Probability in Programming

Prakash Panangaden
School of Computer Science
McGill University

Two crucial issues

Two crucial issues

- Correctness of software:

Two crucial issues

- Correctness of software:
- with respect to precise specifications.

Two crucial issues

- Correctness of software:
- with respect to precise specifications.
- Efficiency of code:

Two crucial issues

- Correctness of software:
- with respect to precise specifications.
- Efficiency of code:
- based on well-designed algorithms.

Logic is the key

Logic is the key

- Precise specifications have to be made in a formal language

Logic is the key

- Precise specifications have to be made in a formal language
- with a rigourous definition of meaning.

Logic is the key

- Precise specifications have to be made in a formal language
- with a rigourous definition of meaning.
- Logic is the “calculus” of computer science.

Logic is the key

- Precise specifications have to be made in a formal language
- with a rigourous definition of meaning.
- Logic is the “calculus” of computer science.
- It comes with a framework for reasoning.

Logic is the key

- Precise specifications have to be made in a formal language
- with a rigourous definition of meaning.
- Logic is the “calculus” of computer science.
- It comes with a framework for reasoning.
- Many kinds of logic: propositional, predicate, modal, ...

Probability

Probability

- Probability is also a framework for reasoning

Probability

- Probability is also a framework for reasoning
- quantitatively.

Probability

- Probability is also a framework for reasoning
- quantitatively.
- But is this relevant for computer programmers?

Probability

- Probability is also a framework for reasoning
- quantitatively.
- But is this relevant for computer programmers?
- **Yes!**

Probability

- Probability is also a framework for reasoning
- quantitatively.
- But is this relevant for computer programmers?
- **Yes!**
- Probabilistic reasoning is everywhere.

Some quotations

Some quotations

- The true logic of the world is the calculus of probabilities — James Clerk Maxwell

Some quotations

- The true logic of the world is the calculus of probabilities — James Clerk Maxwell
- The theory of probabilities is at bottom nothing but common sense reduced to calculus — Pierre Simon Laplace

Why does one use probability?

Why does one use probability?

- Some algorithms use probability as a computational resource: randomized algorithms.

Why does one use probability?

- Some algorithms use probability as a computational resource: randomized algorithms.

- Software for interacting with physical systems have to cope with noise and uncertainty: telecommunications, robotics, vision, control systems, ...

Why does one use probability?

- Some algorithms use probability as a computational resource: randomized algorithms.
- Software for interacting with physical systems have to cope with noise and uncertainty: telecommunications, robotics, vision, control systems, ...
- Big data and machine learning: probabilistic reasoning has had a revolutionary impact.

Basic Ideas

Basic Ideas

Sample space X: the set of things that can possibly happen.

Basic Ideas

Sample space X: the set of things that can possibly happen.

Event: subset of the sample space; $A, B \subset X$.

Basic Ideas

Sample space X: the set of things that can possibly happen.

Event: subset of the sample space; $A, B \subset X$.

Probability: $\mathsf{Pr} : X \to [0,1]$, $\sum_{x \in X} \mathsf{Pr}(x) = 1$.

Basic Ideas

Sample space X: the set of things that can possibly happen.

Event: subset of the sample space; $A, B \subset X$.

Probability: $\mathsf{Pr} : X \to [0, 1]$, $\sum_{x \in X} \mathsf{Pr}(x) = 1$.

Probability of an event A: $\mathsf{Pr}(A) = \sum_{x \in A} \mathsf{Pr}(x)$.

Basic Ideas

Sample space X: the set of things that can possibly happen.

Event: subset of the sample space; $A, B \subset X$.

Probability: $\mathsf{Pr} : X \to [0, 1]$, $\sum_{x \in X} \mathsf{Pr}(x) = 1$.

Probability of an event A: $\mathsf{Pr}(A) = \sum_{x \in A} \mathsf{Pr}(x)$.

A, B are independent: $\mathsf{Pr}(A \cap B) = \mathsf{Pr}(A) \cdot \mathsf{Pr}(B)$.

Basic Ideas

Sample space X: the set of things that can possibly happen.

Event: subset of the sample space; $A, B \subset X$.

Probability: $\mathsf{Pr} : X \to [0, 1]$, $\sum_{x \in X} \mathsf{Pr}(x) = 1$.

Probability of an event A: $\mathsf{Pr}(A) = \sum_{x \in A} \mathsf{Pr}(x)$.

A, B are independent: $\mathsf{Pr}(A \cap B) = \mathsf{Pr}(A) \cdot \mathsf{Pr}(B)$.

Subprobability: $\sum_{x \in X} \mathsf{Pr}(x) \leq 1$.

A Puzzle

A Puzzle

Imagine a town where every birth is equally likely to give a boy or a girl. $\mathsf{Pr}(\text{boy}) = \mathsf{Pr}(\text{girl}) = \frac{1}{2}$.

A Puzzle

Imagine a town where every birth is equally likely to give a boy or a girl. $\mathsf{Pr}(\text{boy}) = \mathsf{Pr}(\text{girl}) = \frac{1}{2}$.

Each birth is an *independent* random event.

A Puzzle

Imagine a town where every birth is equally likely to give a boy or a girl. $\mathsf{Pr}(\text{boy}) = \mathsf{Pr}(\text{girl}) = \frac{1}{2}$.

Each birth is an *independent* random event.

There is a family with two children.

A Puzzle

Imagine a town where every birth is equally likely to give a boy or a girl. $\mathsf{Pr}(\text{boy}) = \mathsf{Pr}(\text{girl}) = \frac{1}{2}$.

Each birth is an *independent* random event.

There is a family with two children.

One of them is a boy (not specified which one), what is the probability that the other one is a boy?

A Puzzle

Imagine a town where every birth is equally likely to give a boy or a girl. $\mathsf{Pr}(\text{boy}) = \mathsf{Pr}(\text{girl}) = \frac{1}{2}$.

Each birth is an *independent* random event.

There is a family with two children.

One of them is a boy (not specified which one), what is the probability that the other one is a boy?

Since the births are independent, the probability that the other child is a boy should be $\frac{1}{2}$. Right?

Puzzle (continued)

Puzzle (continued)

Wrong!

Puzzle (continued)

Wrong!

Initially, there are 4 *equally likely* situations: bb, bg, gb, gg.

Puzzle (continued)

Wrong!

Initially, there are 4 *equally likely* situations: bb, bg, gb, gg.

The possibility gg is ruled out with the additional information.

Puzzle (continued)

Wrong!

Initially, there are 4 *equally likely* situations:
bb, bg, gb, gg.

The possibility gg is ruled out with the additional information.

So of the three *equally likely* scenarios:

bb, bg, gb,

only one has the other child being a boy.

Puzzle (continued)

Wrong!

Initially, there are 4 *equally likely* situations:
bb, bg, gb, gg.

The possibility gg is ruled out with the additional information.

So of the three *equally likely* scenarios:

bb, bg, gb,

only one has the other child being a boy.

The correct answer is $\frac{1}{3}$.

Puzzle (continued)

Wrong!

Initially, there are 4 *equally likely* situations:
bb, bg, gb, gg.

The possibility gg is ruled out with the additional information.

So of the three *equally likely* scenarios:

bb, bg, gb,

only one has the other child being a boy.

The correct answer is $\frac{1}{3}$.

If I had said, "The *elder* child is a boy", then the probability that the other child is a boy is indeed $\frac{1}{2}$

Conditional probability

Conditional probability

Conditioning = revising probability in the presence of new information.

Conditional probability

Conditioning = revising probability in the presence of new information.

Conditional probability/expectation is *the* heart of probabilistic reasoning.

Conditional probability

Conditioning = revising probability in the presence of new information.

Conditional probability/expectation is *the* heart of probabilistic reasoning.

Conditional probability is tricky!

Conditional probability

Conditioning = revising probability in the presence of new information.

Conditional probability/expectation is *the* heart of probabilistic reasoning.

Conditional probability is tricky!

Analogous to *inference* in ordinary logic.

Conditional probability

Conditional probability

Definition: if A and B are events, the **conditional probability** of A *given* B, written $\mathsf{Pr}(A \mid B)$ is defined by

$$\mathsf{Pr}(A \mid B) = \mathsf{Pr}(A \cap B)/\mathsf{Pr}(B).$$

Conditional probability

Definition: if A and B are events, the **conditional probability** of A *given* B, written $\mathsf{Pr}(A \mid B)$ is defined by

$$\mathsf{Pr}(A \mid B) = \mathsf{Pr}(A \cap B)/\mathsf{Pr}(B).$$

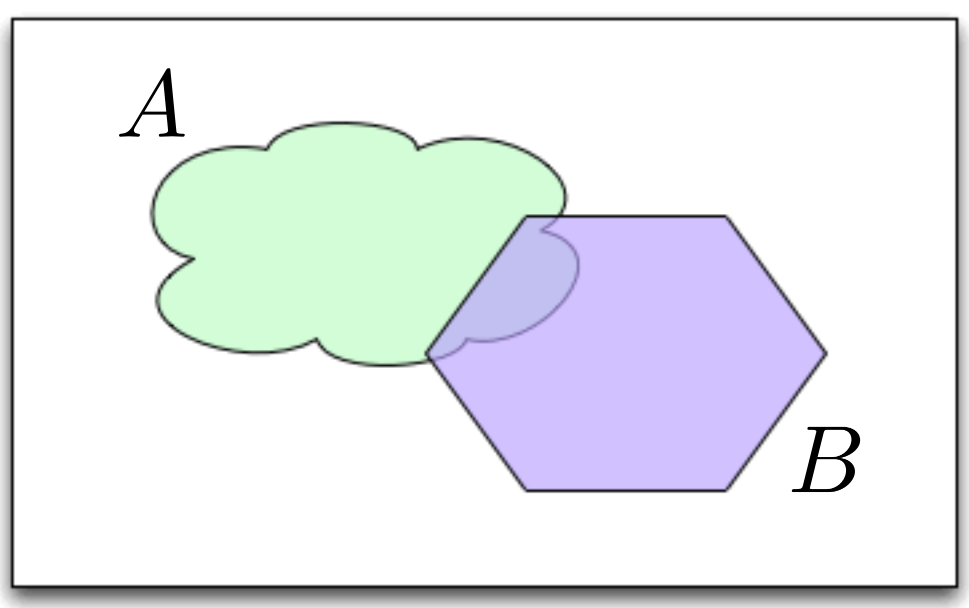
We are told that the outcome is one of the possibilities in B. We now need to change our guess for the outcome being in A.

Conditional probability

Definition: if A and B are events, the **conditional probability** of A *given* B, written $\mathsf{Pr}(A \mid B)$ is defined by

$$\mathsf{Pr}(A \mid B) = \mathsf{Pr}(A \cap B)/\mathsf{Pr}(B).$$

We are told that the outcome is one of the possibilities in B. We now need to change our guess for the outcome being in A.



Bayes' Rule

$$\mathsf{Pr}(A \mid B) = \frac{\mathsf{Pr}(B \mid A) \cdot \mathsf{Pr}(A)}{\mathsf{Pr}(B)}.$$

How to revise probabilities.

Bayes' Rule

$$\mathsf{Pr}(A \mid B) = \frac{\mathsf{Pr}(B \mid A) \cdot \mathsf{Pr}(A)}{\mathsf{Pr}(B)}.$$

How to revise probabilities.

Proof is just from the definition.

Example

Example

Two coins, one fake (two heads) one OK.

Example

Two coins, one fake (two heads) one OK.

One coin chosen with equal probability and then tossed to yield a H.

Example

Two coins, one fake (two heads) one OK.

One coin chosen with equal probability and then tossed to yield a H.

What is the probability the coin was fake?

Example

Two coins, one fake (two heads) one OK.

One coin chosen with equal probability and then tossed to yield a H.

What is the probability the coin was fake? Answer: $\frac{2}{3}$.

Example

Two coins, one fake (two heads) one OK.

One coin chosen with equal probability and then tossed to yield a H.

What is the probability the coin was fake? Answer: $\frac{2}{3}$.

$$\mathsf{Pr}(H \mid \text{Fake}) = 1, \mathsf{Pr}(\text{Fake}) = \tfrac{1}{2}, \mathsf{Pr}(H) = \tfrac{1}{2} \cdot 1 + \tfrac{1}{2} \cdot \tfrac{1}{2} = \tfrac{3}{4}.$$

Example

Two coins, one fake (two heads) one OK.

One coin chosen with equal probability and then tossed to yield a H.

What is the probability the coin was fake? Answer: $\frac{2}{3}$.

$\mathsf{Pr}(H \mid \text{Fake}) = 1, \mathsf{Pr}(\text{Fake}) = \frac{1}{2}, \mathsf{Pr}(H) = \frac{1}{2} \cdot 1 + \frac{1}{2} \cdot \frac{1}{2} = \frac{3}{4}.$

Hence $\mathsf{Pr}(\text{Fake} \mid H) = (\frac{1}{2})/(\frac{3}{4}) = \frac{2}{3}.$

Example

Two coins, one fake (two heads) one OK.

One coin chosen with equal probability and then tossed to yield a H.

What is the probability the coin was fake? Answer: $\frac{2}{3}$.

$\mathsf{Pr}(H \mid \text{Fake}) = 1, \mathsf{Pr}(\text{Fake}) = \frac{1}{2}, \mathsf{Pr}(H) = \frac{1}{2} \cdot 1 + \frac{1}{2} \cdot \frac{1}{2} = \frac{3}{4}$.

Hence $\mathsf{Pr}(\text{Fake} \mid H) = (\frac{1}{2})/(\frac{3}{4}) = \frac{2}{3}$.

Similarly $\mathsf{Pr}(\text{Fake} \mid HHH) = \frac{8}{9}$.

Example

Two coins, one fake (two heads) one OK.

One coin chosen with equal probability and then tossed to yield a H.

What is the probability the coin was fake? Answer: $\frac{2}{3}$.

$\mathsf{Pr}(H \mid \text{Fake}) = 1, \mathsf{Pr}(\text{Fake}) = \frac{1}{2}, \mathsf{Pr}(H) = \frac{1}{2} \cdot 1 + \frac{1}{2} \cdot \frac{1}{2} = \frac{3}{4}$.

Hence $\mathsf{Pr}(\text{Fake} \mid H) = (\frac{1}{2})/(\frac{3}{4}) = \frac{2}{3}$.

Similarly $\mathsf{Pr}(\text{Fake} \mid HHH) = \frac{8}{9}$. $\mathsf{Pr}(\text{Fake} \mid \underbrace{H \ldots H}_{n}) = \frac{1}{1+(\frac{1}{2})^n}$.

Bayes' rule shows how to update the *prior* probability of A with the new information that the outcome was B: this gives the *posterior* probability of A given B.

Expectation values

Expectation values

A *random variable* r is a real-valued function on X.

Expectation values

A *random variable* r is a real-valued function on X.

The *expectation value of* r is $\mathbb{E}[r] = \sum_{x \in X} \mathsf{Pr}(x) r(x)$.

Expectation values

A *random variable* r is a real-valued function on X.

The *expectation value of* r is $\mathbb{E}[r] = \sum_{x \in X} \mathsf{Pr}(x) r(x)$.

The *conditional expectation value* of r given A is:

Expectation values

A *random variable* r is a real-valued function on X.

The *expectation value of* r is $\mathbb{E}[r] = \sum_{x \in X} \mathsf{Pr}(x) r(x)$.

The *conditional expectation value* of r given A is:

$$\mathbb{E}[r \mid A] = \sum_{x \in X} r(x) \mathsf{Pr}(\{x\} \mid A).$$

Expectation values

A *random variable* r is a real-valued function on X.

The *expectation value of* r is $\mathbb{E}[r] = \sum_{x \in X} \mathsf{Pr}(x) r(x)$.

The *conditional expectation value* of r given A is:

$$\mathbb{E}[r \mid A] = \sum_{x \in X} r(x) \mathsf{Pr}(\{x\} \mid A).$$

Conditional probability is a special case of conditional expectation.

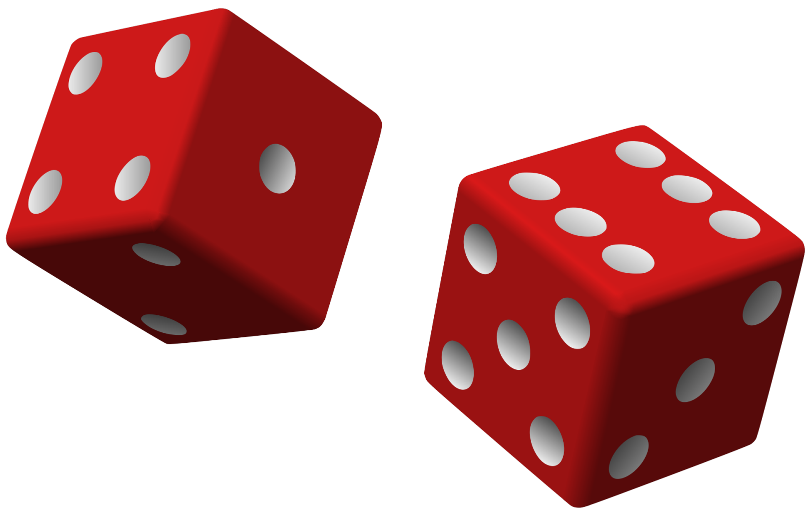
Calculating expectations through conditioning

Calculating expectations through conditioning

Two people roll dice independently.

