RUTGERS
THE STATE UNIVERSITY
OF NEW JERSEY

# Approximate Inference in Bayesian Network

Edgar Granados

# Bayesian Networks

## Complexity

$$
\begin{aligned}
\textit{Sentence} &\rightarrow \textit{AtomicSentence} \mid \textit{ComplexSentence} \\
\textit{AtomicSentence} &\rightarrow \textit{True} \mid \textit{False} \mid P \mid Q \mid R \mid \ldots \\
\textit{ComplexSentence} &\rightarrow (\ \textit{Sentence}\ ) \\
&\quad \mid \neg\ \textit{Sentence} \\
&\quad \mid \textit{Sentence} \land \textit{Sentence} \\
&\quad \mid \textit{Sentence} \lor \textit{Sentence} \\
&\quad \mid \textit{Sentence} \Rightarrow \textit{Sentence} \\
&\quad \mid \textit{Sentence} \Leftrightarrow \textit{Sentence}
\end{aligned}
$$

$D = A \land B$

$C = A \lor B$

$P(C=c \mid a, b) = 1$

$P(C=c \mid \neg a, b) = 1$

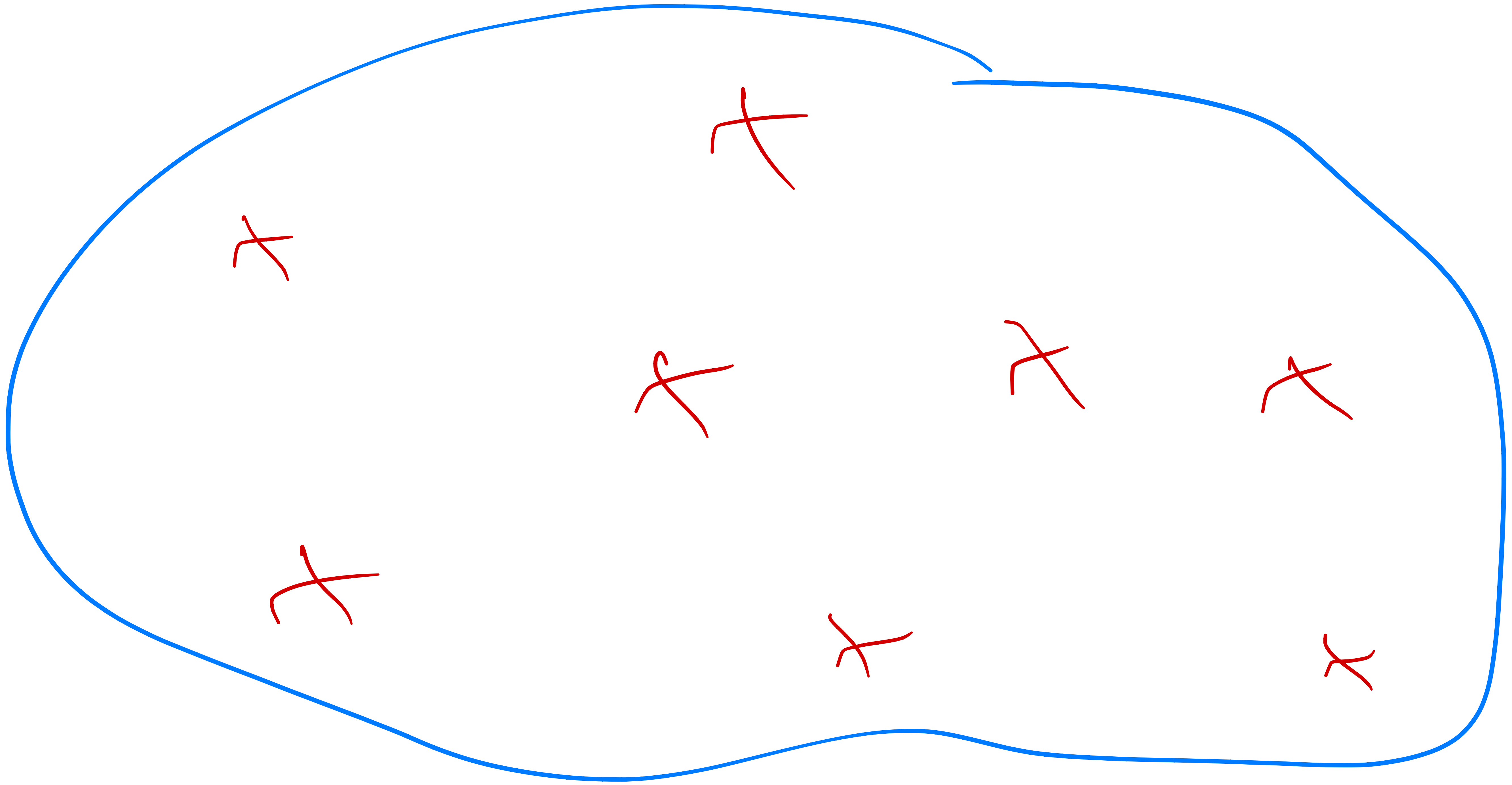
$P(C=c \mid a, \neg b) = 1$

$P(C=c \mid \neg a, \neg b) = 0$

Inference on P.L. contains 3SAT

# Sampling

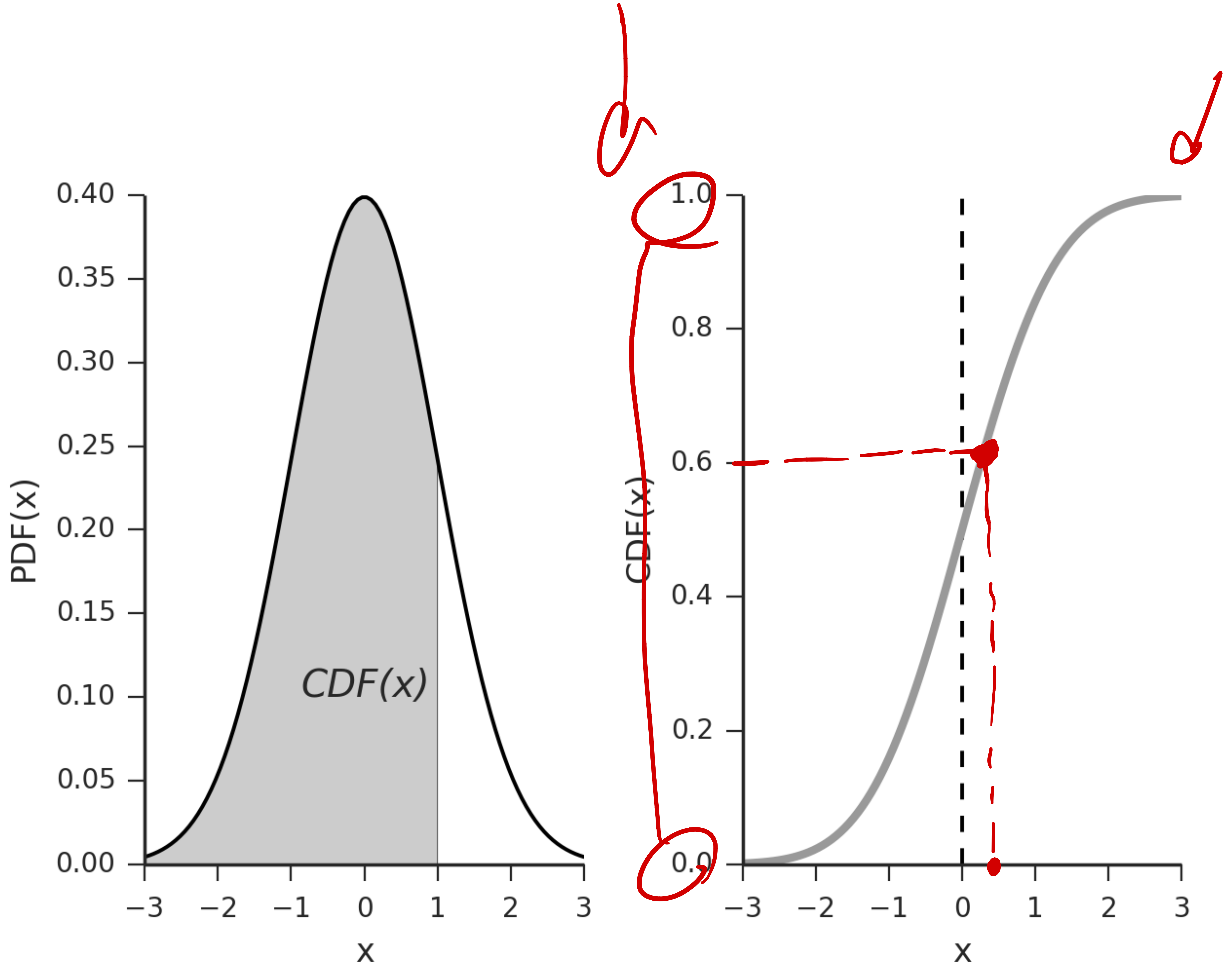
- Knowing CPT
  - Sample
  - Count
    - Frequency



