Y) = (X|Y)P(Y)$ is a special case.

Bayes Nets (cont'd)



- P(W|X,Y)P(X|Y,Z)P(Y|Z)P(Z)
- A Bayes net is very efficient!
- Assume *W,X,Y,Z* are all 10-valued. How many entries in the joint PMF?
- How many entries in CPTs ?



CPT	# entries
$P(Z)$	9
$P(Y\|Z)$	90
$P(X\|Y,Z)$	900
$P(W\|X,Y)$	900

8. Ancestral Sampling

$$\begin{tikzcd} Z \arrow[r] \arrow[d] & Y \arrow[dl] \arrow[d] \\ X \arrow[r] & W \end{tikzcd}$$

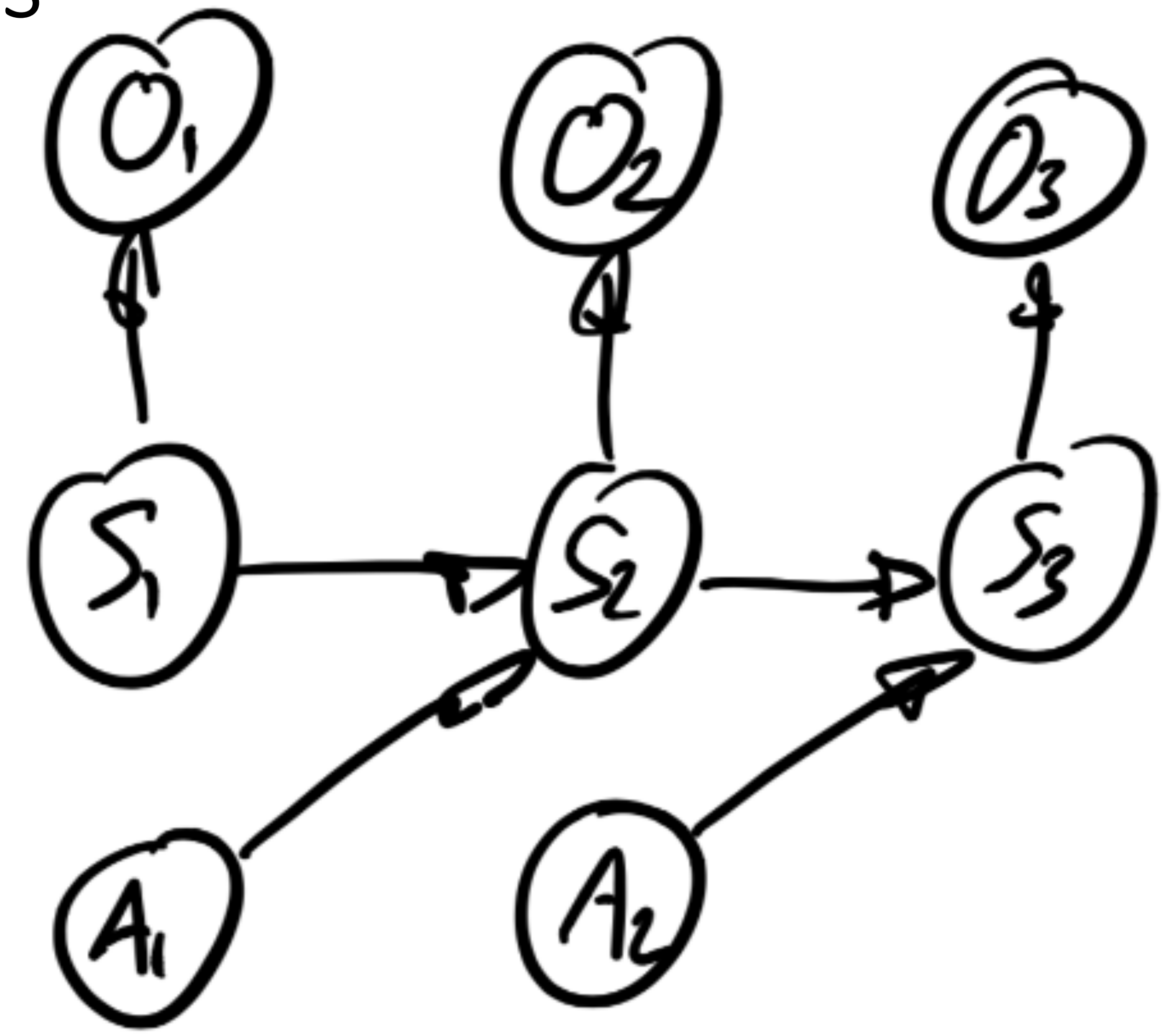
- How do we sample from a Bayes net?
- Recall: sampling from *P(X|Y) P(Y)* ?