Calculating expectations through conditioning

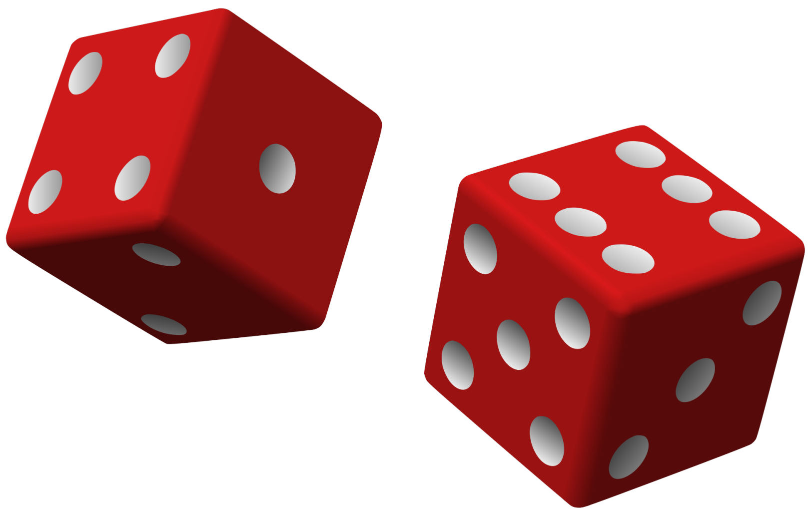
Two people roll dice independently.



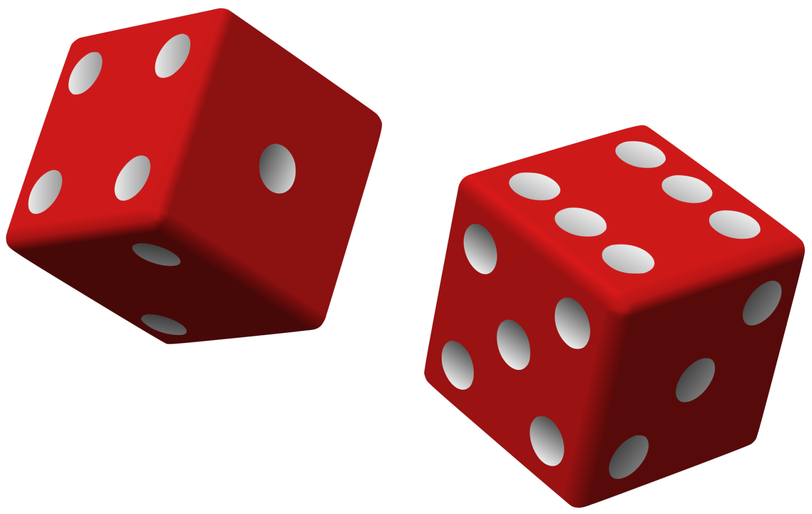
Calculating expectations through conditioning

Two people roll dice independently.

The first one keeps rolling until
he gets a 1 *immediately* followed by a 2.



Calculating expectations through conditioning



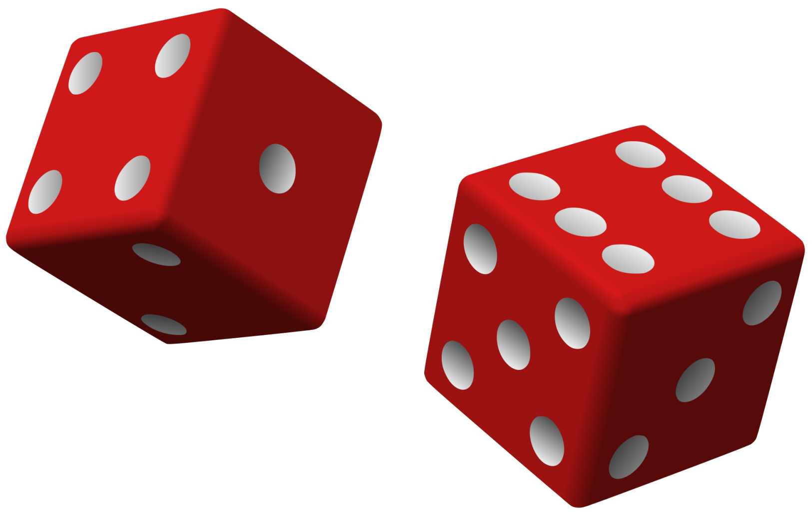
Two people roll dice independently.

The first one keeps rolling until
he gets a 1 *immediately* followed by a 2.

The second one keeps rolling
until she gets a 1 and a 1.

Calculating expectations through conditioning

Two people roll dice independently.

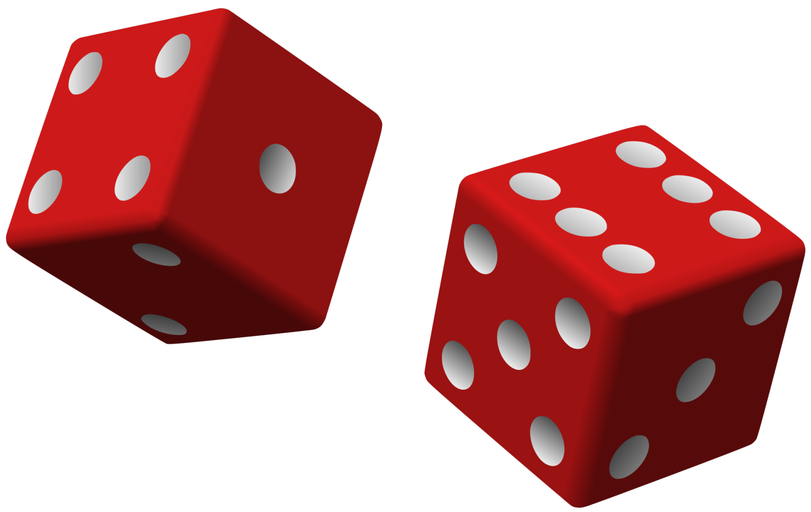


The first one keeps rolling until
he gets a 1 *immediately* followed by a 2.

The second one keeps rolling
until she gets a 1 and a 1.

Do they have the same *expected* number of rolls?

Calculating expectations through conditioning



Two people roll dice independently.

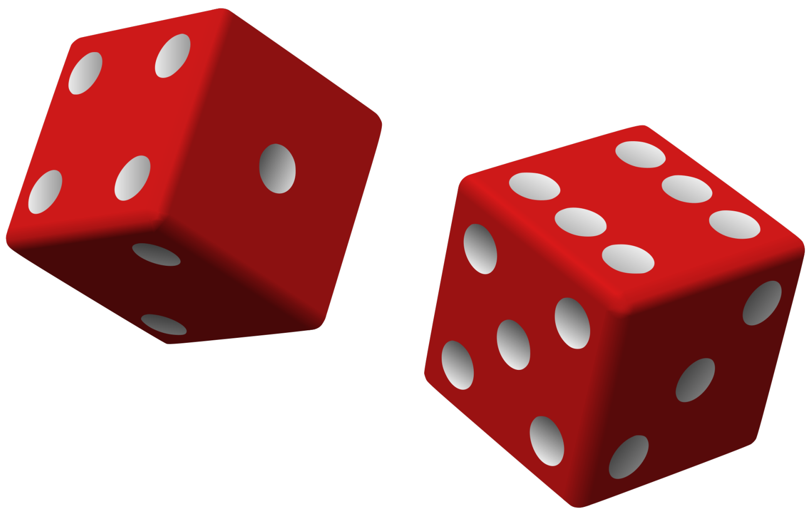
The first one keeps rolling until
he gets a 1 *immediately* followed by a 2.

The second one keeps rolling
until she gets a 1 and a 1.

Do they have the same *expected* number of rolls?

If not, who is expected to finish first?

Calculating expectations through conditioning



Two people roll dice independently.

The first one keeps rolling until
he gets a 1 *immediately* followed by a 2.

The second one keeps rolling
until she gets a 1 and a 1.

Do they have the same *expected* number of rolls?

If not, who is expected to finish first?

What is the expected number of rolls for each one?

For the first: $\frac{1}{6} \cdot (1 + \frac{1}{6} \cdot 1 + \ldots) + \ldots$

For the first: $\frac{1}{6} \cdot (1 + \frac{1}{6} \cdot 1 + \ldots) + \ldots$

Is there a better way?

For the first: $\frac{1}{6} \cdot (1 + \frac{1}{6} \cdot 1 + \ldots) + \ldots$

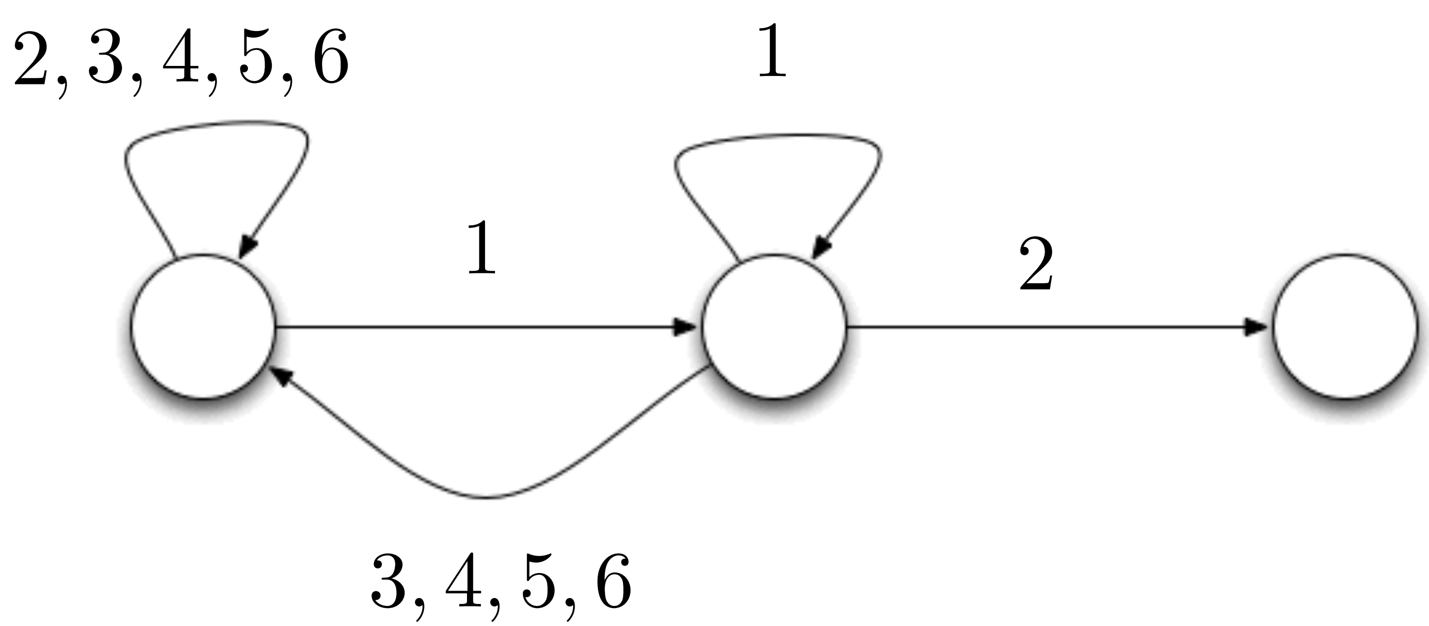
Is there a better way?

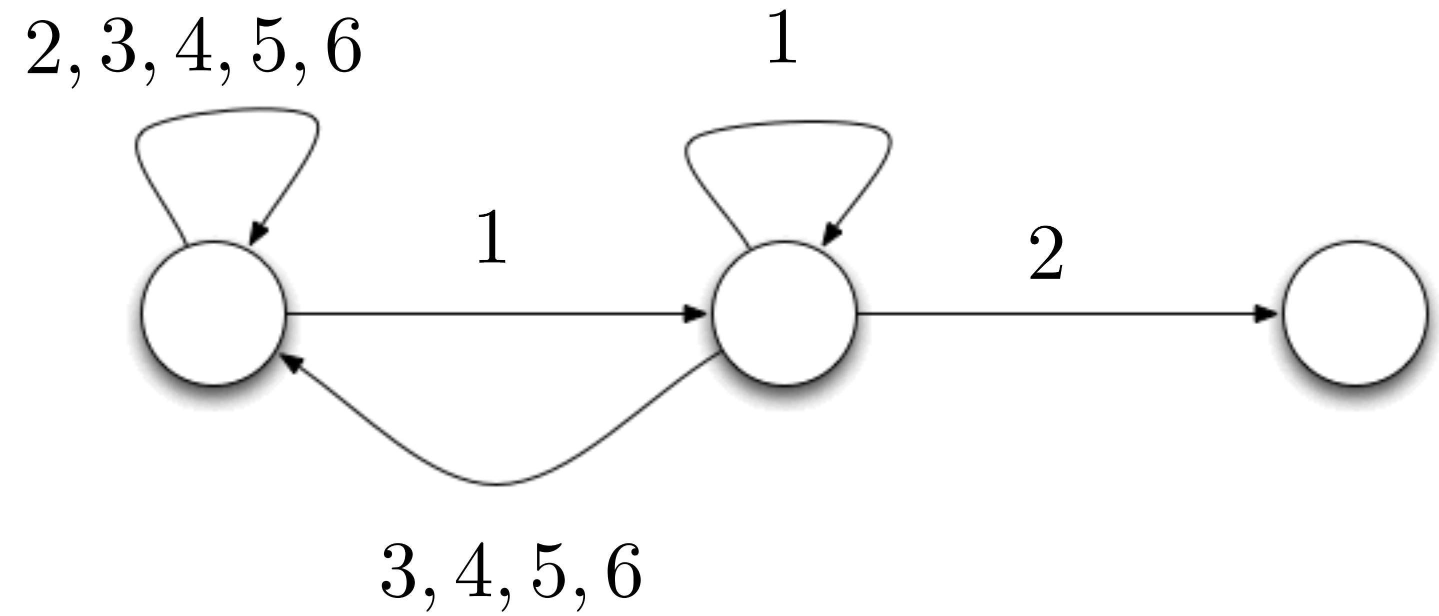
Use conditional expectations and think in terms of state-transition diagrams:

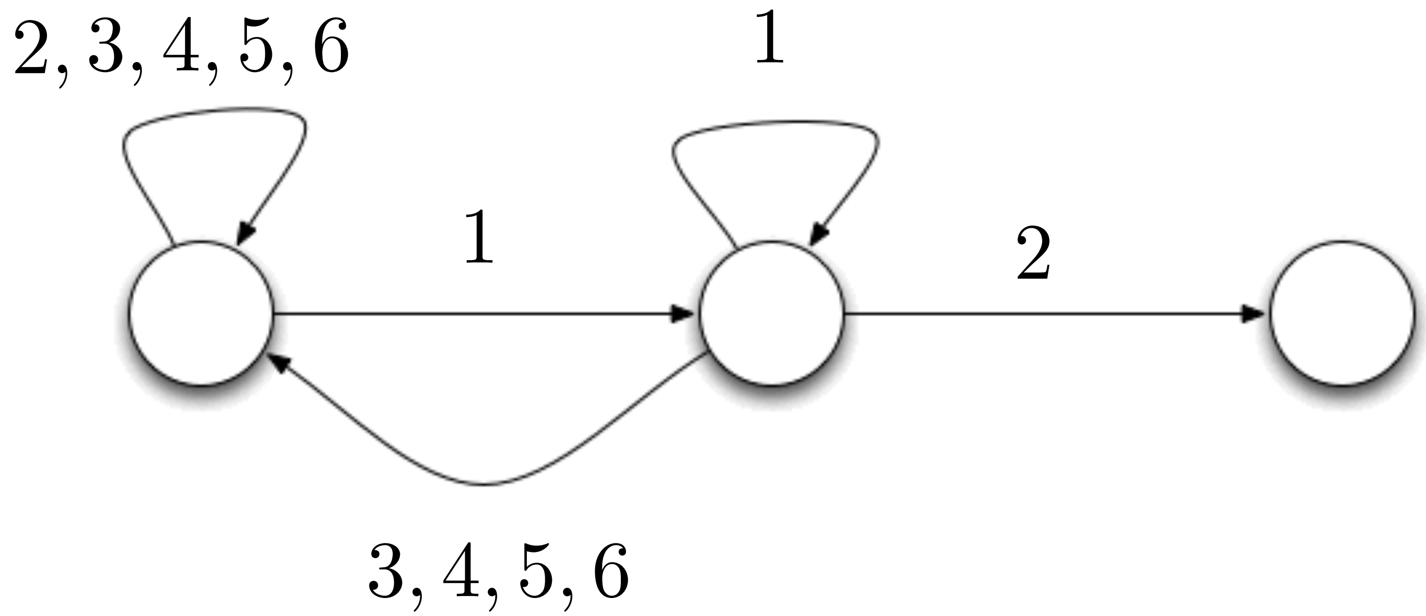
For the first: $\frac{1}{6} \cdot (1 + \frac{1}{6} \cdot 1 + \ldots) + \ldots$

Is there a better way?