# Sampling Continuous CDF

- Obtain one sample from 1D CDF
    - $\mathcal{N}(\mu, \sigma)$

1. [0, 1]
   0.6 = y
2. locate on CDF "y"
3. x = 0.3
   ↳ desired sample



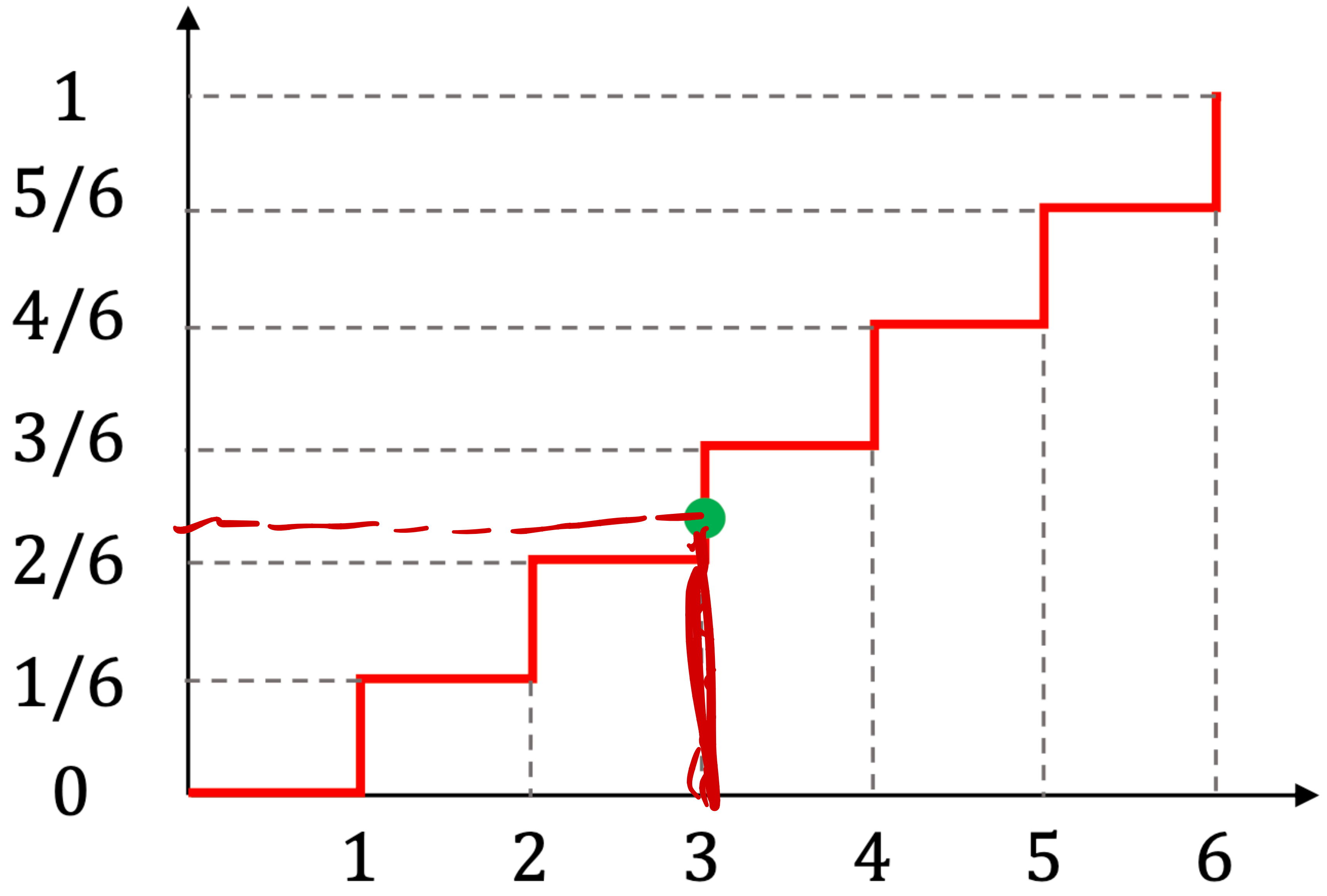
# Sampling Discrete CDF

- Obtain one sample from 1D CDF
  - $\cancel{\mathcal{N}(\mu, \sigma)}$

1 dice