- Generalize:
  1. topological sort (Kahn’s algorithm)
  2. Sample in topological sort order

$$\begin{tikzcd} Z \arrow[r] \arrow[rr, bend left] & Y \arrow[r] \arrow[rr, bend right] & X \arrow[r] & W \end{tikzcd}$$

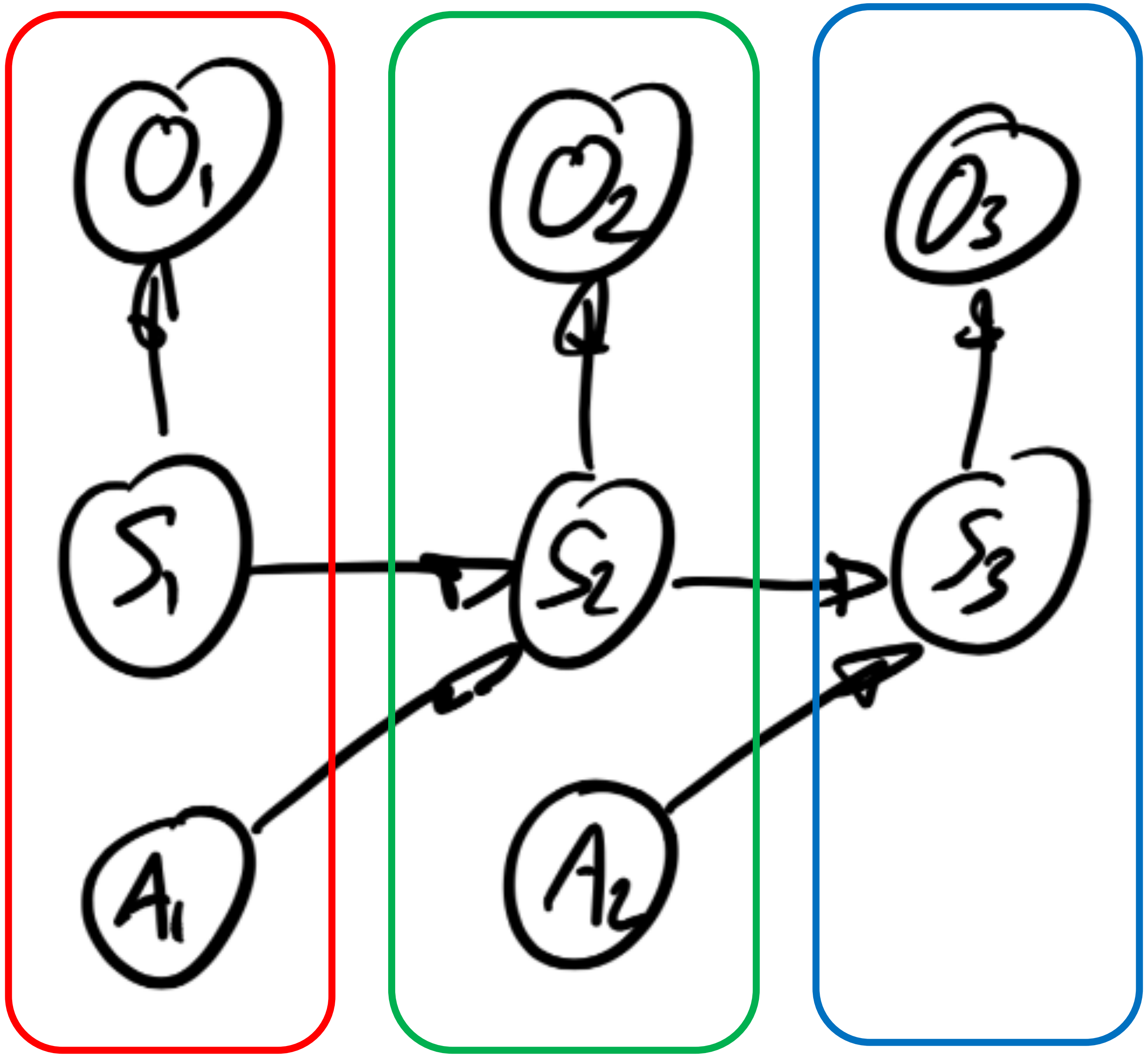
9. Dynamic Bayes Nets

- DBN or dynamic Bayes net: roll out *time*.
- Applied to agents/robots: sequence of sensing and acting!