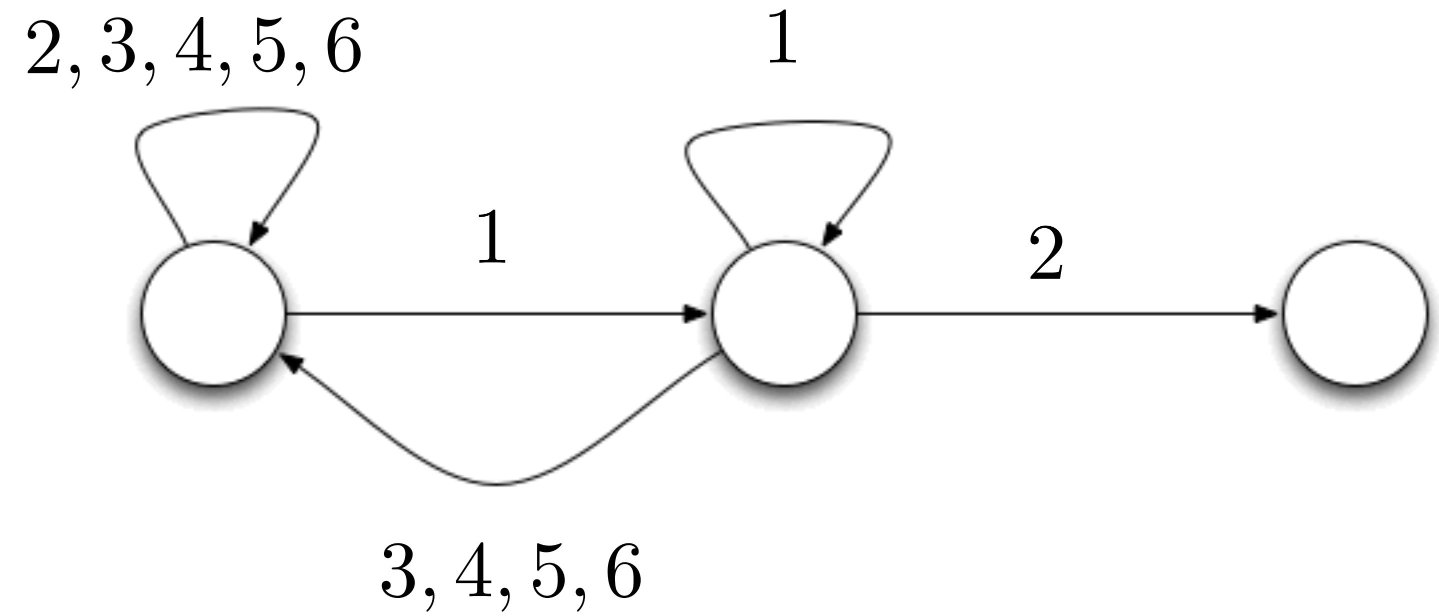
Use conditional expectations and think in terms of state-transition diagrams:



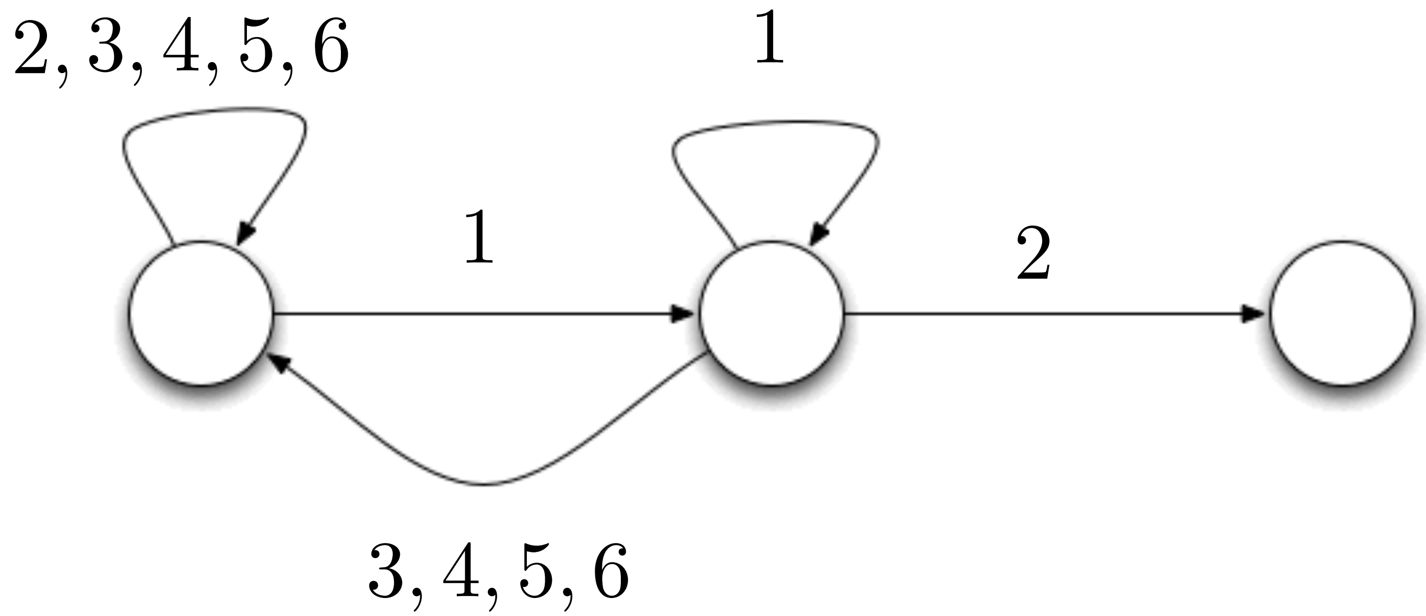

2, 3, 4, 5, 6
1
1
2
3, 4, 5, 6



Let $x = \mathbb{E}[\text{Finish} \mid \text{Start}]$

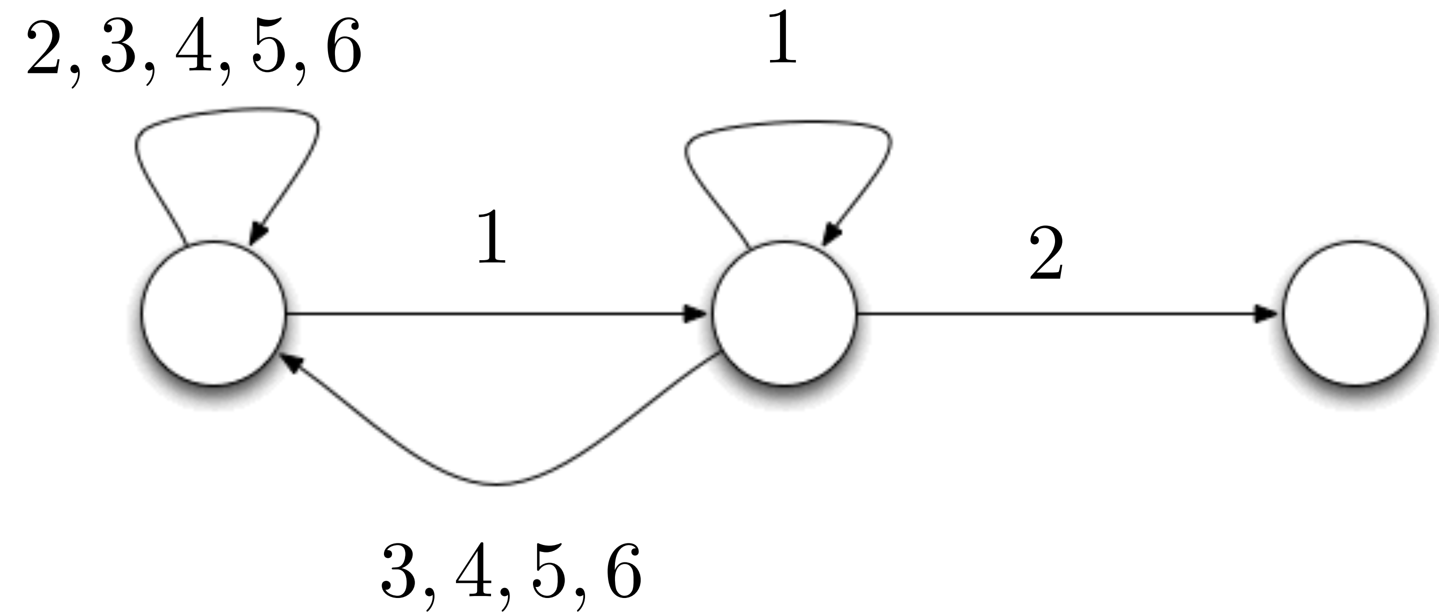


Let $x = \mathbb{E}[\text{Finish} \mid \text{Start}]$ Let $y = \mathbb{E}[\text{Finish} \mid 1]$



Let $x = \mathbb{E}[\text{Finish} \mid \text{Start}]$ Let $y = \mathbb{E}[\text{Finish} \mid 1]$

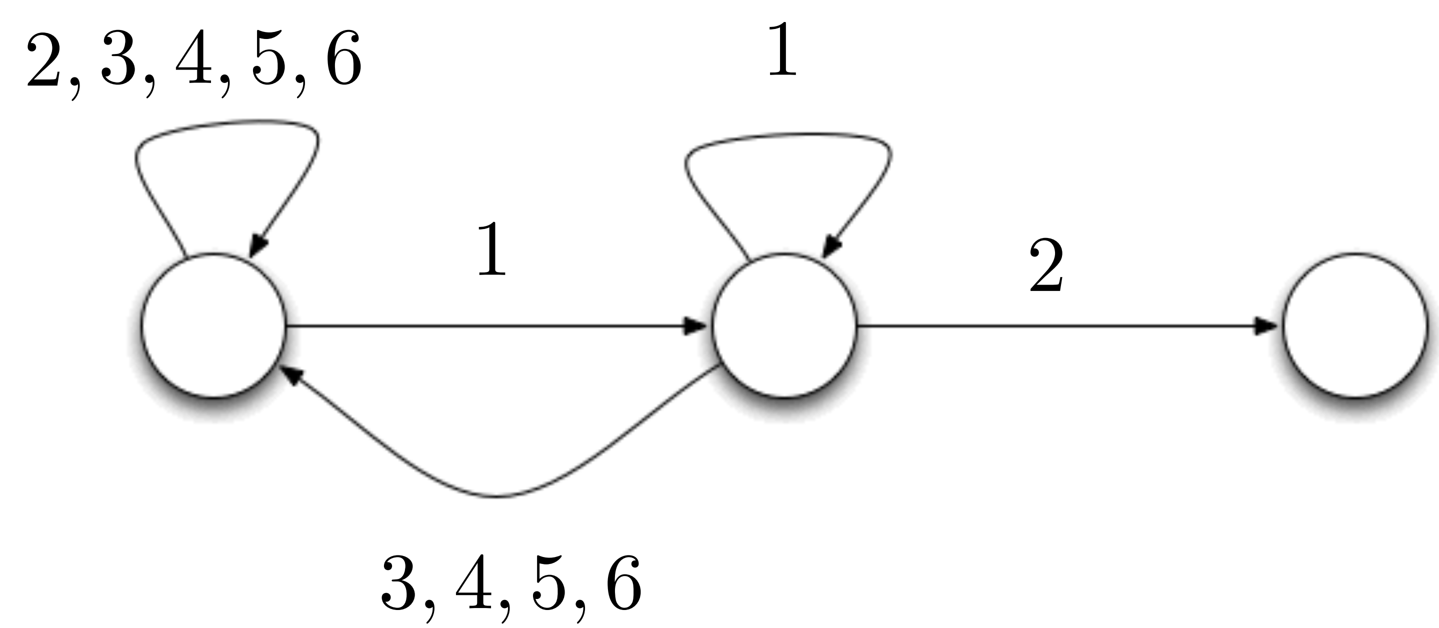
$$x = \frac{5}{6} \cdot (1 + x) + \frac{1}{6} \cdot (1 + y)$$



Let $x = \mathbb{E}[\text{Finish} \mid \text{Start}]$ Let $y = \mathbb{E}[\text{Finish} \mid 1]$

$$x = \tfrac{5}{6} \cdot (1 + x) + \tfrac{1}{6} \cdot (1 + y)$$

$$y = \tfrac{1}{6} \cdot 1 + \tfrac{1}{6} \cdot (1 + y) + \tfrac{2}{3} \cdot (1 + x)$$

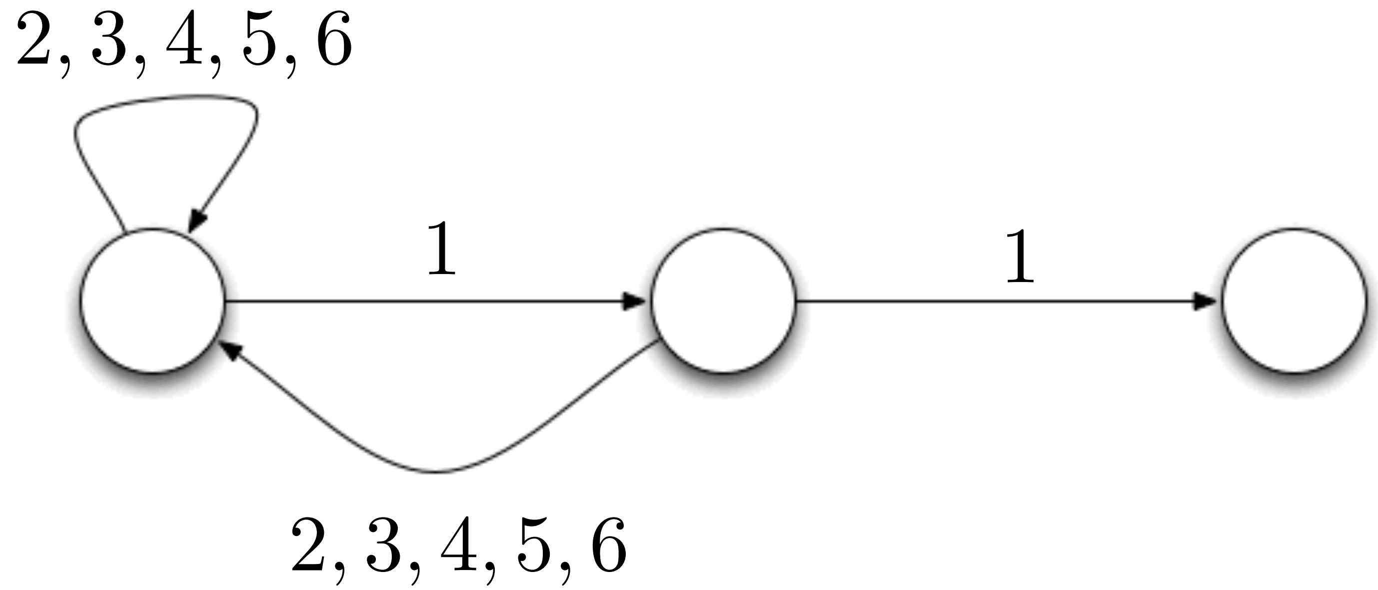


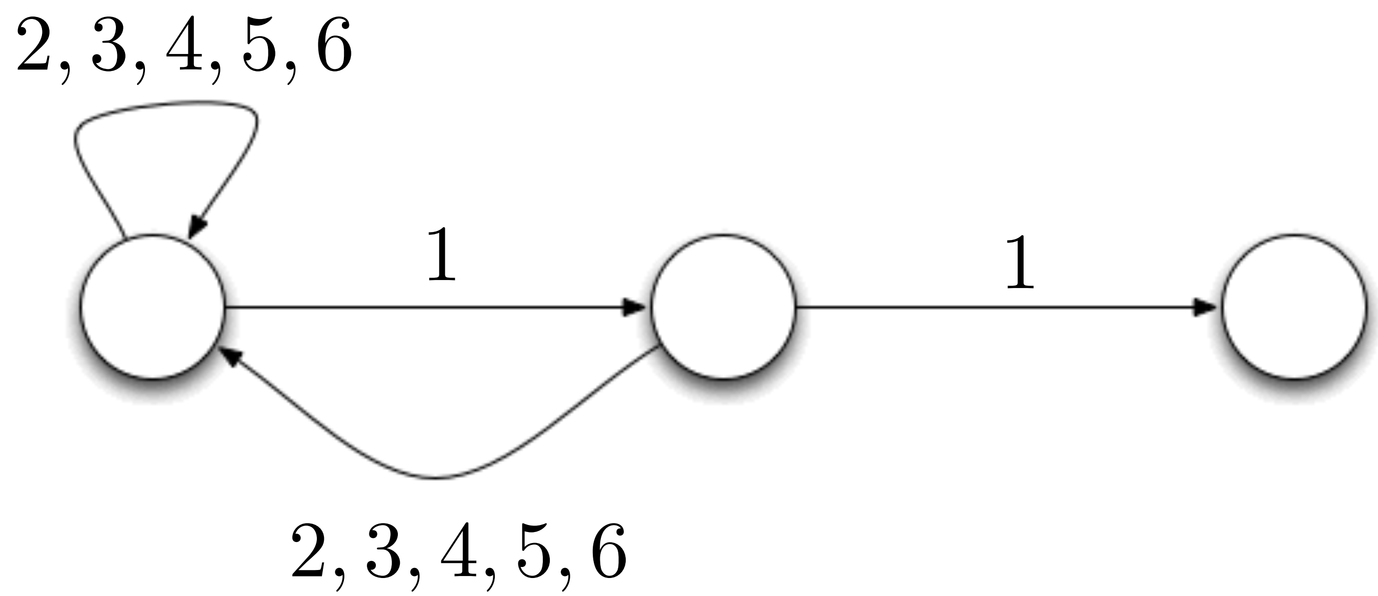
Let $x = \mathbb{E}[\text{Finish} \mid \text{Start}]$ Let $y = \mathbb{E}[\text{Finish} \mid 1]$

$$x = \frac{5}{6} \cdot (1 + x) + \frac{1}{6} \cdot (1 + y)$$

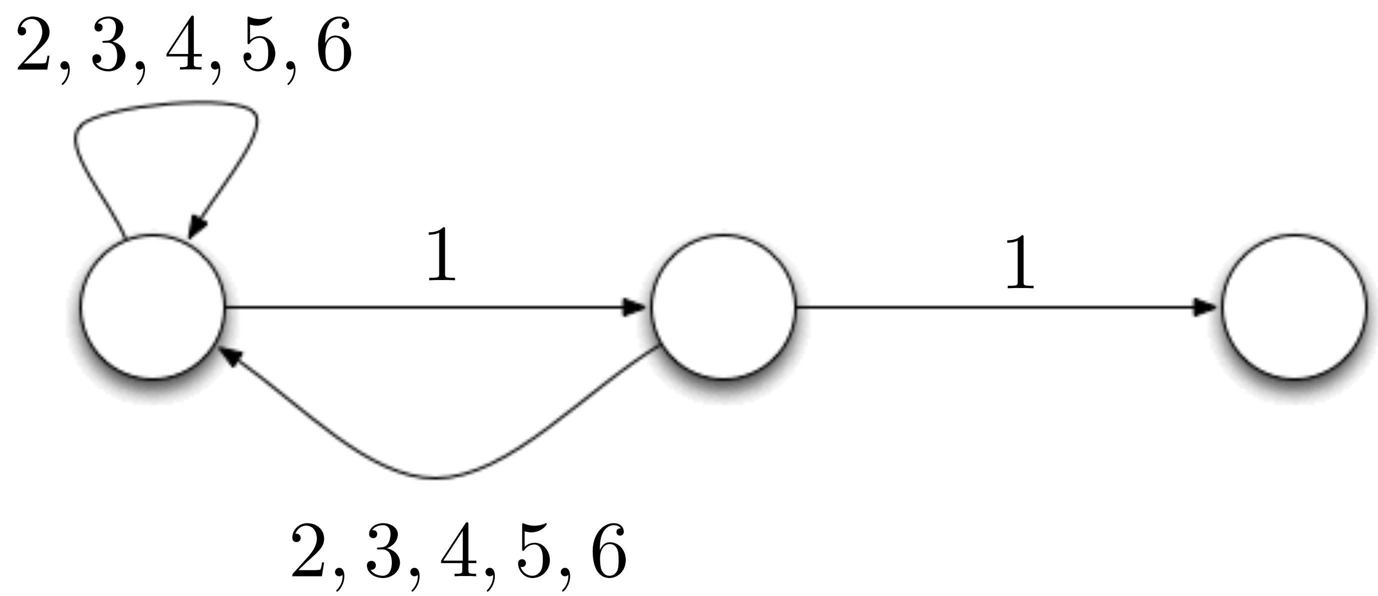
$$y = \frac{1}{6} \cdot 1 + \frac{1}{6} \cdot (1 + y) + \frac{2}{3} \cdot (1 + x)$$