↳ .4 = y

→ x = 3



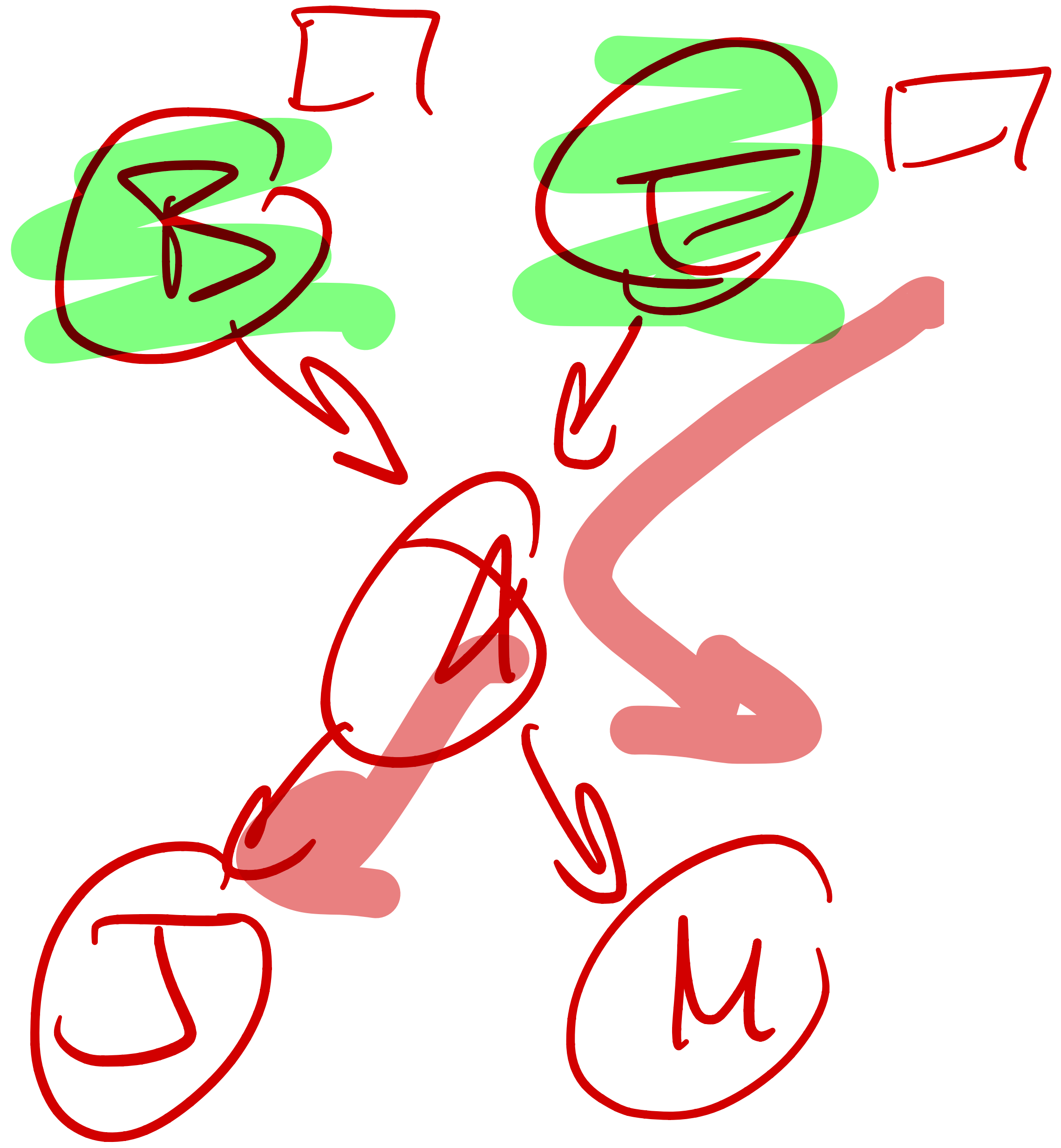
# Direct Sampling of Joint Probability

- Sample atomic events

$(x_1, \dots, x_n)$

Procedure

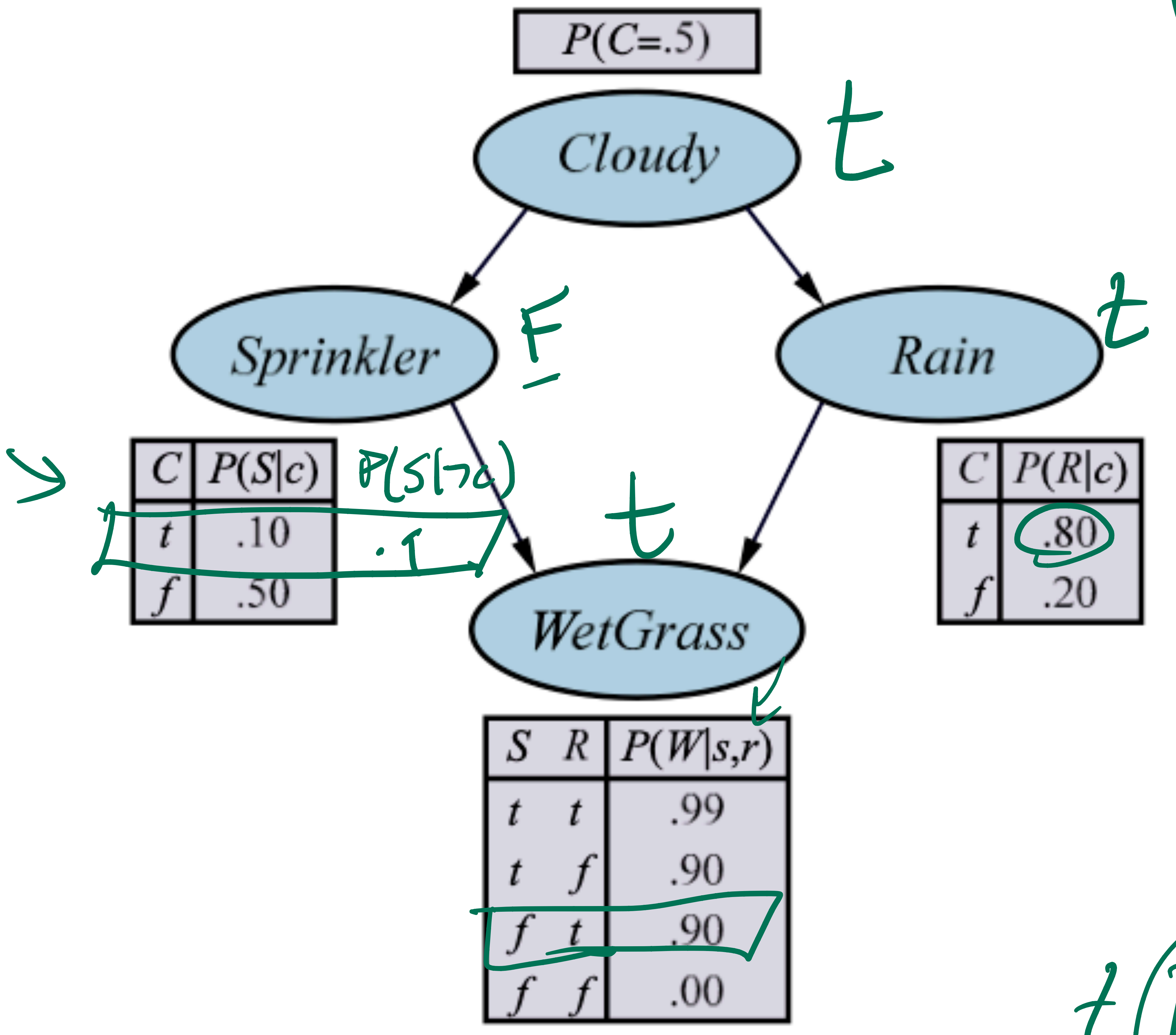
1. Start from RV with no parents
2. Sample following topological order
3. Collect *n* samples
4. Compute the frequency




