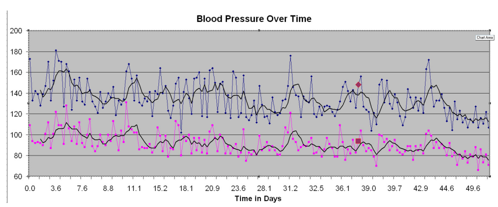
# PROBABILISTIC REASONING OVER TIME

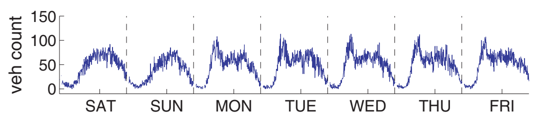
## Today

- Dynamic Bayesian Networks
- Hidden Markov Models
- Exact Inference in HMMs
  - Filtering
  - Smoothing
- Approximate Inference in HMMs

# Modeling uncertainty over time

- In a *dynamic* process, the value of a random variable changes over time



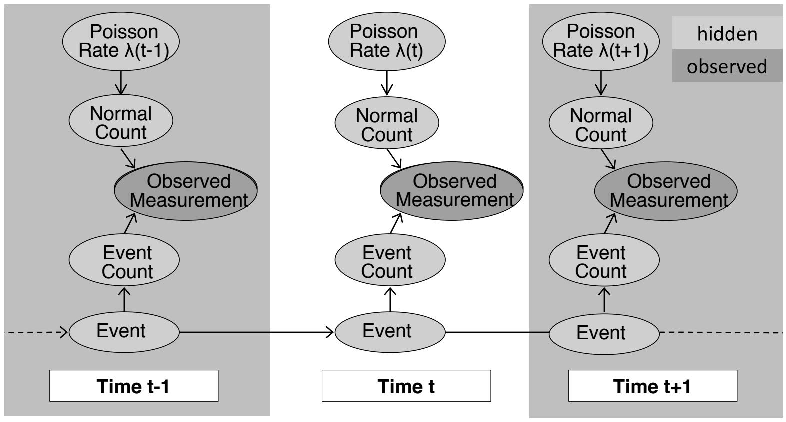




# How to model a dynamic process?

- Dynamic Bayesian network
  - A Bayesian network composed of a series of time slices

- Each time slice contains a set of random variables
  - Evidence variables whose value we observe ($E_t$)
  - State variables whose value we don't observe ($X_t$)

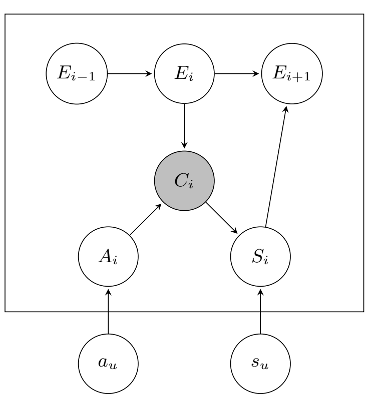
# Examples of DBN




J. Hutchins et al. Probabilistic analysis of a large-scale urban traffic sensor data set.

# Examples of DBN

**Track: Data Mining / Session: Click Models**




- $E_i$: did the user *examine* the url?
- $A_i$: was the user *attracted* by the url?
- $S_i$: was the user *satisfied* by the landing page?

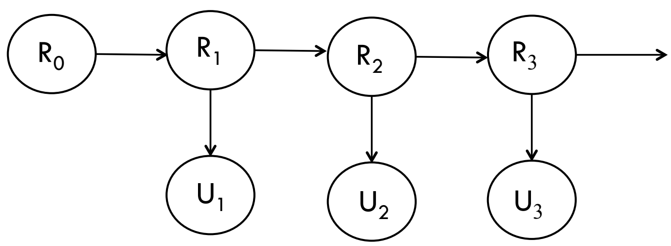
**Figure 1: The DBN used for clicks modeling. $C_i$ is the the only observed variable.**

## Transmission Model

## Sensor (Emission) Model

# Hidden Markov Model

- A Hidden Markov Model is the simplest type of dynamic Bayesian network
- The state is a single discrete random variable