Easy to solve: $x = 30, y = 36.$

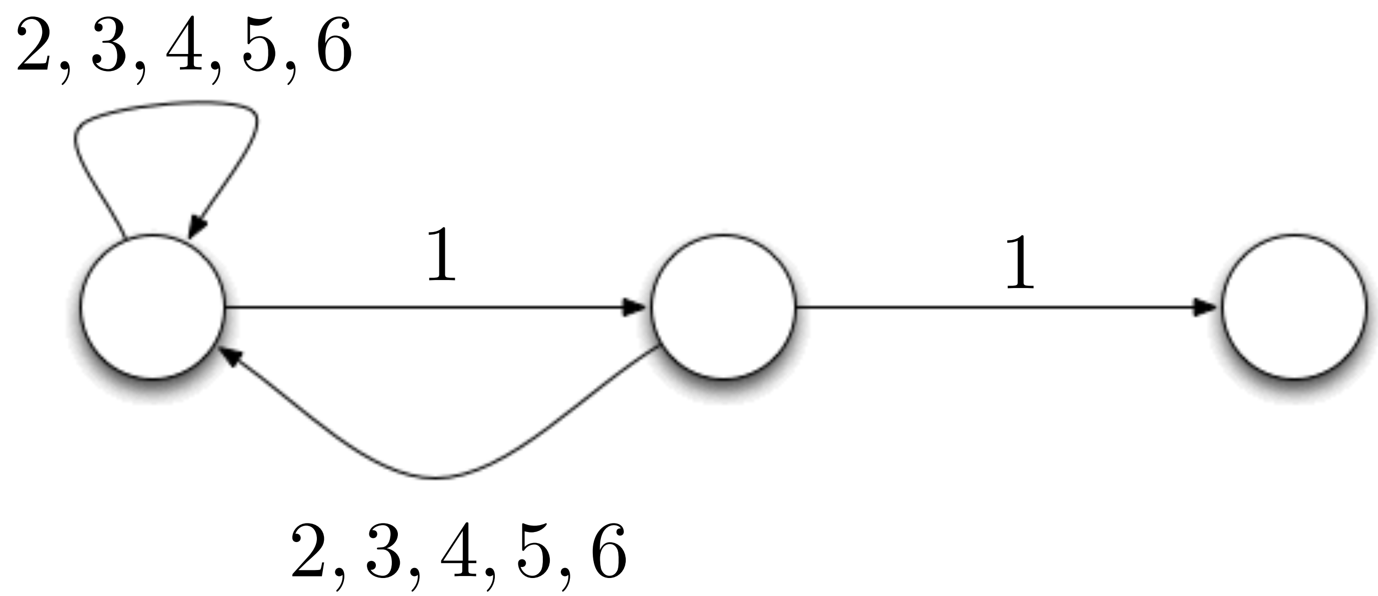

2, 3, 4, 5, 6
1
1
2, 3, 4, 5, 6



Let $x = \mathbb{E}[\text{Finish} \mid \text{Start}]$

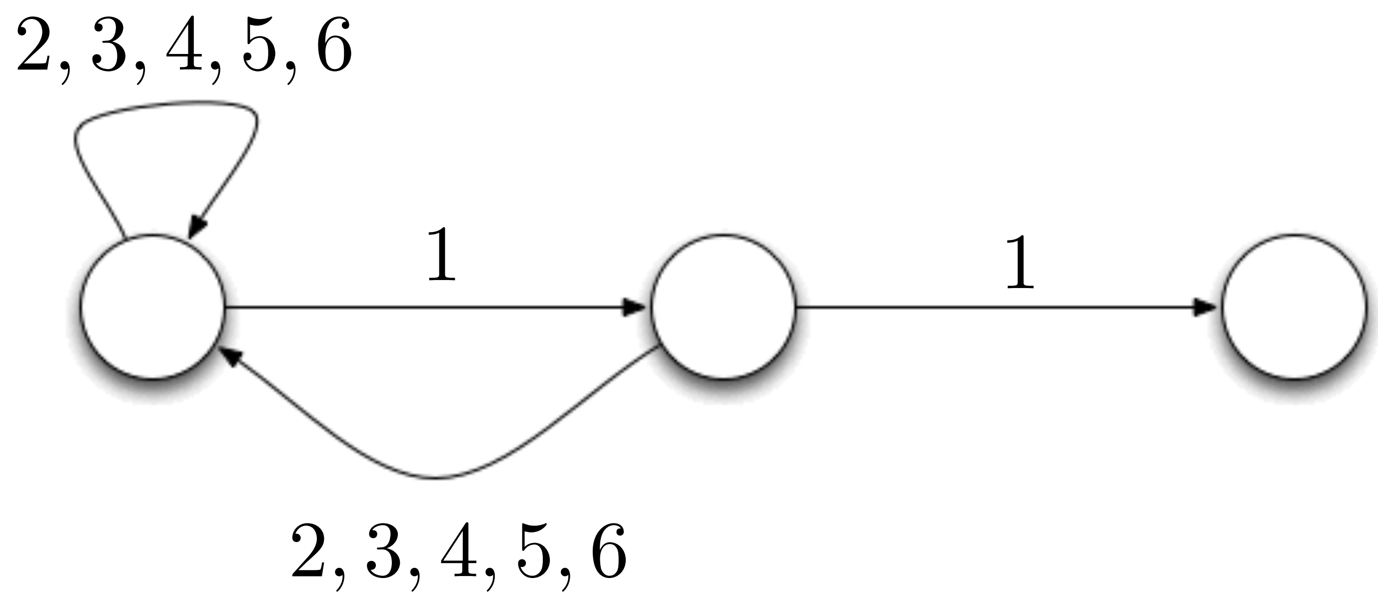


Let $x = \mathbb{E}[\text{Finish} \mid \text{Start}]$ Let $y = \mathbb{E}[\text{Finish} \mid 1]$



Let $x = \mathbb{E}[\text{Finish} \mid \text{Start}]$ Let $y = \mathbb{E}[\text{Finish} \mid 1]$

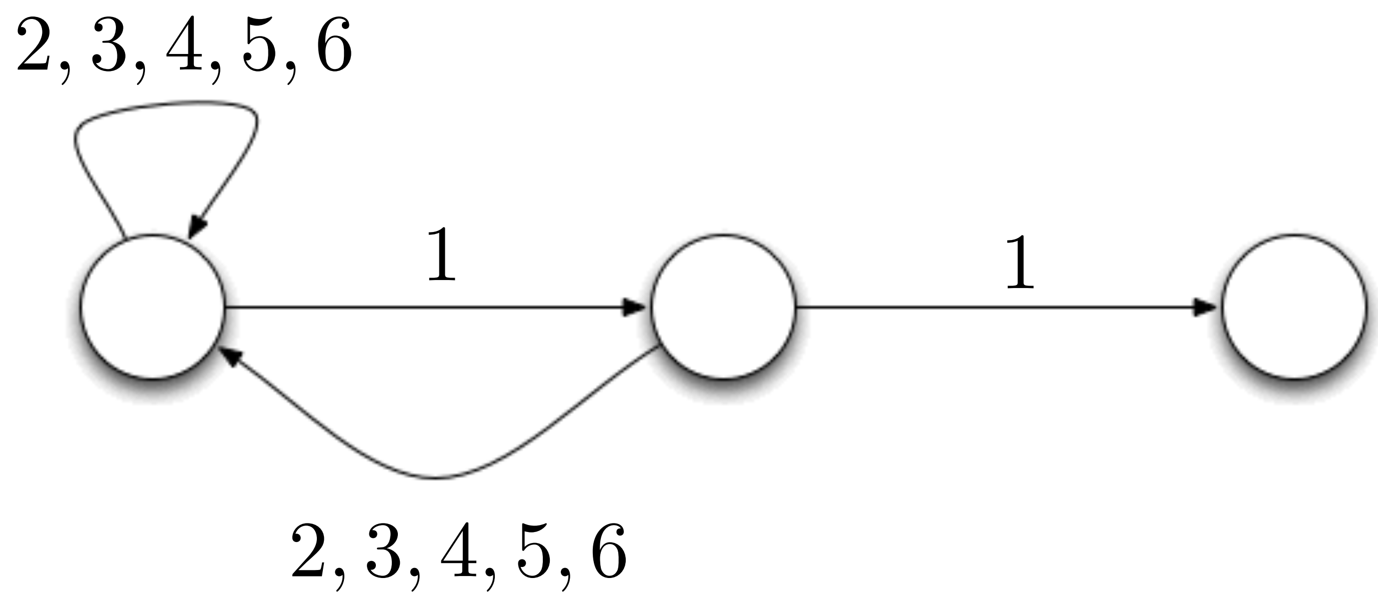
$$x = \frac{1}{6} \cdot (1 + y) + \frac{5}{6} \cdot (1 + x)$$



Let $x = \mathbb{E}[\text{Finish} \mid \text{Start}]$ Let $y = \mathbb{E}[\text{Finish} \mid 1]$

$$x = \frac{1}{6} \cdot (1 + y) + \frac{5}{6} \cdot (1 + x)$$

$$y = \frac{1}{6} \cdot 1 + \frac{5}{6} \cdot (1 + x)$$

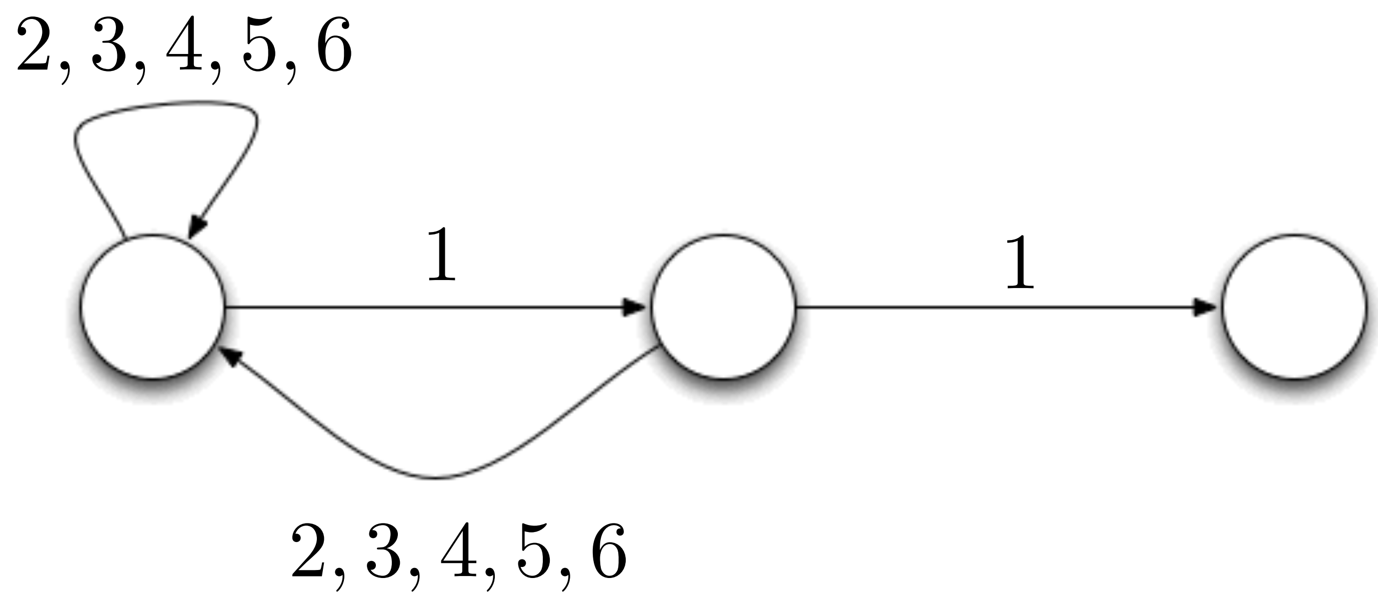


Let $x = \mathbb{E}[\text{Finish} \mid \text{Start}]$ Let $y = \mathbb{E}[\text{Finish} \mid 1]$

$$x = \tfrac{1}{6} \cdot (1 + y) + \tfrac{5}{6} \cdot (1 + x)$$

$$y = \tfrac{1}{6} \cdot 1 + \tfrac{5}{6} \cdot (1 + x)$$

Easy to solve: $x = 42, y = 36.$



Let $x = \mathbb{E}[\text{Finish} \mid \text{Start}]$ Let $y = \mathbb{E}[\text{Finish} \mid 1]$

$$x = \tfrac{1}{6} \cdot (1 + y) + \tfrac{5}{6} \cdot (1 + x)$$

$$y = \tfrac{1}{6} \cdot 1 + \tfrac{5}{6} \cdot (1 + x)$$

Easy to solve: $x = 42, y = 36.$

Did you expect this to be the slower one?

Understanding programs

Understanding programs

The *state* of a program is the correspondence between names and values.

Understanding programs

The *state* of a program is the correspondence between names and values. $[X \mapsto 3, Y \mapsto 4, Z \mapsto -2.5]$

Understanding programs

The *state* of a program is the correspondence between names and values. $[X \mapsto 3, Y \mapsto 4, Z \mapsto -2.5]$

Running a part of a program changes the state.

Understanding programs

The *state* of a program is the correspondence between names and values. $[X \mapsto 3, Y \mapsto 4, Z \mapsto -2.5]$

Running a part of a program changes the state.

$$\textsf{if } X > 1 \textsf{ then } Y = Y + Z \textsf{ else } Y = Z$$

Understanding programs

The *state* of a program is the correspondence between names and values. $[X \mapsto 3, Y \mapsto 4, Z \mapsto -2.5]$

Running a part of a program changes the state.

$$\textsf{if } X > 1 \textsf{ then } Y = Y + Z \textsf{ else } Y = Z$$

$$[X \mapsto 3, Y \mapsto 4, Z \mapsto -2.5] \longrightarrow [X \mapsto 3, Y \mapsto 1.5, Z \mapsto -2.5]$$

Understanding programs

The *state* of a program is the correspondence between names and values. $[X \mapsto 3, Y \mapsto 4, Z \mapsto -2.5]$

Running a part of a program changes the state.

$$\textsf{if } X > 1 \textsf{ then } Y = Y + Z \textsf{ else } Y = Z$$

$$[X \mapsto 3, Y \mapsto 4, Z \mapsto -2.5] \longrightarrow [X \mapsto 3, Y \mapsto 1.5, Z \mapsto -2.5]$$

Ordinary programs define state-transformer *functions*.

Understanding programs

The *state* of a program is the correspondence between names and values. $[X \mapsto 3, Y \mapsto 4, Z \mapsto -2.5]$

Running a part of a program changes the state.

$$\textsf{if } X > 1 \textsf{ then } Y = Y + Z \textsf{ else } Y = Z$$

$$[X \mapsto 3, Y \mapsto 4, Z \mapsto -2.5] \longrightarrow [X \mapsto 3, Y \mapsto 1.5, Z \mapsto -2.5]$$

Ordinary programs define state-transformer *functions*.

When one combines program pieces one can *compose* the functions to find the combined effect.

Understanding programs

The *state* of a program is the correspondence between names and values. $[X \mapsto 3, Y \mapsto 4, Z \mapsto -2.5]$

Running a part of a program changes the state.

$$\textsf{if } X > 1 \textsf{ then } Y = Y + Z \textsf{ else } Y = Z$$

$$[X \mapsto 3, Y \mapsto 4, Z \mapsto -2.5] \longrightarrow [X \mapsto 3, Y \mapsto 1.5, Z \mapsto -2.5]$$

Ordinary programs define state-transformer *functions*.

When one combines program pieces one can *compose* the functions to find the combined effect.

How do we understand probabilistic programs?

Understanding programs

The *state* of a program is the correspondence between names and values. $[X \mapsto 3, Y \mapsto 4, Z \mapsto -2.5]$

Running a part of a program changes the state.

$$\textsf{if } X > 1 \textsf{ then } Y = Y + Z \textsf{ else } Y = Z$$

$$[X \mapsto 3, Y \mapsto 4, Z \mapsto -2.5] \longrightarrow [X \mapsto 3, Y \mapsto 1.5, Z \mapsto -2.5]$$

Ordinary programs define state-transformer *functions*.