$$P(x_1, \dots, x_n) = \lim_{N \to \infty} \frac{N(x_1, \dots, x_n)}{N}$$

# Direct Sampling of Joint Probability

**Example**




$$\boxed{e_i = [t, F, t, t]}$$

$$P(t,F,t,t) = \prod_{i=1}^{4} P(x_i \mid Parents(x_i))$$

$$= .5 \cdot .9 \cdot .8 \cdot .9 = .324$$

$$\boxed{32.4\%}$$

t, F, F, t

# Rejection Sampling

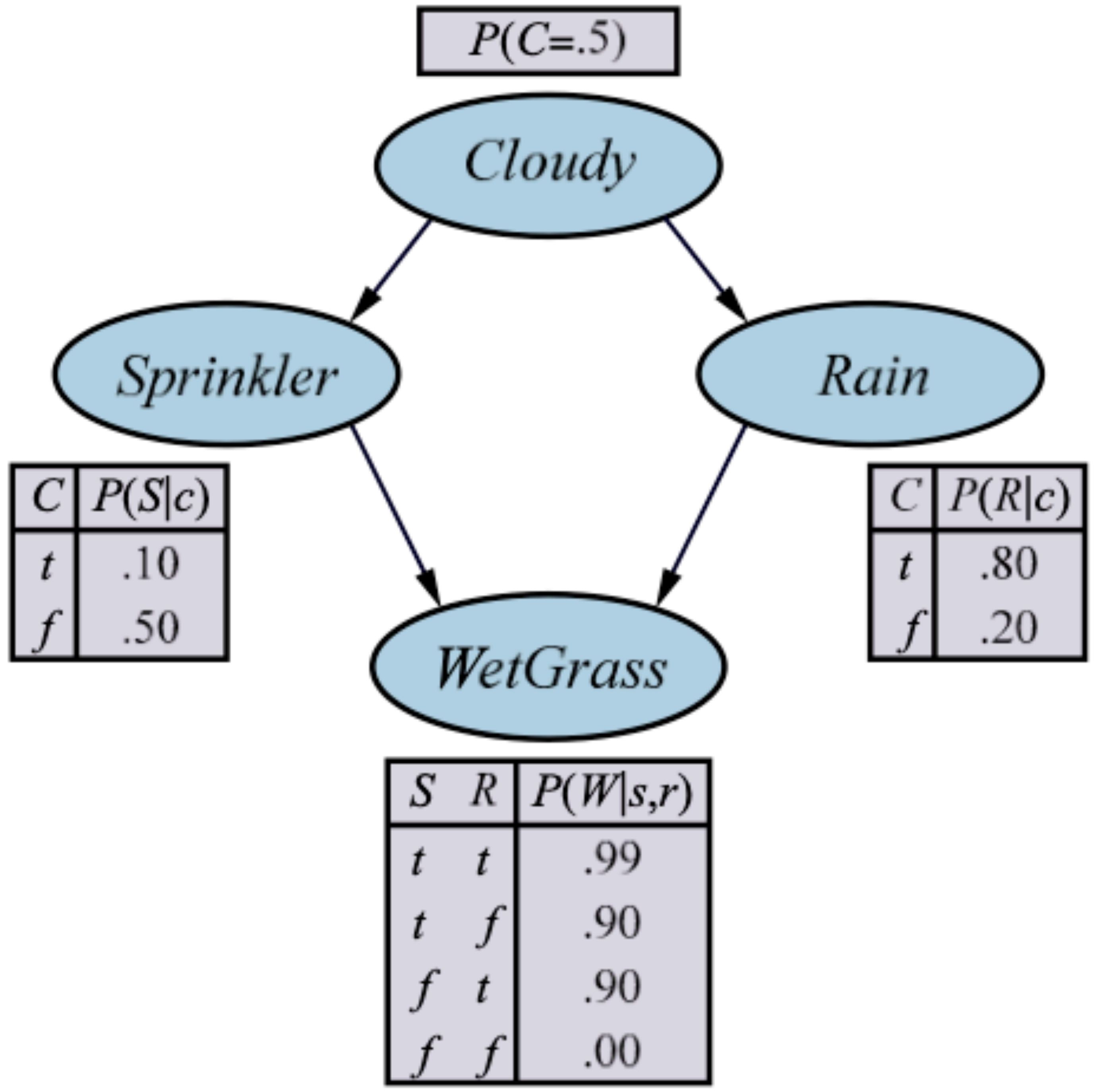
- What if some variables have fixed values?
  - $P(X \mid e)$

P( Rain | Sprinkler=true )

Procedure

1. Generate $n$ samples
2. **Reject** samples that do not match $e$
3. Estimate $\hat{P}(x \mid e)$ by counting $X = x$

$$\hat{P}(x|e) = \frac{number\ of\ samples\ with\ X = x\ and\ E = e}{number\ of\ samples\ with\ E = e}$$