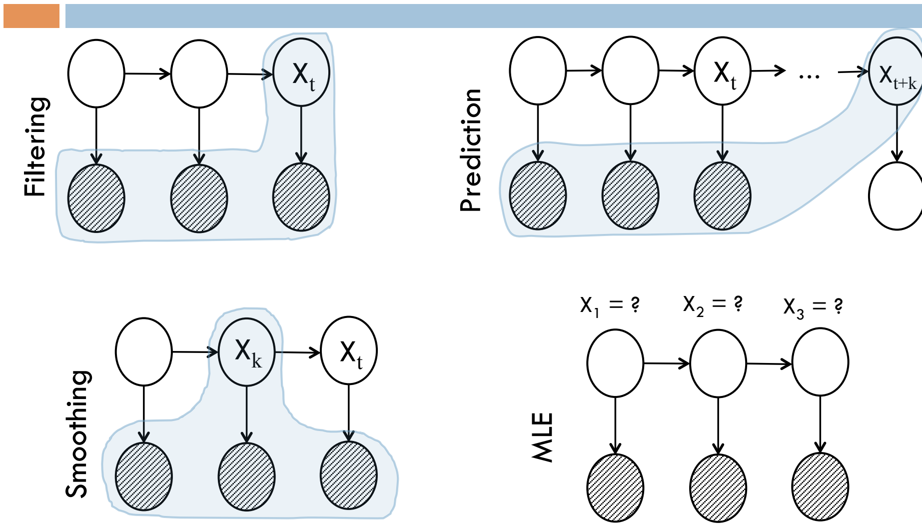



# Exact Inference in HMMs

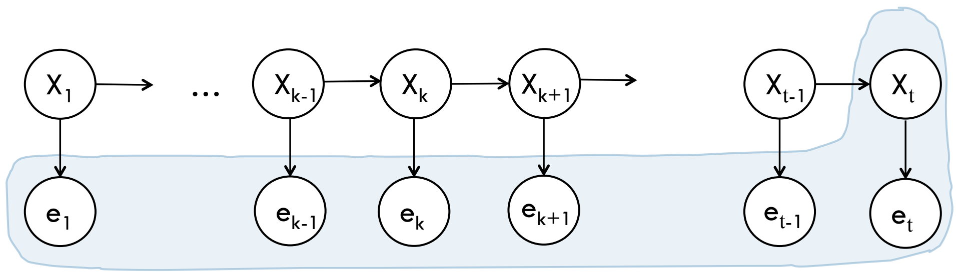
# Types of Exact Inference

- Filtering
- Prediction
- Smoothing
- Most Likely Explanation

# Types of Exact Inference



## Filtering




$$P(X_t \mid e_{1:t})$$

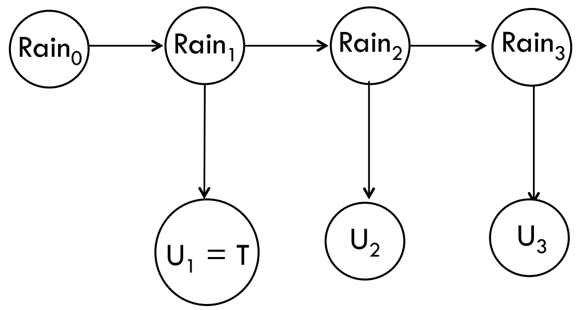
## Deriving Forward Algorithm

# Filtering Practice

$p(R_0)$ = <0.5, 0.5>

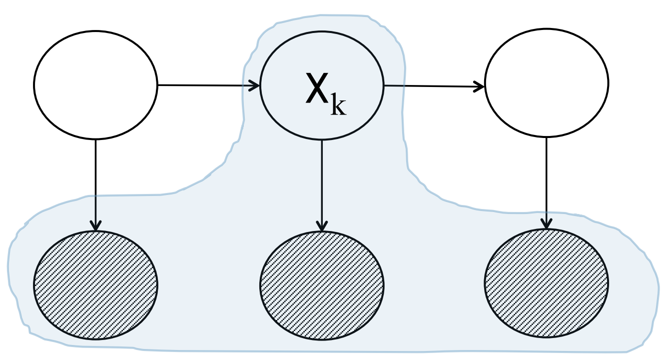
| $R_{t-1}$ | $p(R_t \mid R_{t-1})$ |
|---|---|
| T | 0.7 |
| F | 0.3 |

| $R_t$ | $p(U_t \mid R_t)$ |
|---|---|
| T | 0.9 |
| F | 0.2 |




$$p(X_t|e_{1:t}) \propto p(e_t|X_t) \sum_{X_{t-1}} p(X_t|X_{t-1})\, p(X_{t-1}|e_{1:t-1})$$