When one combines program pieces one can *compose* the functions to find the combined effect.

How do we understand probabilistic programs?

As distribution transformers.

$X = 0$; $C =$ toss; if $C = 1$ then $X = X + 1$ else $X = X - 1$.

$$X = 0;\ C = \mathsf{toss};\ \mathsf{if}\ C = 1\ \mathsf{then}\ X = X + 1\ \mathsf{else}\ X = X - 1.$$

$$\text{Initial distribution: } [X \mapsto (0, 1.0), C \mapsto (0, 1.0)]$$

$$X = 0;\ C = \mathsf{toss};\ \mathsf{if}\ C = 1\ \mathsf{then}\ X = X + 1\ \mathsf{else}\ X = X - 1.$$

$$\text{Initial distribution: } [X \mapsto (0, 1.0), C \mapsto (0, 1.0)]$$

$$\text{Final distribution: } [X \mapsto (1, 0.5)(-1, 0.5), C \mapsto (0, 0.5)(1, 0.5)]$$

$X = 0$; $C = \textsf{toss}$; $\textsf{if } C = 1 \textsf{ then } X = X + 1 \textsf{ else } X = X - 1.$

Initial distribution: $[X \mapsto (0, 1.0), C \mapsto (0, 1.0)]$

Final distribution: $[X \mapsto (1, 0.5)(-1, 0.5), C \mapsto (0, 0.5)(1, 0.5)]$

A Markov chain has S: states and a probability distribution transformer T.

$$X = 0;\ C = \textsf{toss};\ \textsf{if}\ C = 1\ \textsf{then}\ X = X + 1\ \textsf{else}\ X = X - 1.$$

$$\text{Initial distribution: } [X \mapsto (0, 1.0), C \mapsto (0, 1.0)]$$

$$\text{Final distribution: } [X \mapsto (1, 0.5)(-1, 0.5), C \mapsto (0, 0.5)(1, 0.5)]$$

A Markov chain has S: states and a probability distribution transformer T.

$$T : St \times St \to [0, 1] \text{ or } T : St \to Dist(St).$$

$$X = 0;\ C = \mathsf{toss};\ \mathsf{if}\ C = 1\ \mathsf{then}\ X = X + 1\ \mathsf{else}\ X = X - 1.$$

$$\text{Initial distribution: } [X \mapsto (0, 1.0), C \mapsto (0, 1.0)]$$

$$\text{Final distribution: } [X \mapsto (1, 0.5)(-1, 0.5), C \mapsto (0, 0.5)(1, 0.5)]$$

A Markov chain has S: states and a probability distribution transformer T.

$$T : St \times St \to [0, 1] \text{ or } T : St \to Dist(St).$$

$T(s_1, s_2)$ is the conditional probability of being in state s_2 after the transition given that the state was s_1 before.

$X = 0$; $C = \mathsf{toss}$; $\mathsf{if}\ C = 1\ \mathsf{then}\ X = X + 1\ \mathsf{else}\ X = X - 1$.

Initial distribution: $[X \mapsto (0, 1.0), C \mapsto (0, 1.0)]$

Final distribution: $[X \mapsto (1, 0.5)(-1, 0.5), C \mapsto (0, 0.5)(1, 0.5)]$

A Markov chain has S: states and a probability distribution transformer T.

$$T : St \times St \to [0, 1] \text{ or } T : St \to Dist(St).$$

$T(s_1, s_2)$ is the conditional probability of being in state s_2 after the transition given that the state was s_1 before.

Markov property: the transition probability only depends on the current state, not on the whole history.

Because of the Markov property, one can describe the effect of a transition by a matrix.

Because of the Markov property, one can describe the effect of a transition by a matrix.

When one combines probabilistic program pieces one can *multiply* the transition matrices to find the combined effect.

Because of the Markov property, one can describe the effect of a transition by a matrix.

When one combines probabilistic program pieces one can *multiply* the transition matrices to find the combined effect.

We are understanding the program by stepping forwards.

Because of the Markov property, one can describe the effect of a transition by a matrix.

When one combines probabilistic program pieces one can *multiply* the transition matrices to find the combined effect.

We are understanding the program by stepping forwards.

This is called “forwards” or state-transformer semantics.

Backwards semantics

Backwards semantics

Maybe we do not want to track every detail of the state as it changes.

Backwards semantics

Maybe we do not want to track every detail of the state as it changes.

Perhaps we want to know if a *property* holds, *e.g.* $X > 0$.

Backwards semantics

Maybe we do not want to track every detail of the state as it changes.

Perhaps we want to know if a *property* holds, *e.g.* $X > 0$.

We write $\{P\}$ step $\{Q\}$ to mean that P holds before the step and Q holds after the step.

Backwards semantics

Maybe we do not want to track every detail of the state as it changes.

Perhaps we want to know if a *property* holds, *e.g.* $X > 0$.

We write $\{P\}$ step $\{Q\}$ to mean that P holds before the step and Q holds after the step.

$$\{X > 0\}\ X = X - 5\ \{X > 0\}\ ??$$

Backwards semantics

Maybe we do not want to track every detail of the state as it changes.

Perhaps we want to know if a *property* holds, *e.g.* $X > 0$.

We write $\{P\}$ step $\{Q\}$ to mean that P holds before the step and Q holds after the step.

$$\{X > 0\}\ X = X - 5\ \{X > 0\}\ ??$$

We cannot say anything for sure after the step!

But we can go backwards!

But we can go backwards!

$$\{X > 5\}\ X = X - 5\ \{X > 0\}$$

But we can go backwards!

$$\{X > 5\}\ X = X - 5\ \{X > 0\}$$

But we can go backwards!

$$\{X > 5\}\ X = X - 5\ \{X > 0\}$$

We must read this differently: if we want $X > 0$ after the step, we must make sure $X > 5$ before the step.

But we can go backwards!

$$\{X > 5\}\ X = X - 5\ \{X > 0\}$$

We must read this differently: if we want $X > 0$ after the step, we must make sure $X > 5$ before the step.

This is called “predicate-transformer” semantics.

But we can go backwards!

$$\{X > 5\}\ X = X - 5\ \{X > 0\}$$

We must read this differently: if we want $X > 0$ after the step, we must make sure $X > 5$ before the step.

This is called "predicate-transformer" semantics.

Can we do something like this for probabilistic programs?

Classical logic	Generalization
Truth values $\{0, 1\}$	Probabilities $[0, 1]$
Predicate	Random variable
State	Distribution
The satisfaction relation $\models$	Integration $\int$

Instead of predicates we have real-valued functions.

Instead of predicates we have real-valued functions.

Suppose that your probabilistic program describes a search for some resource.

Instead of predicates we have real-valued functions.

Suppose that your probabilistic program describes a search for some resource.

Suppose that $r : St \to \mathbb{R}$ is the "expected reward" in each state.

Instead of predicates we have real-valued functions.

Suppose that your probabilistic program describes a search for some resource.

Suppose that $r : St \to \mathbb{R}$ is the "expected reward" in each state.

We write $\hat{T}r$ for a new reward function defined by

$$(\hat{T}r)(s) = \sum_{s' \in S} T(s, s')r(s').$$

Instead of predicates we have real-valued functions.

Suppose that your probabilistic program describes a search for some resource.

Suppose that $r : St \to \mathbb{R}$ is the "expected reward" in each state.

We write $\hat{T}r$ for a new reward function defined by

$$(\hat{T}r)(s) = \sum_{s' \in S} T(s, s')r(s').$$

This tells you the expected reward *before* the transition assuming that r is the reward after the transition.

Probabilistic Models

Probabilistic Models

- A general framework to reason about situations computationally and quantitatively.

Probabilistic Models

- A general framework to reason about situations computationally and quantitatively.
- Most important class of models: graphical models.

Probabilistic Models

- A general framework to reason about situations computationally and quantitatively.
- Most important class of models: graphical models.
- They capture dependence and independence and

Probabilistic Models

- A general framework to reason about situations computationally and quantitatively.
- Most important class of models: graphical models.
- They capture dependence and independence and
- conditional independence.

Complex systems

Complex systems

- Complex systems have many variables.

Complex systems

- Complex systems have many variables.
- They are related in intricate ways: correlated or independent.

Complex systems

- Complex systems have many variables.
- They are related in intricate ways: correlated or independent.
- They may be conditionally independent or dependent.

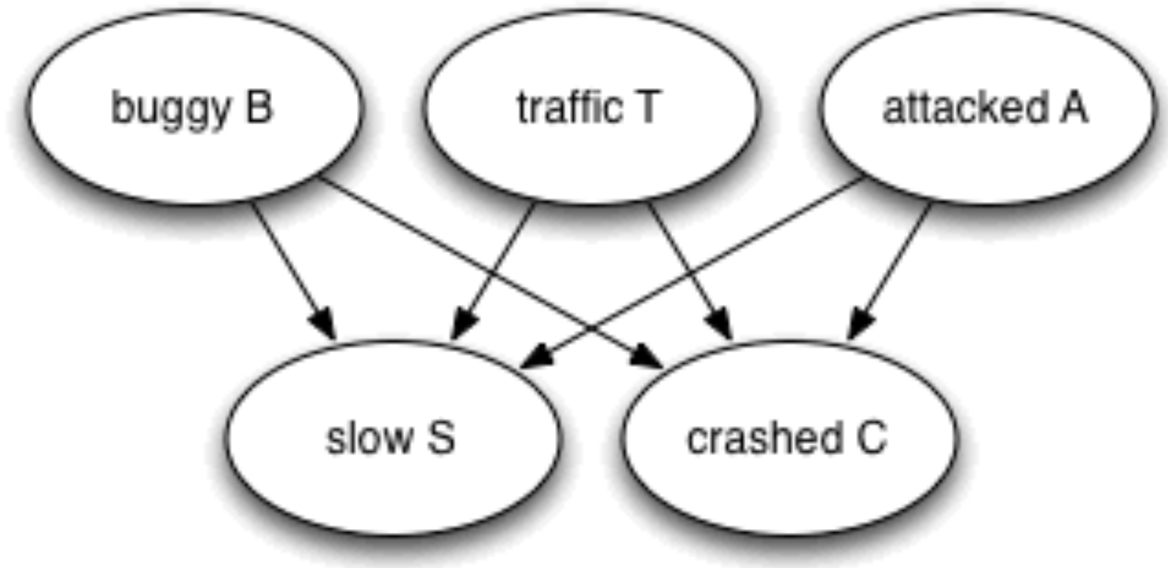
Complex systems

- Complex systems have many variables.
- They are related in intricate ways: correlated or independent.
- They may be conditionally independent or dependent.
- There may be causal connections.

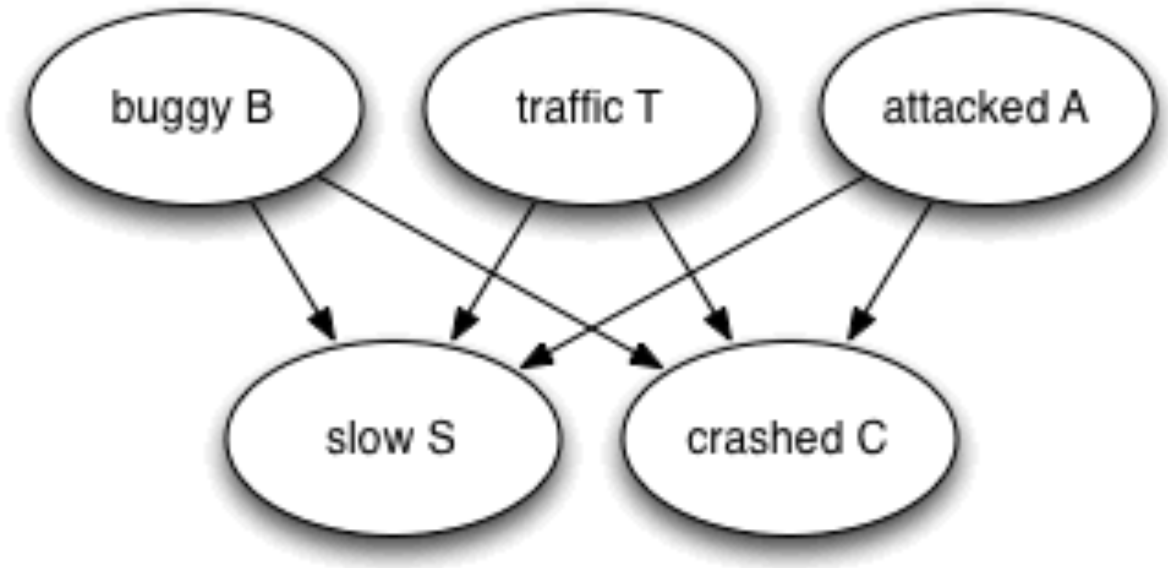
Complex systems

- Complex systems have many variables.
- They are related in intricate ways: correlated or independent.
- They may be conditionally independent or dependent.
- There may be causal connections.
- We want to represent several random variables that may be connected in different ways.

A simple graphical model

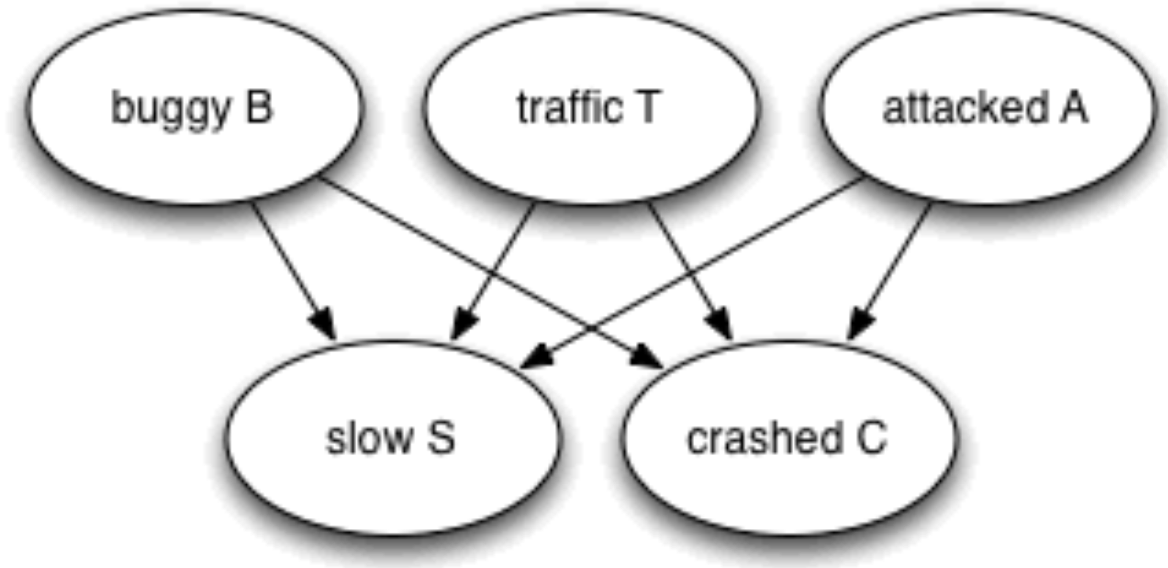


A simple graphical model



Trying to analyze what is wrong with a web site:

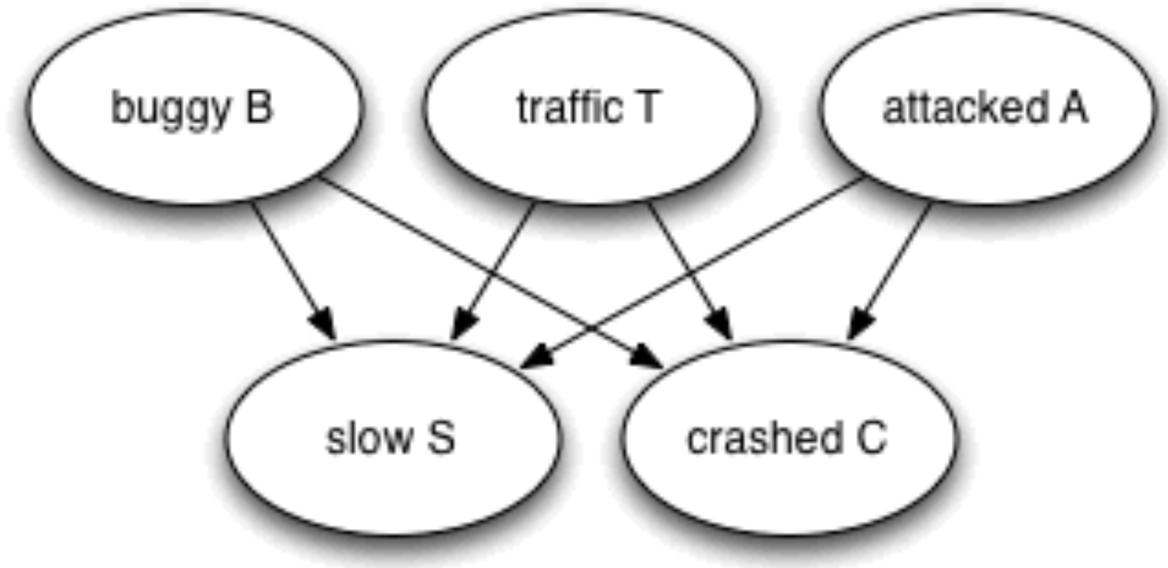
A simple graphical model



Trying to analyze what is wrong with a web site:

could be buggy (yes/no)