# Rejection Sampling

## Example

- Estimate $P(Rain \mid Sprinkler = true)$
- 100 samples

73 Sp=false    73%

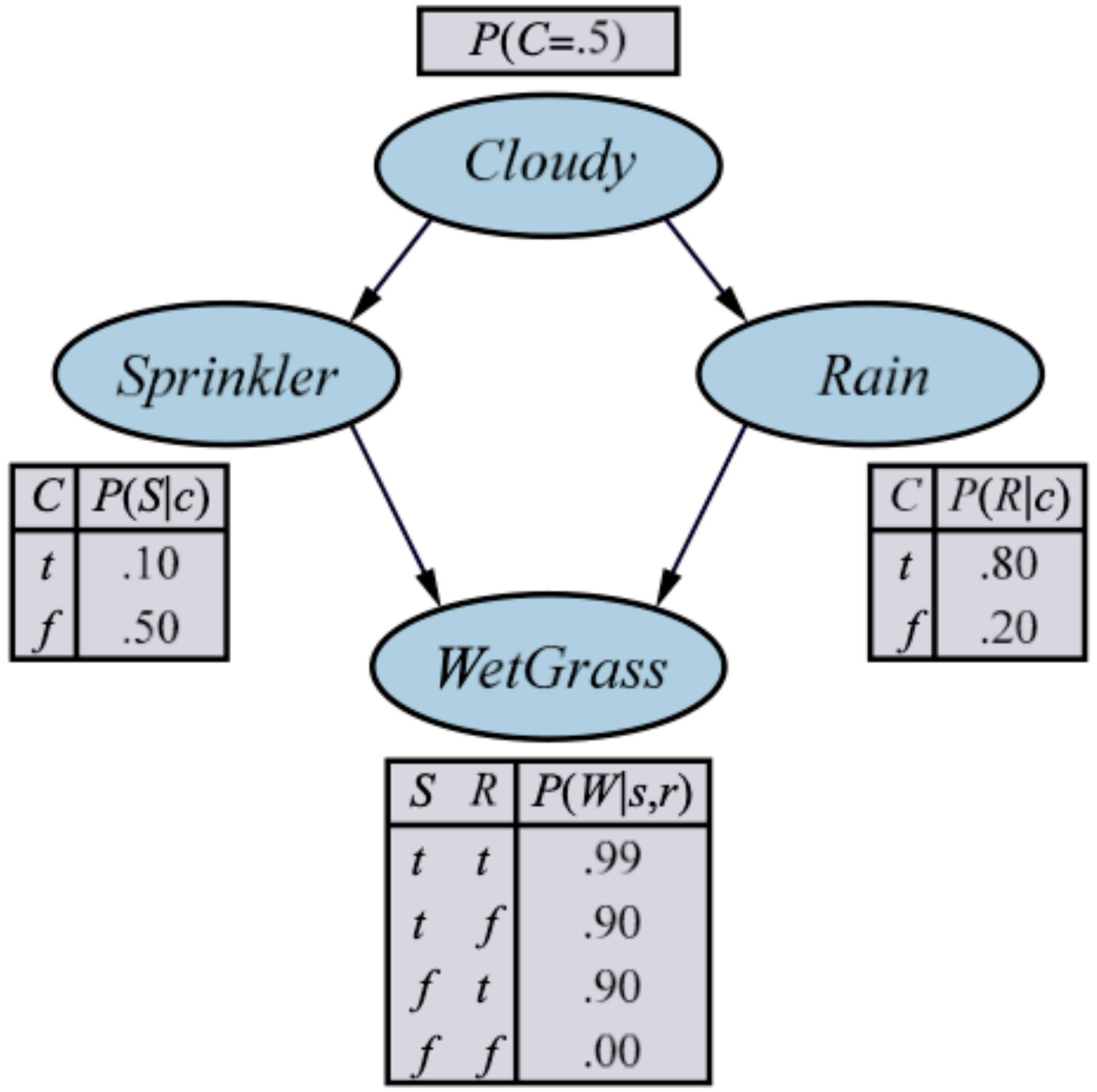
27 Sp=true

8 Rain=true

19 Rain=false

$P(R \mid Sp=true) \approx \text{Normalize}(\langle 8, 19 \rangle)$

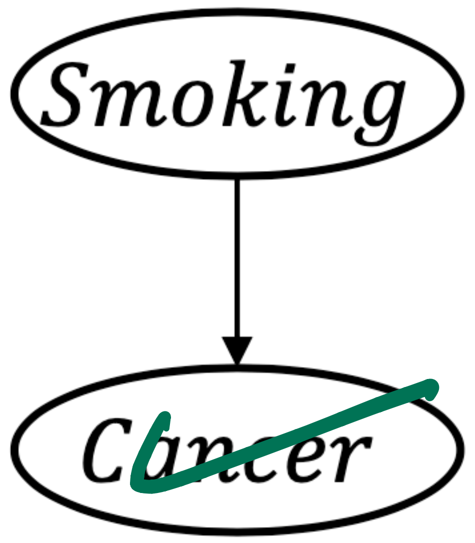
$= \langle .296, .704 \rangle$



# Forcing Evidence Variable Value?

- Evidence: $\boxed{C=c}$
  - P(s|c)?
  - $\hat{P}(x|e) = \dfrac{number\ of\ samples\ with\ X = x\ and\ E = e}{number\ of\ samples\ with\ E = e}$ $= 0.3$

$P(s) = 0.3$




$\boxed{P(s) = 0.3}$

| S | P(C\|s) |
|---|---|
| t | 0.2 |
| f | 0.01 |

Bayes

$$P(s|c) = \frac{P(c|s)P(s)}{P(c|s)P(s) + P(c|\neg s)P(\neg s)} = \frac{.2 \cdot .3}{.2 \cdot .3 + .01 \cdot .7} = 0.9$$