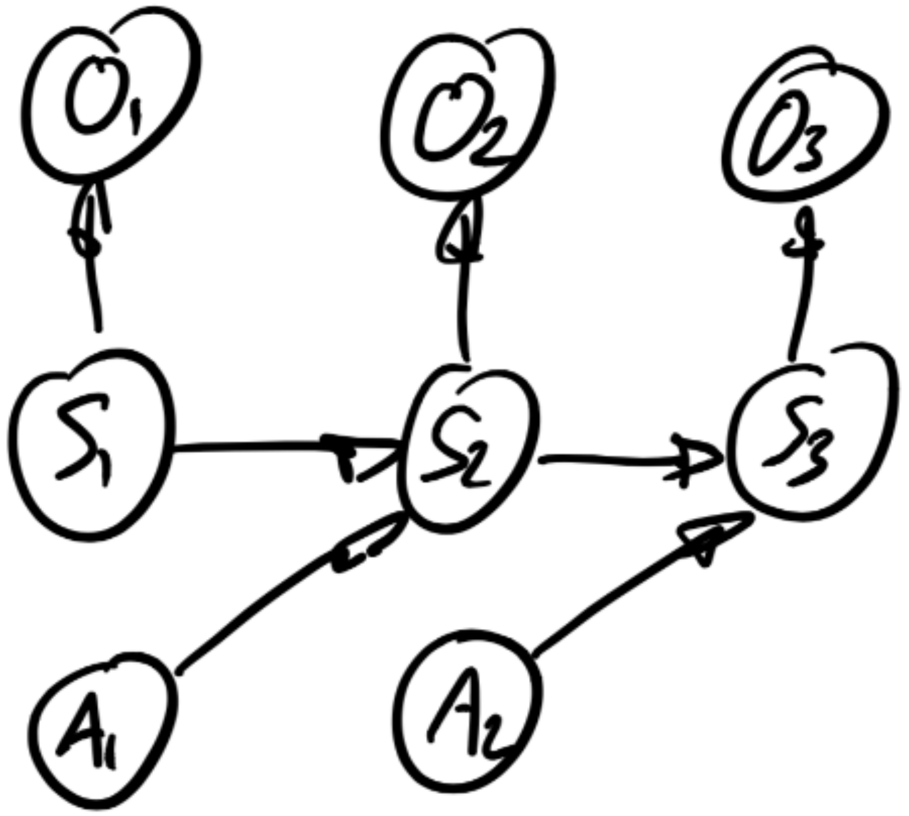
Simulation of Agents

1. Slice 1:
   a) Sample from $P(S_1)$
   b) Sense $P(O_1|S_1)$
   c) Sample from $P(A_1)$
2. Slice 2:
   a) Act $P(S_2|S_1, A_1)$
   b) Sense $P(O_2|S_2)$
   c) Sample from $P(A_2)$
3. *Slice 3:*
   1. …



In-class exercise

- Simulate a realization from this DBN:



1. First, sample the initial state s_1 from $P(S_1)$, a prior over the state. Set $k = 1$.
2. Next, simulate the sensor by sampling from the sensor model $P(O_k|S_k = s_k)$, yielding o_k.
3. Then, sample an action a_k from $P(A_k)$.
4. Lastly, simulate the effect of this action by sampling the next state s_k from

$$P(S_{k+1}|S_k = s_k, A_k = a_k).$$

5. Increase k and return to step 2.

10. Inference in Bayes Nets

- MPE or most probable explanation, given some Z values

$$P(\mathcal{X}|\mathcal{Z} = \mathfrak{z}) \propto P(\mathcal{X}, \mathcal{Z} = \mathfrak{z}).$$

- Find assignment to remaining *X* values such that above is maximized!
- Simple algorithm:
  1. Enumerate all combinations of X values
  2. Calculate posterior
  3. Pick maximum

- More sophisticated algorithm: branch & bound. Discuss !

Naïve inference, exercises

- Exercise 1:
  - condition on W, Z
  - How big is the table?
- Exercise 2:
  - Condition on Y, X
  - Try branch & bound
- Exercise 3:
  - In DBN, assume states are given
  - What is complexity of inferring the actions?

$$\begin{tikzcd}
Z \arrow[rr] \arrow[dr] & & Y \arrow[dl] \arrow[dr] & \\
& X \arrow[rr] & & W
\end{tikzcd}$$

$$\begin{tikzcd}
O_1 & O_2 & O_3 \\
S_1 \arrow[u] \arrow[r] & S_2 \arrow[u] \arrow[r] & S_3 \arrow[u] \\
A_1 \arrow[ur] & A_2 \arrow[ur] &
\end{tikzcd}$$

Inference in Bayes Nets (cont'd)

- MAP or maximum a posteriori estimate, given some Z values

$$P(\mathcal{X}|\mathcal{Z}=\mathfrak{z}) = \sum_{\mathfrak{y}} P(\mathcal{X},\mathcal{Y}=\mathfrak{y}|\mathcal{Z}=\mathfrak{z}) \propto \sum_{\mathfrak{y}} P(\mathcal{X},\mathcal{Y}=\mathfrak{y},\mathcal{Z}=\mathfrak{z}).$$

- We now have nuisance variables Y, which we need to marginalize out.
- At least as expensive as MPE, in many cases much more so.

Summary

- **Discrete distributions** model the outcome of a single categorical random variable.
- **Inverse transform sampling** allows us to simulate a single variable.
- **Conditional probability** distributions allow for dependence on other variables.
- **Models for sensing and acting** can be built using parametric conditional distributions.
- We can compute a **joint probability** distribution, and marginal and conditionals from it.
- **Bayes' rule** allows us to infer knowledge about a state from a given observation.
- **Bayes nets** allow us to encode more general joint probability distributions over many variables.
- **Ancestral sampling** is a technique to simulate from any Bayes net.
- **Dynamic Bayes nets** unroll time and can be used to simulate robots over time.
- **Inference in Bayes nets** is a simple matter of enumeration, but this can be expensive.