A simple graphical model

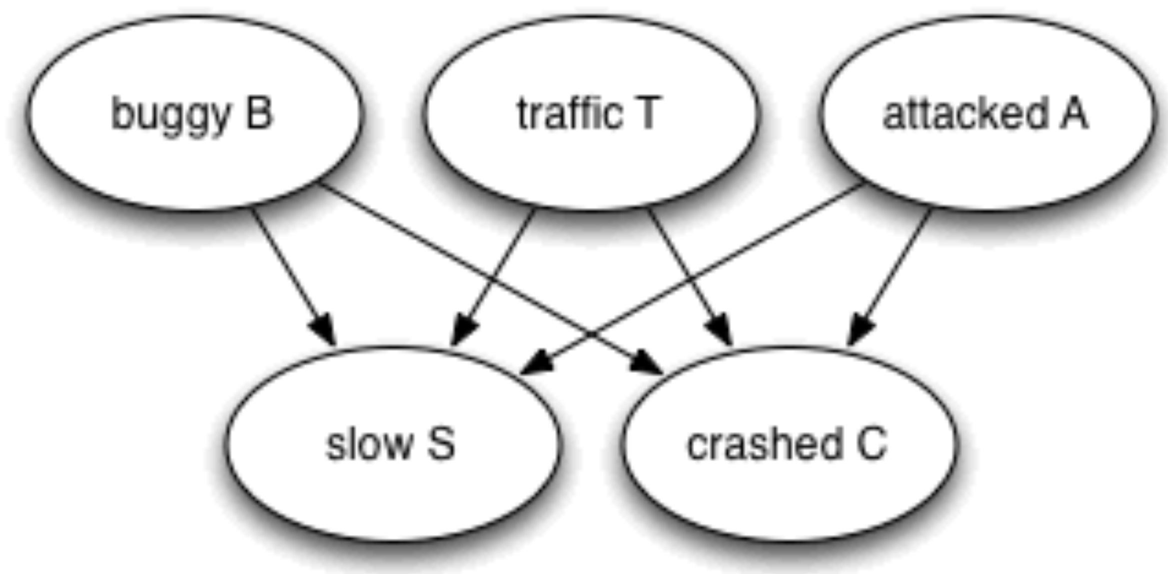


Trying to analyze what is wrong with a web site:

could be buggy (yes/no)

could be under attack (yes/no)

A simple graphical model



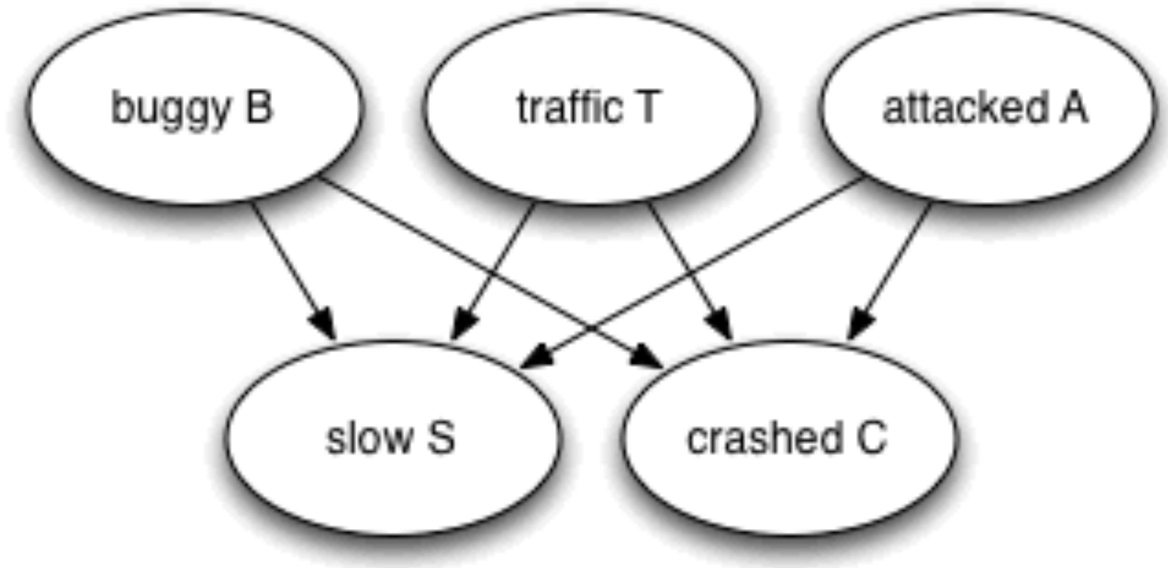
Trying to analyze what is wrong with a web site:

could be buggy (yes/no)

could be under attack (yes/no)

traffic (very high/high/moderate/low)

A simple graphical model



Trying to analyze what is wrong with a web site:

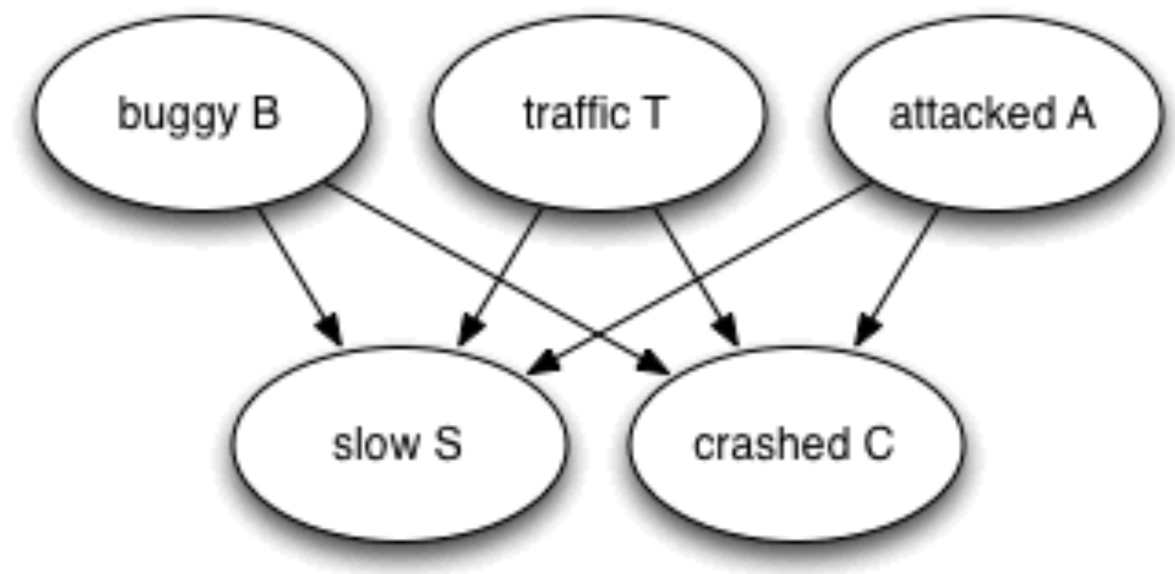
could be buggy (yes/no)

could be under attack (yes/no)

traffic (very high/high/moderate/low)

Symptoms: crashed or slow

A simple graphical model



Trying to analyze what is wrong with a web site:

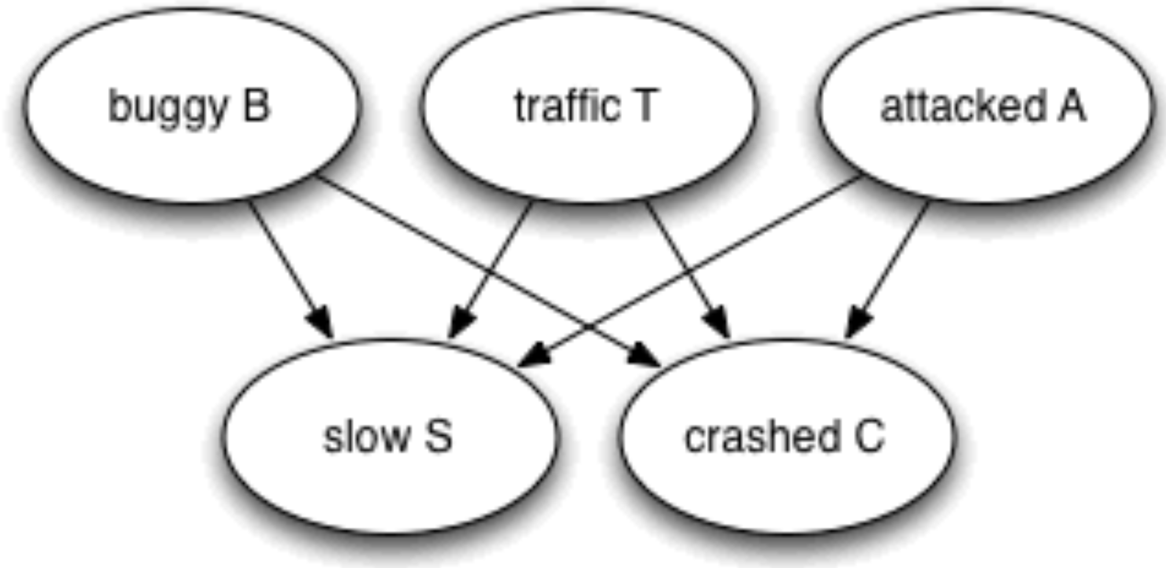
could be buggy (yes/no)

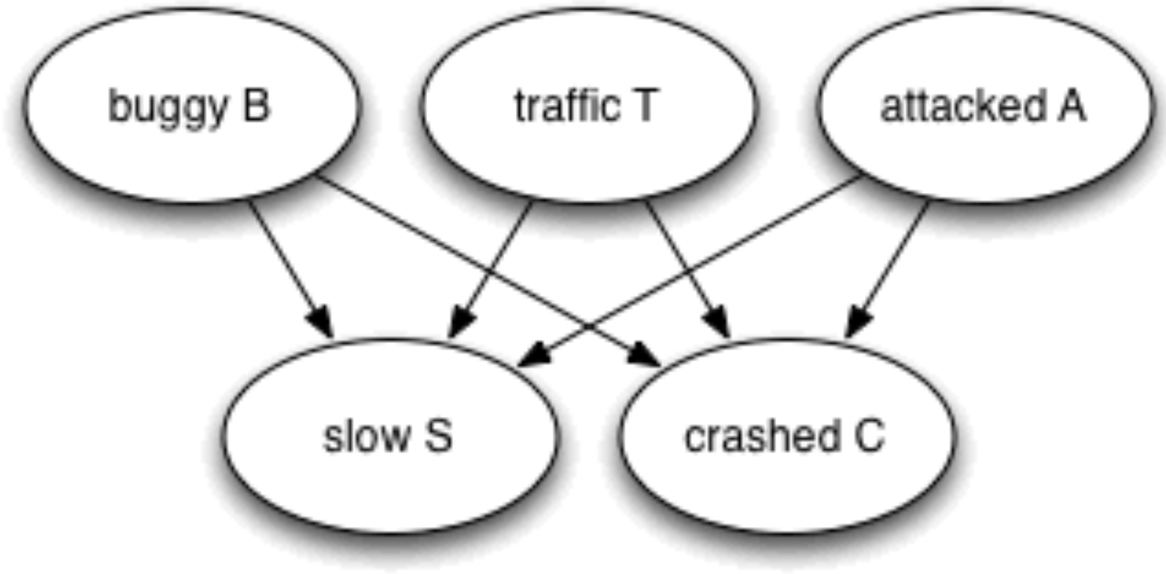
could be under attack (yes/no)

traffic (very high/high/moderate/low)

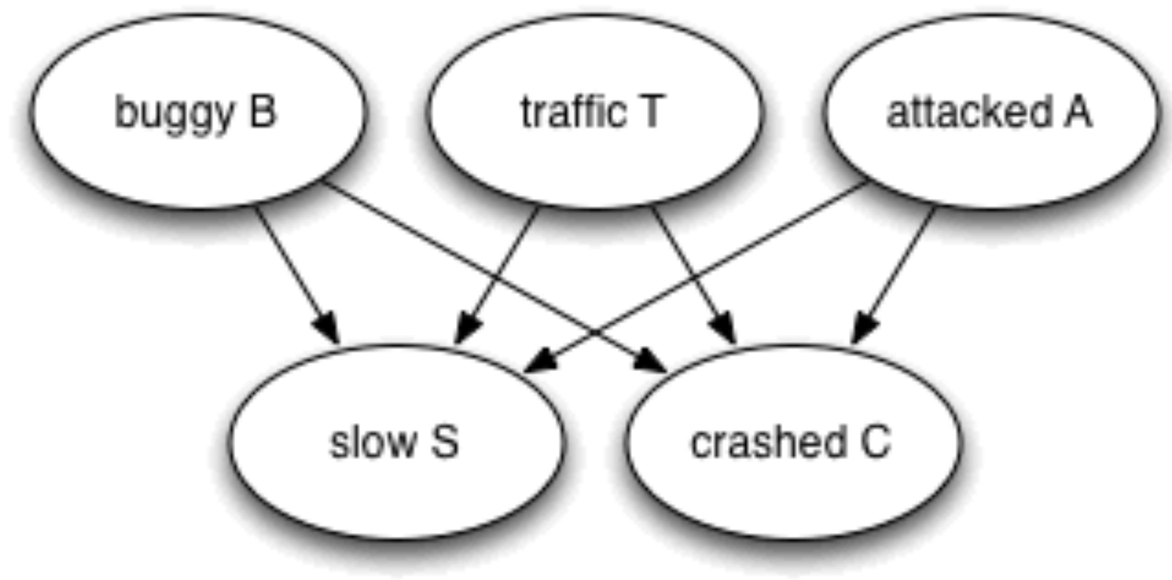
Symptoms: crashed or slow

There are 64 states.


buggy B
traffic T
attacked A
slow S
crashed C

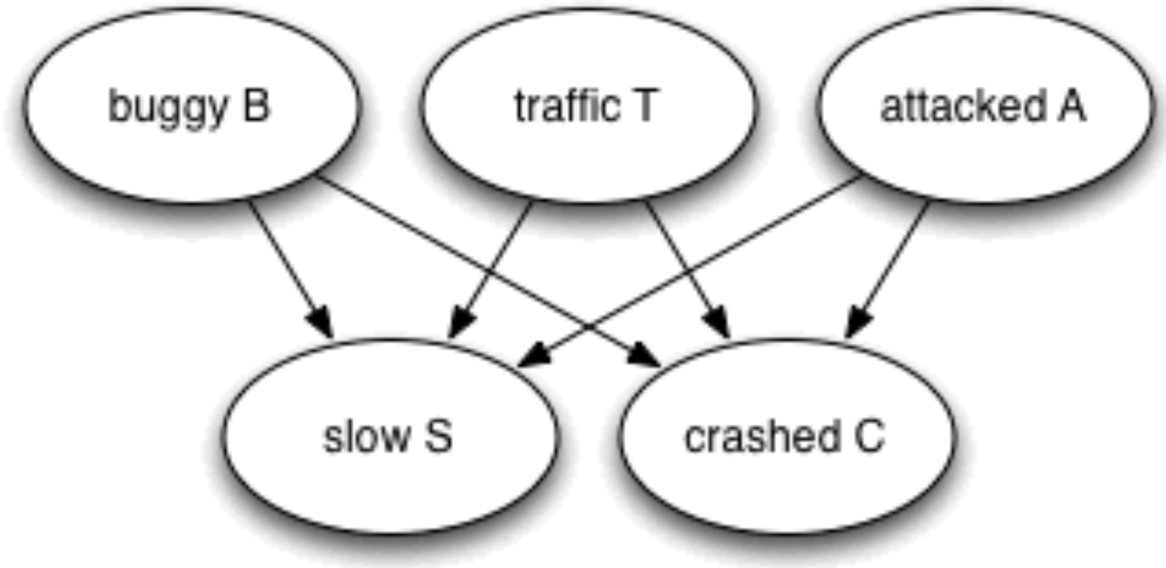


The arrows show probabilistic dependencies.



The arrows show probabilistic dependencies.

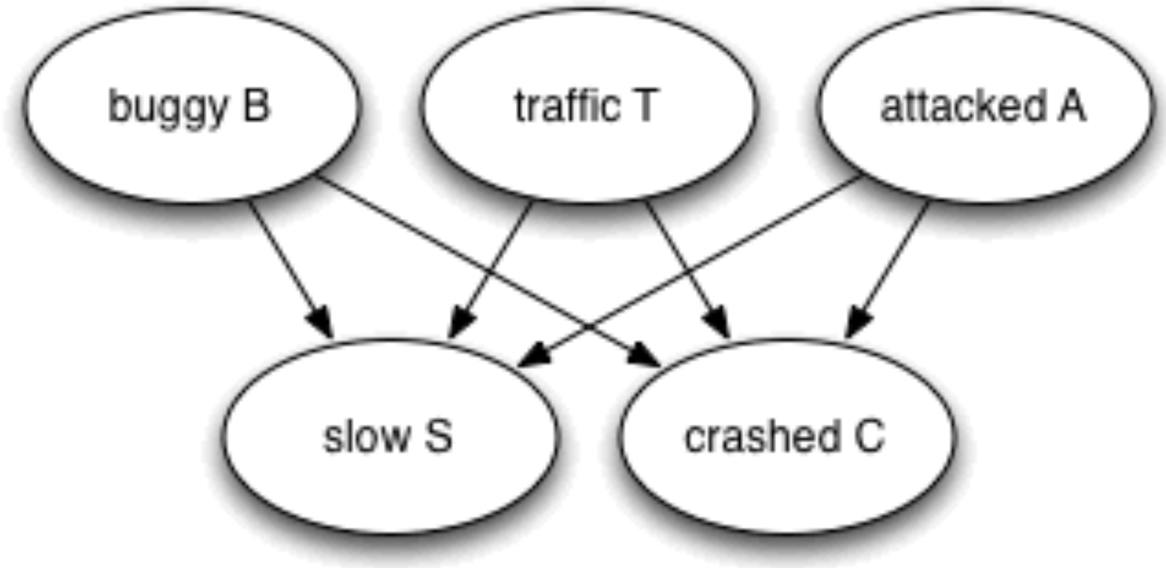
Everything in the top row is independent of each other.



The arrows show probabilistic dependencies.

Everything in the top row is independent of each other.