# Likelihood Weighting

- $P(Rain \mid c, w)$

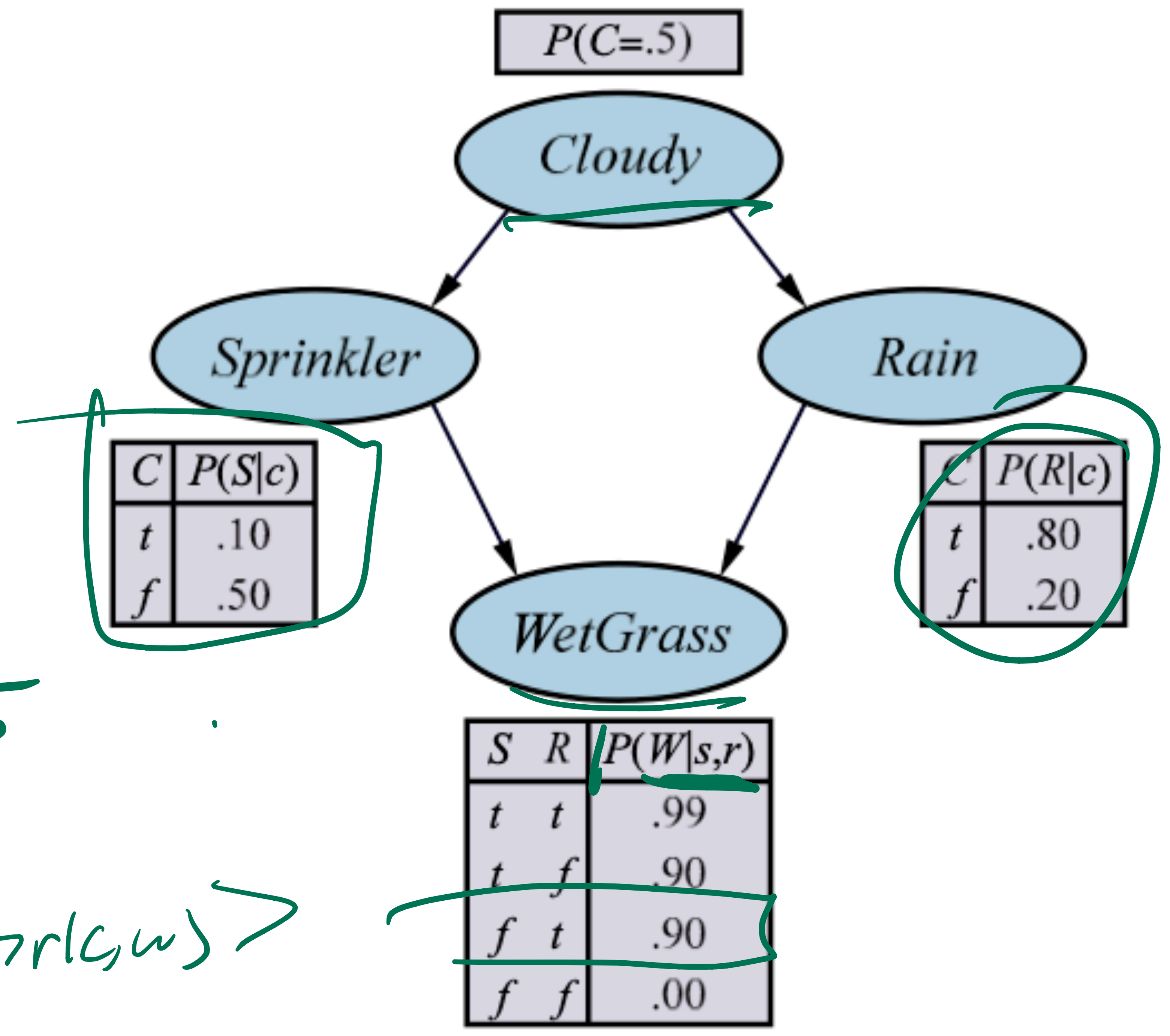
Weight $w = 1$

- **Cloudy** – true $\quad w \leftarrow w \cdot P(c) = 1 \cdot .5 = .5$
- **Sprinkler** – $\neg s$
- **Rain** – $r$
- **WetGrass** – $w \leftarrow w \cdot P(w \mid \neg s, r) = .5 \cdot .9 = .45$

samples (example)

Maintain vector of weights $\langle w_{P(r \mid c,w)}, w_{P(\neg r \mid c,w)} \rangle$