# Smoothing




$P( X_k \mid e_{1:t} )$

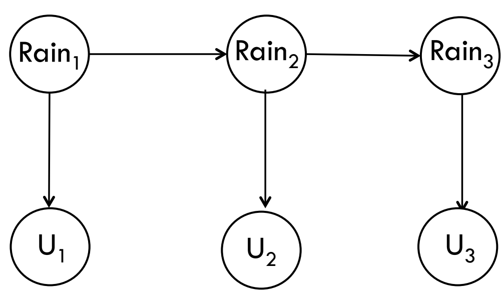
## Deriving The Forward Backward Algorithm

## Smoothing Practice

$p(R_0) = <0.5, 0.5>$

| $R_{t-1}$ | $p(R_t \mid R_{t-1})$ |
|---|---|
| T | 0.7 |
| F | 0.3 |

| $R_t$ | $p(U_t \mid R_t)$ |
|---|---|
| T | 0.9 |
| F | 0.2 |




| $P(r_1 \mid u_1)$ | $P(r_2 \mid u_1, u_2)$ | $P(r_1 \mid u_1, u_2)$ |
|---|---|---|
| 0.818 | 0.883 | ? |

# Most Likely Explanation

- Find the state sequence that makes the observed evidence sequence most likely

$$\underset{X_{1:t}}{\text{argmax}}\ P(X_{1:t} | e_{1:t})$$

- Recursive formulation:
  - The most likely state sequence for $X_{1:t}$ is the most likely state sequence for $X_{1:t-1}$ followed by the transition to $X_t$
  - Equivalent to Filtering algorithm except summation replaced with max
  - Called the Viterbi Algorithm

# Approximate Inference for HMMs

## Approximate Inference in Dynamic BNs

- Approximate inference in BNs
  - Direct sampling, rejection sampling, likelihood weighting
  - Gibbs sampling

- Likelihood weighting applied to a dynamic Bayesian network (with some modifications) is known as a Particle filter

## Particle Filtering

- Likelihood weighting fixes the evidence variables and samples only the non-evidence variables
- Introduces a weight to correct for the fact that we're sampling from the prior distribution instead of the posterior distribution

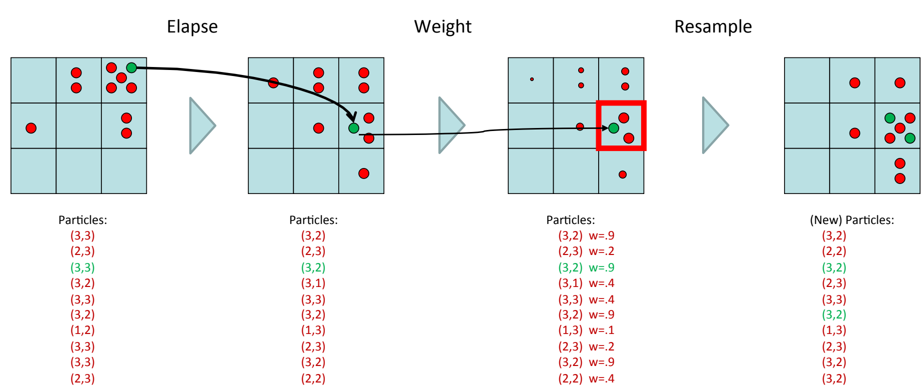
$$\text{weight} = p(e_1 \mid \text{Parents}(e_1)) * p(e_2 \mid \text{Parents}(e_2)) \ldots$$

## Particle Filtering

- Initialize
  - Draw N particles (i.e. samples) for $X_0$ from the prior distribution $p(X_0)$
- Propagate
  - Propagate each particle forward by sampling a value $x_{t+1}$ from $p(X_{t+1} \mid X_t)$
- Weight
  - Weight each particle by $p(e_{t+1} | X_{t+1}=x_{t+1})$
- Resample
  - Generate N new particles by sampling proportional to the weights. The new particles are unweighted

## Particle Filtering Example

- Particles: track samples of states rather than an explicit distribution



# Summary

- Dynamic Bayesian networks are useful when quantity of interest changes over time
- Hidden Markov Model is the simplest type of DBN
- Exact inference
- Approximate inference