We write $A \perp B$ for A is independent of B.

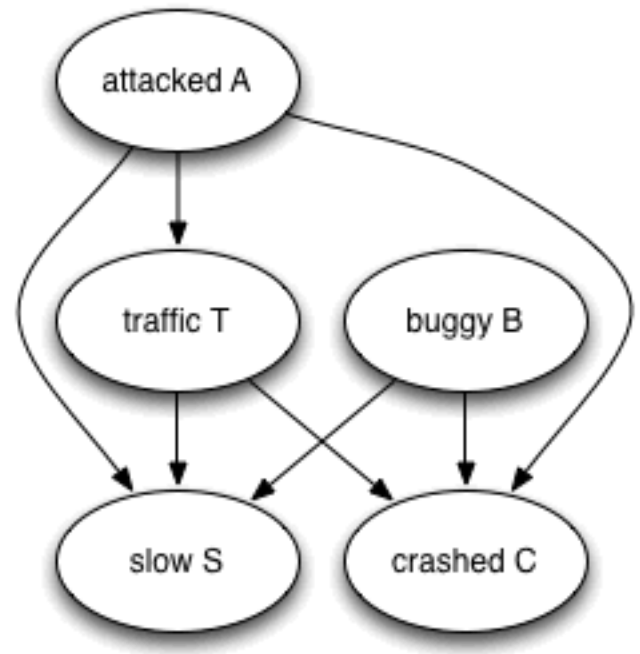


The arrows show probabilistic dependencies.

Everything in the top row is independent of each other.

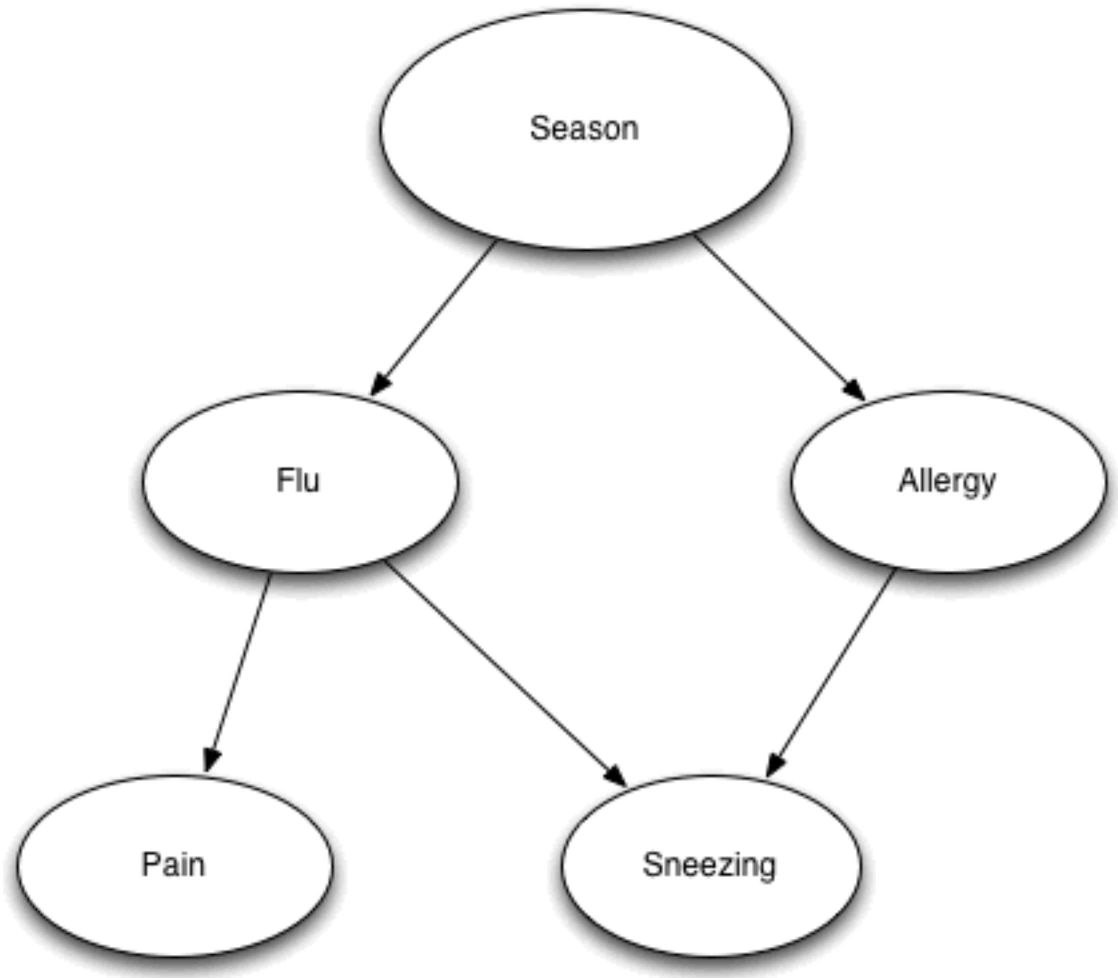
We write $A \perp B$ for A is independent of B.

Perhaps too simple:
if attacked then the traffic should be heavy.

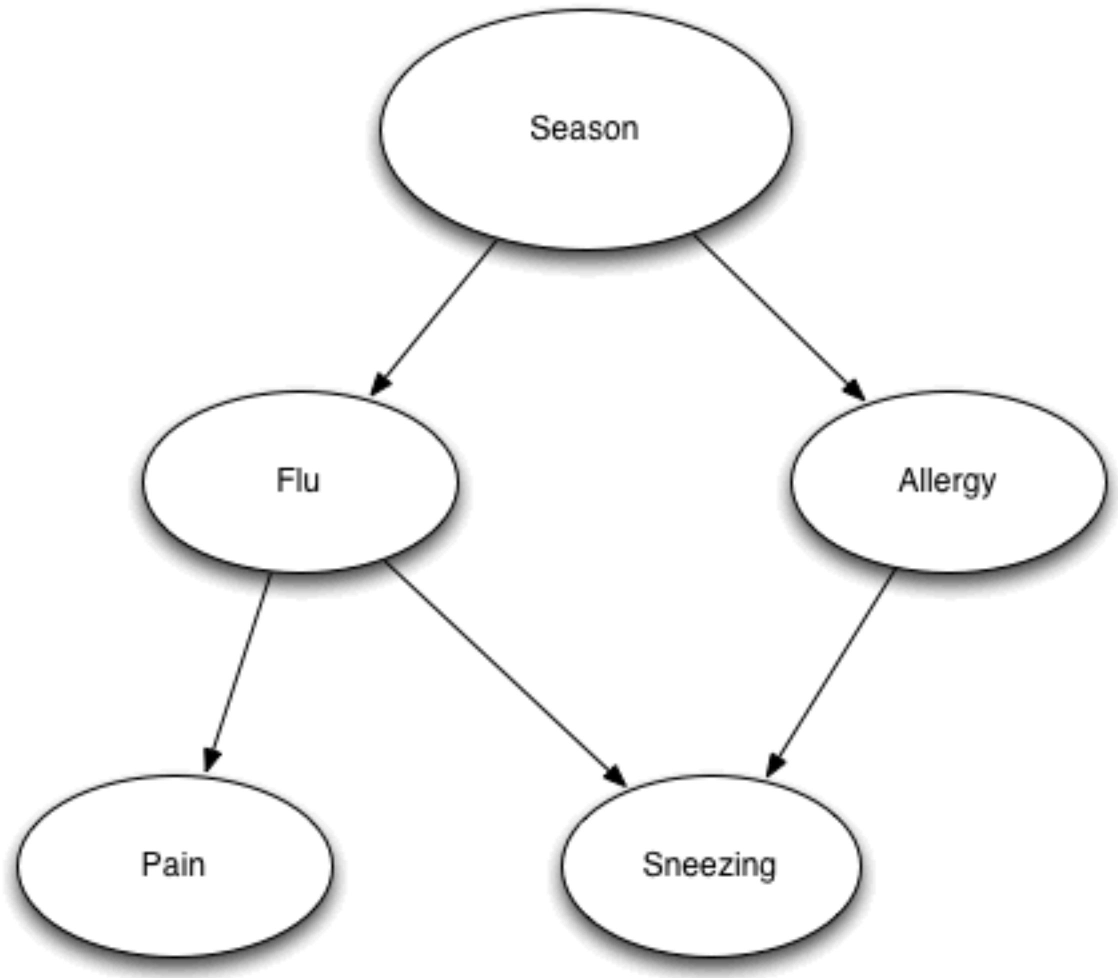


This version: traffic is affected by being attacked.

Medical example (from Koller and Friedman)

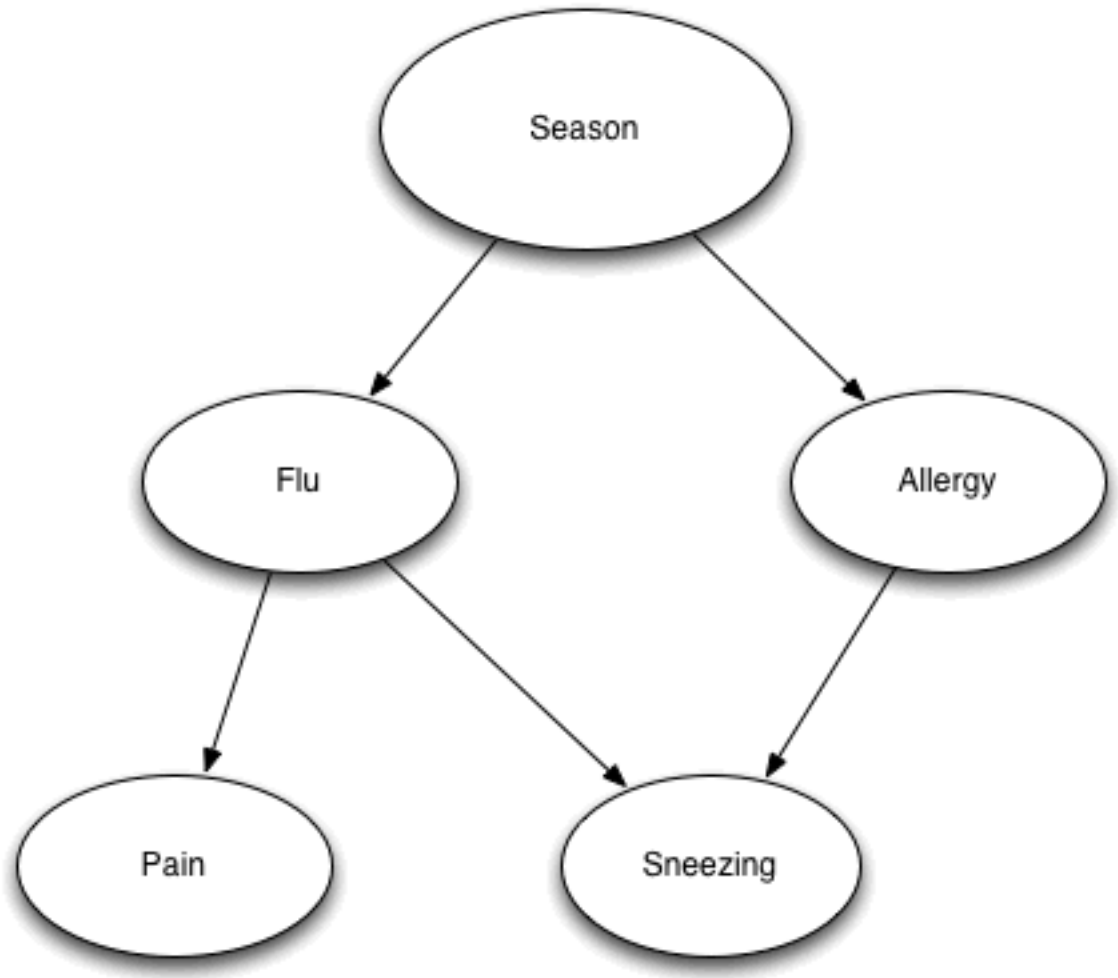


Medical example (from Koller and Friedman)



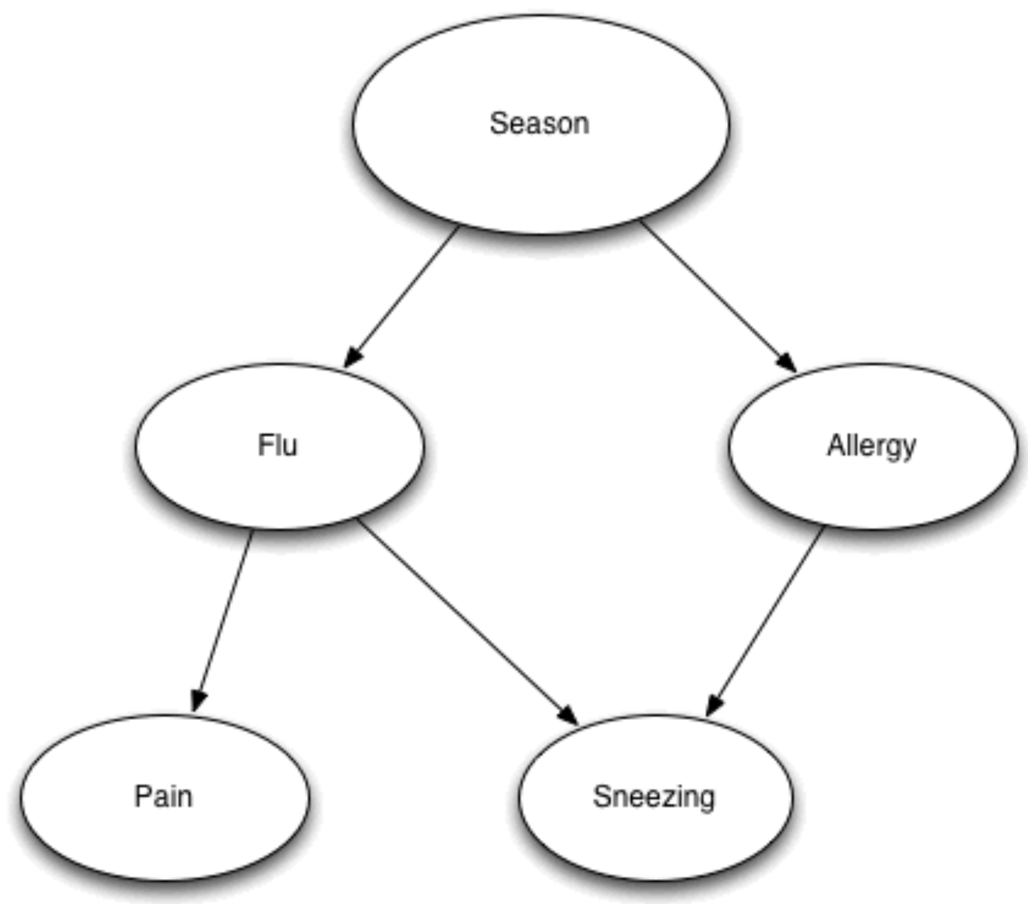
Flu and allergy are correlated through season.

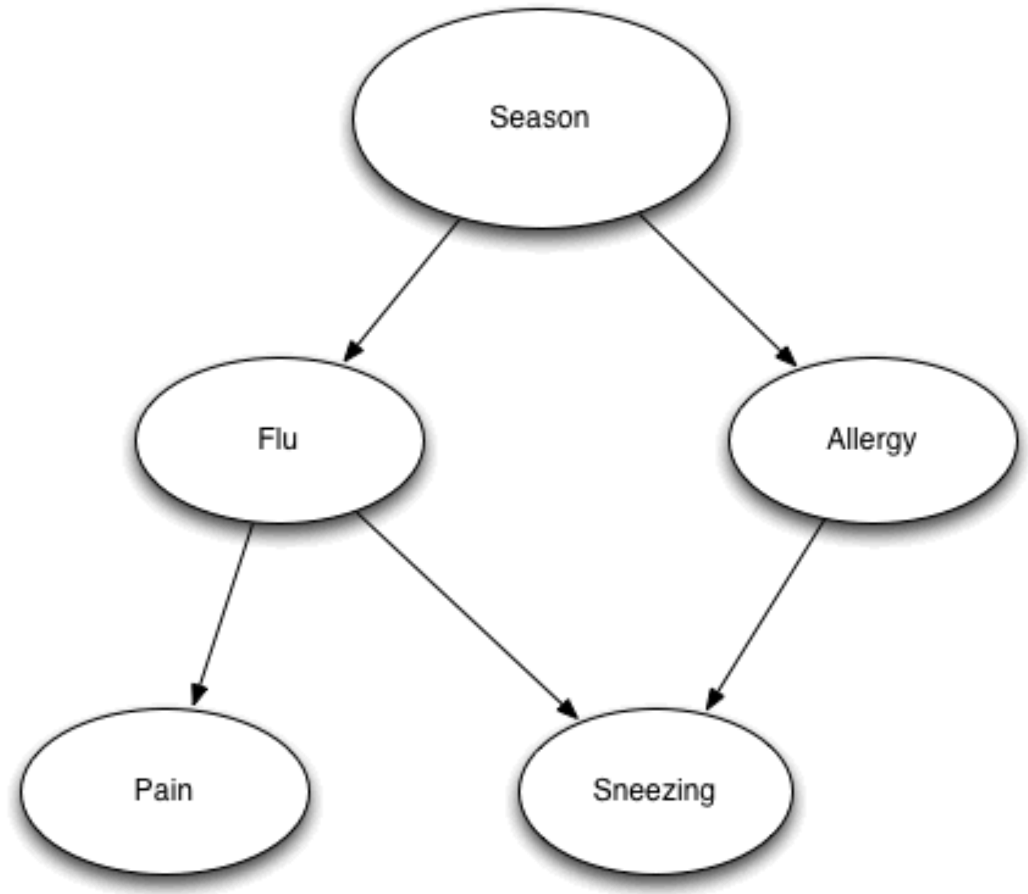
Medical example (from Koller and Friedman)