↳ $\langle .45, 0 \rangle$

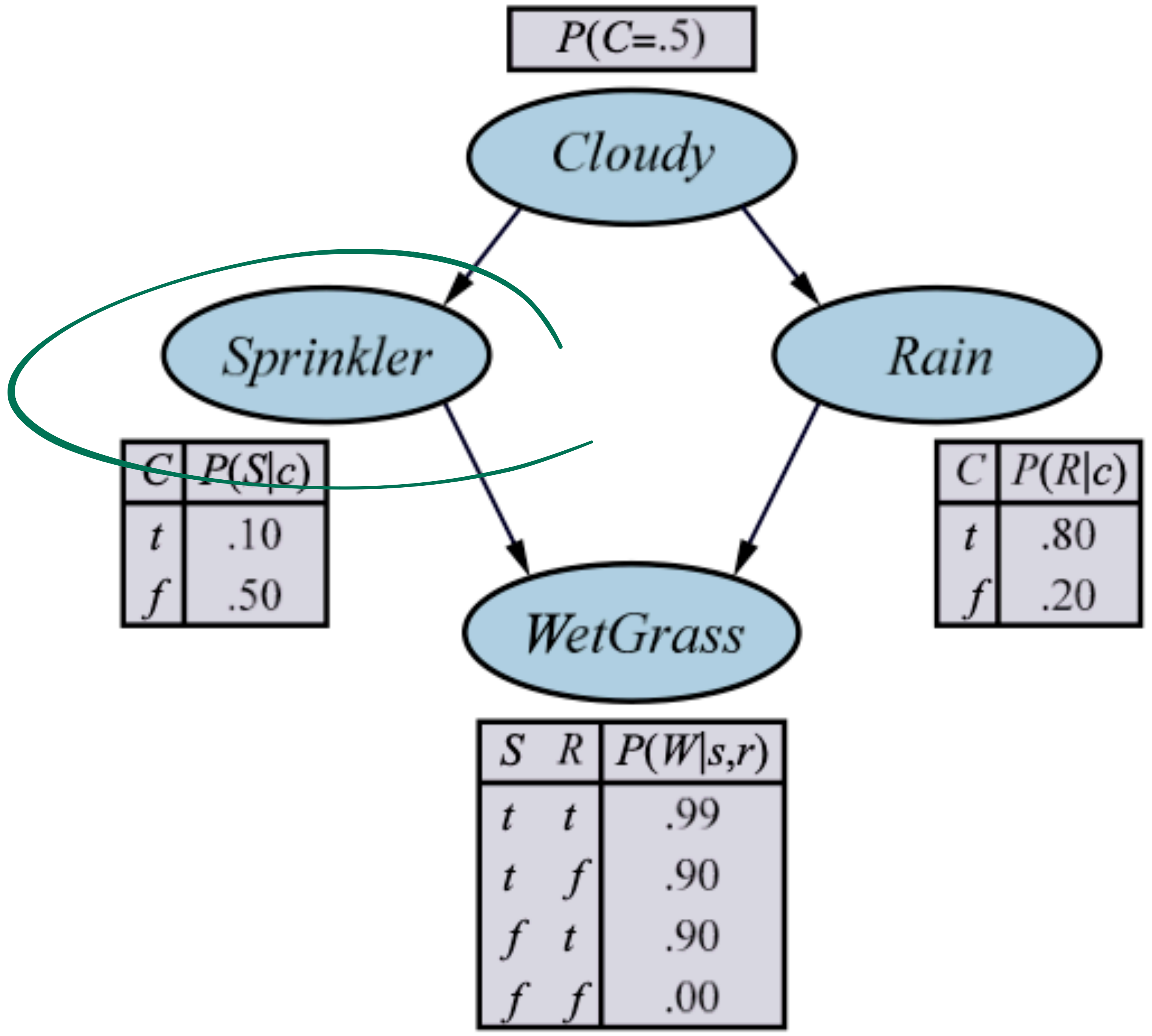


# Likelihood Weighting

- $P(Rain \mid c, w)$

$e = c, w$
$x = R$
$y = S$

- **Cloudy**
- **Sprinkler**
- **Rain**
- **WetGrass**



# Likelihood Weighting

- Let $[z, e]$ be a possible assignment of values to all RVs, in which $e$ corresponds to the evidence variables and $z = \boxed{[x, y]}$ corresponds to the sampled variables. $y$ contains values for the sampled variables that are not in what we want
  - i.e., $P(X \mid e)$
- For all variables that are sampled, the probability of the samples taking $z$:

$$S(x, y, e) = S(z, e) = \prod_{i=1}^{\ell} P(z_i | parents(Z_i))$$

- Weight that goes in the procedure

$$w(x, y, e) = w(z, e) = \prod_{i=1}^{m} P(e_i | parents(E_i))$$

# Likelihood Weighting

- For all variables that are sampled, the probability of the samples taking $z$:

$$S(x, y, e) = S(z, e) = \prod_{i=1}^{\ell} P(z_i | parents(Z_i))$$

- Weight that goes in the procedure

$$w(x, y, e) = w(z, e) = \prod_{i=1}^{m} P(e_i | parents(E_i))$$

- $m + l = n$

$$P(x, y, e) = P(z, e) = \prod_{i=1}^{m} \prod_{i=1}^{\ell} P(z_i | parents(Z_i)) P(e_i | parents(E_i))$$

# Likelihood Weighting

$$P(x, y, e) = P(z, e) = \prod_{i=1}^{m} \prod_{i=1}^{\ell} P(z_i | parents(Z_i))\, P(e_i | parents(E_i))$$

- Likelihood weighting procedure

as N becomes large

$$\underline{\hat{P}(x|e)} = \frac{\sum_y N(x,y,e) w(x,y,e)}{\underline{\sum_x \sum_y} \underline{N(x,y,e)} \underline{w(x,y,e)}} \approx \frac{\sum_y S(x,y,e) w(x,y,e)}{\sum_x \sum_y S(x,y,e) w(x,y,e)} = \frac{\sum_y P(x,y,e)}{\sum_x \sum_y P(x,y,e)}$$

$$= \frac{P(x,e)}{P(e)} = P(x|e)$$

# Likelihood Weighting

$$P(x,y,e) = P(z,e) = \prod_{i=1}^{m} \prod_{i=1}^{\ell} P(z_i | parents(Z_i))\, P(e_i | parents(E_i))$$

- Likelihood weighting procedure

$$\hat{P}(x|e) = \frac{\sum_y N(x,y,e) w(x,y,e)}{\sum_x \sum_y N(x,y,e) w(x,y,e)} \approx \frac{\sum_y S(x,y,e) w(x,y,e)}{\sum_x \sum_y S(x,y,e) w(x,y,e)} = \frac{\sum_y P(x,y,e)}{\sum_x \sum_y P(x,y,e)} = \frac{P(x,e)}{P(e)} = P(x|e)$$

# Markov Chain Monte Carlo

- Generate each sample by making random changes to the preceding sample
- **State:** assignment of values to all variables

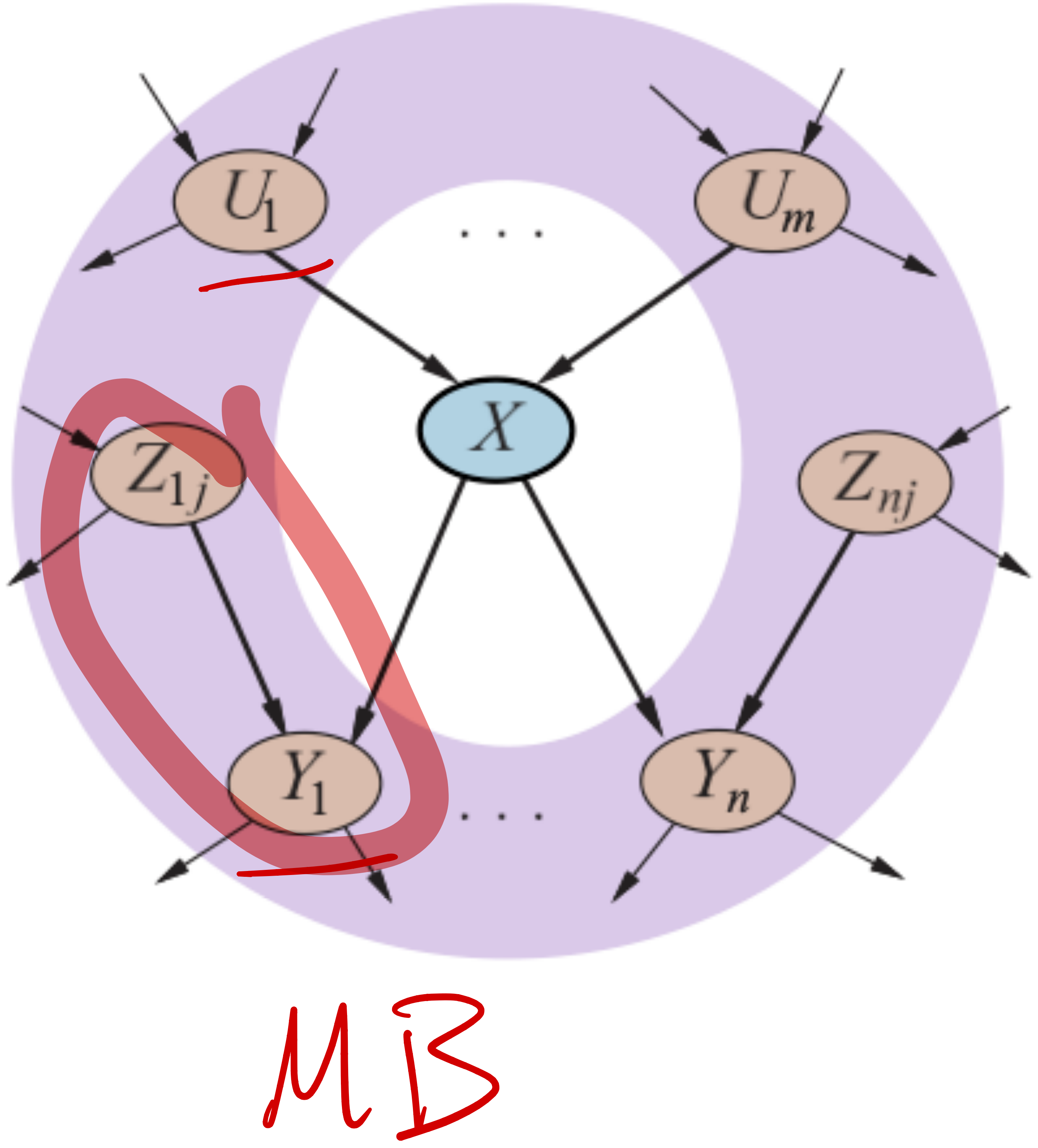
- Frequency of each value of the inquiry variable $X$ converges to its true distribution
  - $P(X \mid e)$

# Markov Chain Monte Carlo

$(X_1, \ldots, X_n)$

## Procedure

1. For each $X_i$
   1. Sample a new value $P(X_i \mid \boxed{MB(X_i)})$
2. Add the current state to **Samples**
3. Repeat



1. For each $X_i$
    1. Sample a new value $P(X_i \mid MB(X_i))$
2. Add the current state to **Samples**
3. Repeat

# Markov Chain Monte Carlo

## Example

- Compute $P(Rain \mid s, w)$
    - $S = s, W = w$ → Fixing

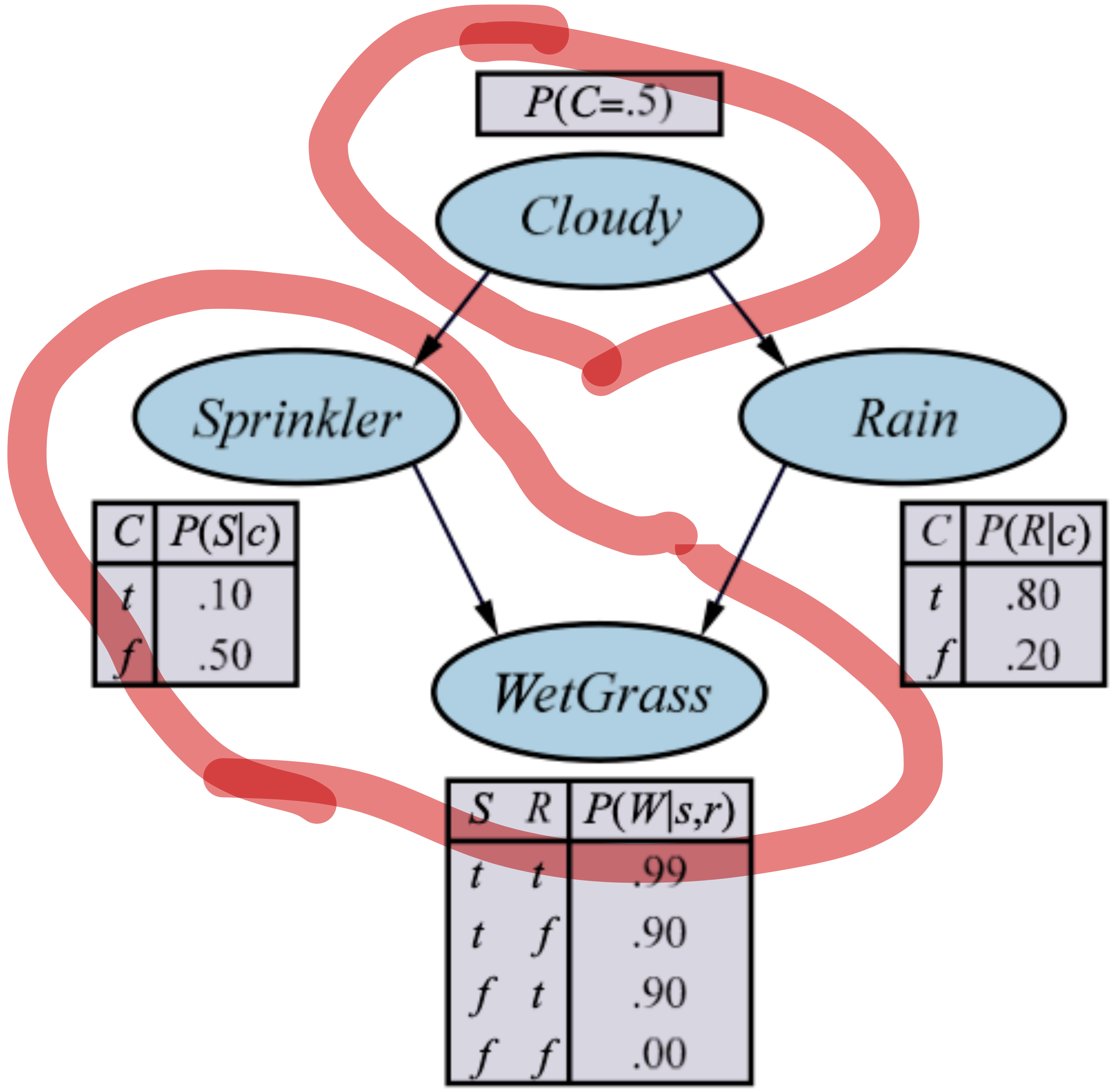
$[c, s, \neg r, w]$

Sample $P(C \mid s, \neg r) \rightarrow Cloudy = \neg c$

Sample $P(R \mid \neg c, s, w) \rightarrow Rain = r$

$[\neg c, s, r, w]$

After 100 iterations



# Markov Chain Monte Carlo

1. For each $X_i$
   1. Sample a new value $P(X_i \mid MB(X_i))$
2. Add the current state to **Samples**
3. Repeat

## Example

- Compute $P(Rain \mid s, w)$
  - $S = s, W = w$
- (...)
- After 100 iteration:
  - $r$ - 20 times
  - $\neg r$ - 80 times

$\Rightarrow P(Rain \mid s, w) = \text{Normalize}(\langle 20, 80 \rangle)$
$= \langle .2, .8 \rangle$