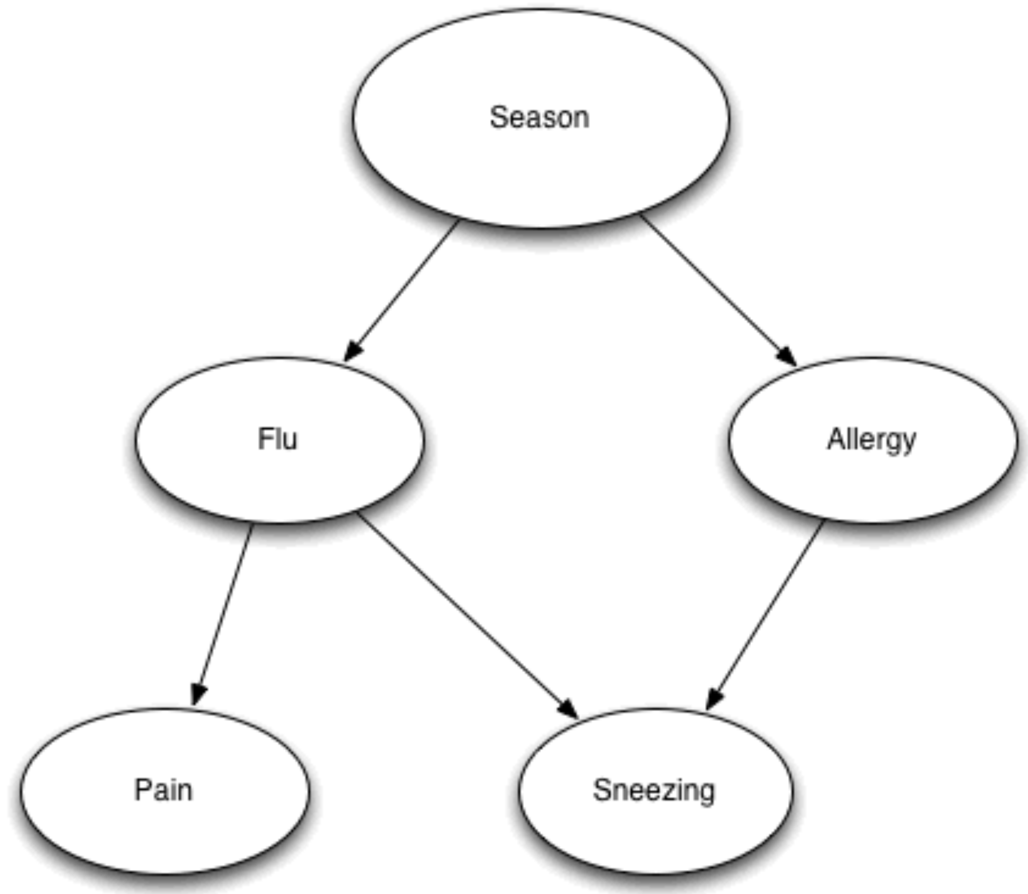
Flu and allergy are correlated through season.

Given the season, they are independent: $(A \perp F \mid S)$


Season
Flu
Allergy
Pain
Sneezing

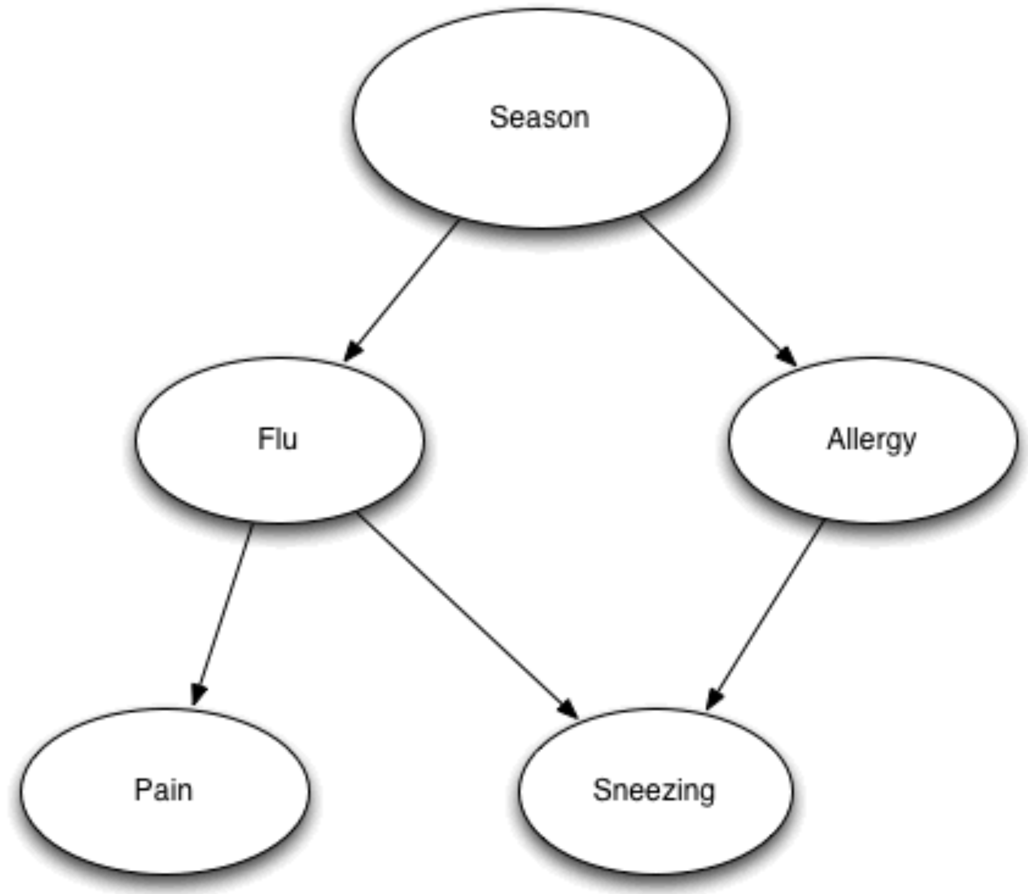


Independence relations:



Independence relations:

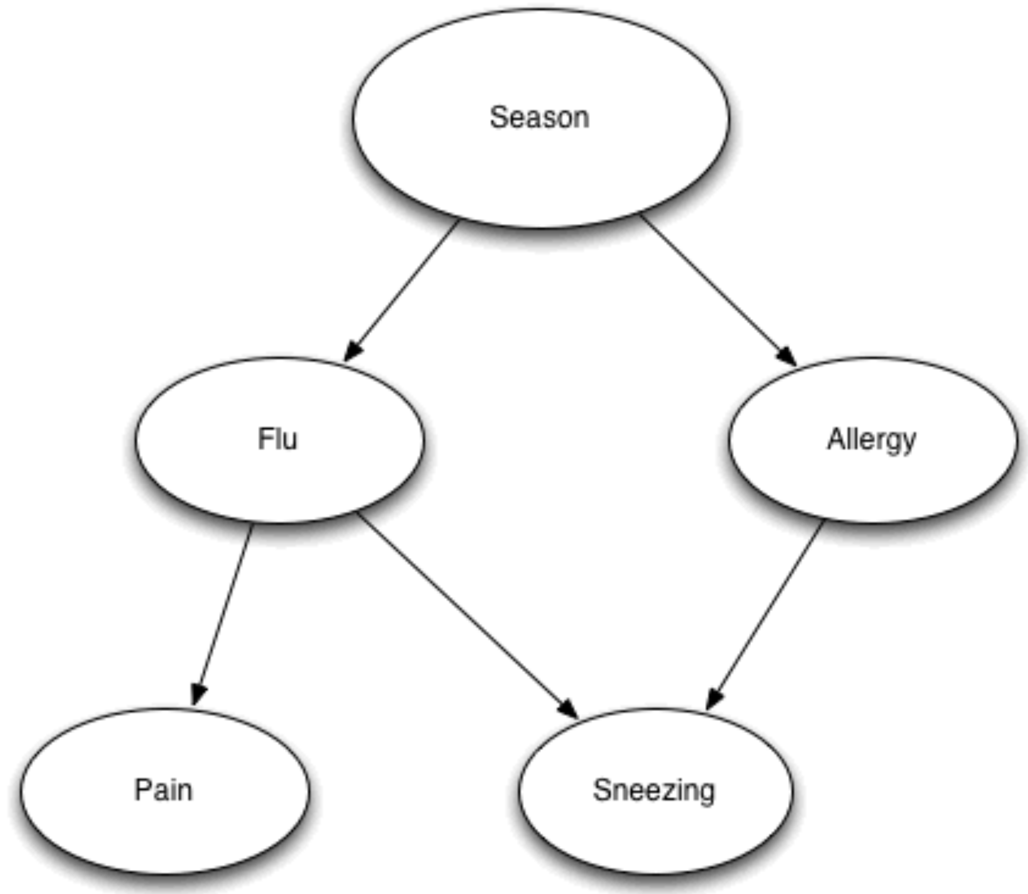
$$(F \perp A \mid S), (Sn \perp S \mid F, A), (P \perp A, Sn \mid F), (P \mid Sn \mid F)$$



Independence relations:

$$(F \perp A \mid S), (Sn \perp S \mid F, A), (P \perp A, Sn \mid F), (P \mid Sn \mid F)$$

$$P(Sn \mid F, A, S) = P(Sn \mid F, A)$$



Independence relations:

$$(F \perp A \mid S), (Sn \perp S \mid F, A), (P \perp A, Sn \mid F), (P \mid Sn \mid F)$$

$$P(Sn \mid F, A, S) = P(Sn \mid F, A)$$

Sn depends on S but it is *conditionally* independent given A and F.

Independence and Factorization

Independence and Factorization

Why do we care about conditional independence?

Independence and Factorization

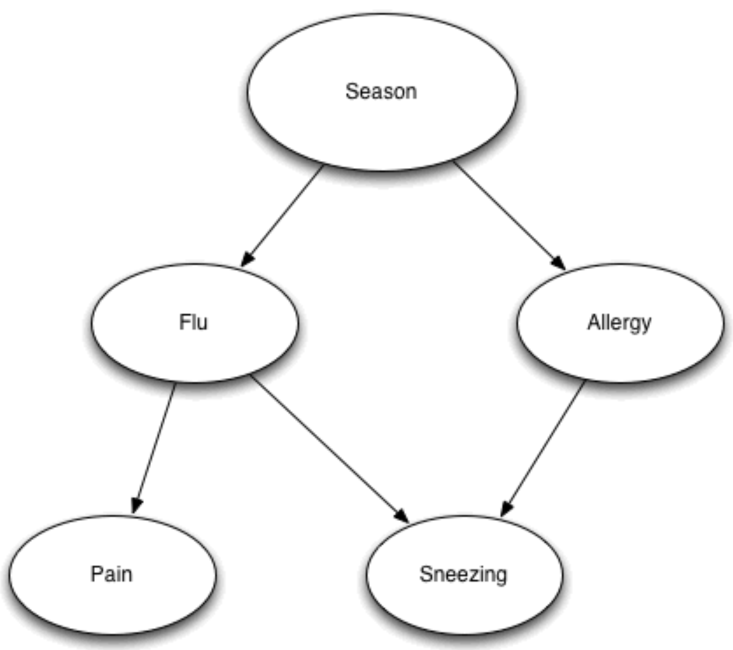
Why do we care about conditional independence?

Because we can *factorize* the joint distributions.

Independence and Factorization

Why do we care about conditional independence?

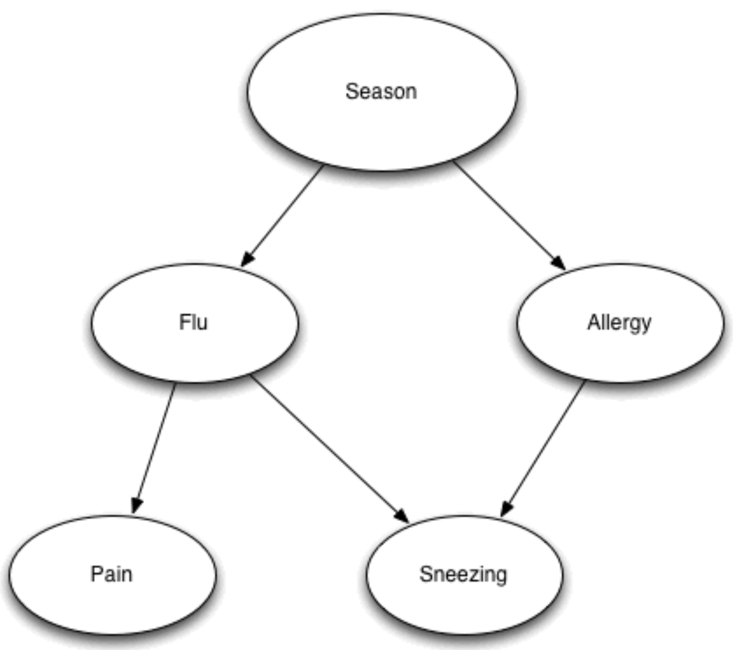
Because we can *factorize* the joint distributions.



Independence and Factorization

Why do we care about conditional independence?

Because we can *factorize* the joint distributions.



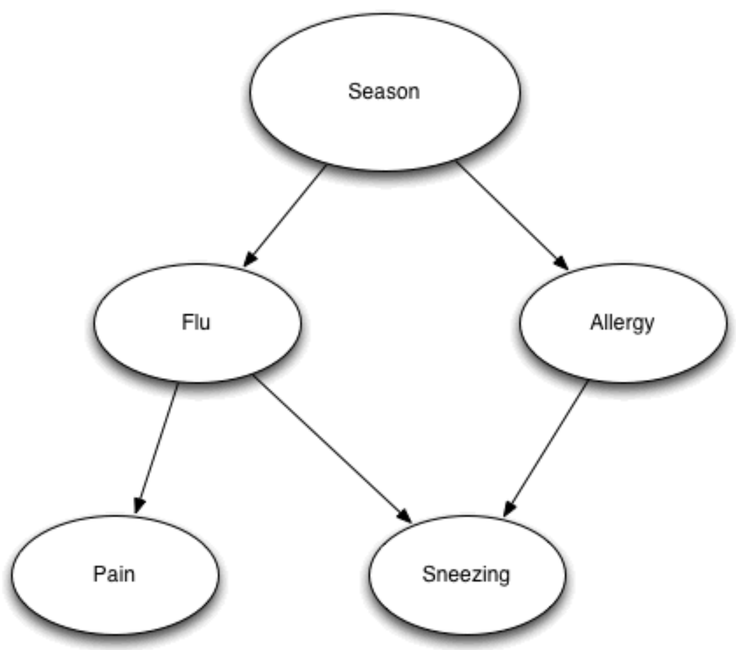
$$P(S, A, F, P, Sn) =$$

$$P(S)P(F|S)P(A|S)P(Sn|A, F)P(Pain|F)$$

Independence and Factorization

Why do we care about conditional independence?

Because we can *factorize* the joint distributions.



$$P(S, A, F, P, Sn) =$$

$$P(S)P(F|S)P(A|S)P(Sn|A, F)P(Pain|F)$$

A huge advantage for representing, computing and reasoning.

Where can we go from here?

Where can we go from here?

Continuous state spaces: robotics, telecommunication, control systems, sensor systems.

Where can we go from here?

Continuous state spaces: robotics, telecommunication, control systems, sensor systems.

Requires measure theory and analysis.

Where can we go from here?

Continuous state spaces: robotics, telecommunication, control systems, sensor systems.

Requires measure theory and analysis. Why?

Where can we go from here?

Continuous state spaces: robotics, telecommunication, control systems, sensor systems.

Requires measure theory and analysis. Why?

It is subtle to answer questions like:

Where can we go from here?

Continuous state spaces: robotics, telecommunication, control systems, sensor systems.

Requires measure theory and analysis. Why?

It is subtle to answer questions like:

I choose a real number between 0 and 1 uniformly, what is the probability of getting a rational number?

Where can we go from here?

Continuous state spaces: robotics, telecommunication, control systems, sensor systems.

Requires measure theory and analysis. Why?

It is subtle to answer questions like:

I choose a real number between 0 and 1 uniformly, what is the probability of getting a rational number? Answer: 0!

Where can we go from here?

Continuous state spaces: robotics, telecommunication, control systems, sensor systems.

Requires measure theory and analysis. Why?

It is subtle to answer questions like:

I choose a real number between 0 and 1 uniformly, what is the probability of getting a rational number? Answer: 0!

If every single point has probability 0 how can I have anything interesting?

Where can we go from here?

Continuous state spaces: robotics, telecommunication, control systems, sensor systems.

Requires measure theory and analysis. Why?

It is subtle to answer questions like:

I choose a real number between 0 and 1 uniformly, what is the probability of getting a rational number? Answer: 0!

If every single point has probability 0 how can I have anything interesting?

Is it possible to assign a probability to every set?

Where can we go from here?

Continuous state spaces: robotics, telecommunication, control systems, sensor systems.

Requires measure theory and analysis. Why?

It is subtle to answer questions like:

I choose a real number between 0 and 1 uniformly, what is the probability of getting a rational number? Answer: 0!

If every single point has probability 0 how can I have anything interesting?

Is it possible to assign a probability to every set? Answer: No!

Where can we go from here?

Continuous state spaces: robotics, telecommunication, control systems, sensor systems.

Requires measure theory and analysis. Why?

It is subtle to answer questions like:

I choose a real number between 0 and 1 uniformly, what is the probability of getting a rational number? Answer: 0!

If every single point has probability 0 how can I have anything interesting?

Is it possible to assign a probability to every set? Answer: No!

There is a rich and fascinating theory of programming and reasoning about probabilistic systems.

Logic and Probability are your weapons.

Go forth and conquer the software world!

Thank you!