Time and uncertainty

- The world changes! We need to track and predict it.
  - Robot localization, diabetes management, vehicle diagnosis, etc.

- **Basic idea:** copy state and evidence variables for each time step.

  \[ X_t = \text{set of unobservable state variables at time } t \]
  
  E.g. *BloodSugar*, *StomachContents*, etc.

  \[ E_t = \text{set of observable evidence variables at time } t \]
  
  E.g. *MeasuredBloodSugar*, *PulseRate*, *FoodEaten*, etc.

- Assume **discrete time** (step size depends on problem.)
Temporal variables

- Consider a variable that changes over time, $X_t$.
  - Notation for time series sequence: $X_{t-p:k} = X_p, X_{t+1}, ..., X_{t+k}$

- Changes in the state variable over time are determined by the transition model, $P(X_t | X_{0:t-1})$ (= "the probability of being in state $X=x$ at time $t$, conditioned on the previous states visited at times 0 to $t-1"$).

- In some cases, the state variable is (partially) observed through a sensor model, $P(E_t | X_0:t, E_{0:t-1})$

- These are defined over the full domains of $X$ and $E$.

Markov processes (Markov chains)

- **Markov assumption**: $X_t$ depends on bounded subset of $X_{0:t-1}$.
  - First-order Markov process: $P(X_t | X_{0:t-1}) = P(X_t | X_{t-1})$
  - Second-order Markov process: $P(X_t | X_{0:t-1}) = P(X_t | X_{t-1}, X_{t-2})$
Markov processes (Markov chains)

- **Markov assumption**: $X_t$ depends on bounded subset of $X_{0:t-1}$.
  - First-order Markov process: $P(X_t \mid X_{0:t-1}) = P(X_t \mid X_{t-1})$
  - Second-order Markov process: $P(X_t \mid X_{0:t-1}) = P(X_t \mid X_{t-2}, X_{t-1})$

- **Stationary process assumption**: transition and sensor model are fixed for all time steps, $t$: $P(X_t \mid X_{0:t-1}) = P(X_t \mid X_{1:t})$
  \[ P(E_t \mid X_{0:t}, E_{0:t-1}) = P(E_t \mid X_t) \]

Example

\[ \begin{align*}
R_{t-1} & \quad P(R_{t-1}) \\
\text{Rain}_{t-1} & \quad t & 0.3 & \quad f & 0.7
\end{align*} \]

\[ \begin{align*}
R_t & \quad P(R_t) \\
\text{Rain}_t & \quad R_t & 0.2 & \quad P(U_{t-1}) & 0.8
\end{align*} \]

\[ \begin{align*}
\text{Rain}_{t+1} & \quad X_t \\
\text{Rain}_t & \quad E_t \\
\text{Umbrella}_{t-1} & \quad \text{Umbrella}_t & \quad \text{Umbrella}_{t+1}
\end{align*} \]
Example

\[
\begin{array}{c}
\text{Rain}_{t-1} \quad \text{Rain}_t \quad \text{Rain}_{t+1} \\
\downarrow \quad \quad \quad \downarrow \quad \quad \quad \downarrow \\
\text{Umbrella}_{t-1} \quad \text{Umbrella}_t \quad \text{Umbrella}_{t+1}
\end{array}
\]

\[
\begin{array}{c}
P(R_t|\text{Rain}_{t-1}) \\
0.7 \\
0.3
\end{array}
\]

\[
\begin{array}{c}
P(R_{t+1}|\text{Rain}_t) \\
0.9 \\
0.2
\end{array}
\]

\[X_t, E_t\]

Caveat: (Ignore this problem for today’s lecture.)
- First-order Markov assumption not exactly true in real world!
- Possible fixes:
  - Increase order of Markov process.
  - Augment state, e.g. add \(\text{Temp}_t, \text{Pressure}_t\)

Applications

- Text processing
- Speech recognition
- Biological sequences
- Robot navigation
- Financial time series

Many, many more!
- Other related applications (e.g. in computer vision) assume a spatial version, where variable is replicated in X-Y space, rather than over time steps.
Inference tasks in temporal models

1. **Filtering**: $P(X_t|e_{1:t})$
   
   *E.g.* In text analysis, infer topic probability, based on observed words.

2. **Prediction**: $P(X_{t+k}|e_{1:t})$ for $k>0$
   
   *E.g.* In financial modeling, predict prices based on previous trends.
Inference tasks in temporal models

1. **Filtering:** \( P(X_t|e_{1:t}) \)
   
   *E.g.* In text analysis, infer topic probability, based on observed words.

2. **Prediction:** \( P(X_{t+k}|e_{1:t}) \) for \( k>0 \)
   
   *E.g.* In financial modeling, predict prices based on previous trends.

3. **Smoothing:** \( P(X_k|e_{1:t}) \) for \( 0\leq k<t \)
   
   *E.g.* In genome analysis, produce plausible sequence alignments.

4. **Most likely explanation:** \( \arg\max_{X_{1:t}} P( X_{1:t} | e_{1:t} ) \)
   
   *E.g.* In speech recognition, infer most likely words, based on observed phonemes.

It is commonly assumed that \( X \) is the latent state and \( E \) is the evidence. In that case we treat \( X \) as **missing data**.
1. Filtering

- Goal: Devise a recursive state estimation algorithm.

\[ P(X_{t+1} | e_{t+1}) = f(e_{t+1}, P(X_t | e_{t+1})) \quad \text{[Define } f(\text{)]} \]

\[ P(X_{t+1} | e_{t+1}) = P(X_{t+1} | e_{t+1}, e_{t+1}) \]
1. Filtering

• Goal: Devise a recursive state estimation algorithm.

\[ P(X_{t+1} | e_{t+1}) = f(e_{t+1}, P(X_t | e_{1:t})) \]  
[Define \( f() \)]

\[ P(X_{t+1} | e_{t+1}) = P(X_{t+1} | e_t, e_{t+1}) \]

\[ = a P(e_{t+1} | X_{t+1}, e_t) P(X_{t+1} | e_t) \]  
[Baye's rule]

\[ = a P(e_{t+1} | X_{t+1}) P(X_{t+1} | e_t) \]  
[Cond. Indep.]

\[ = a P(e_{t+1} | X_{t+1}) \sum_{x_t} P(X_{t+1} | x_t, e_{t+1}) P(x_t | e_t) \]  
[Sum out \( X_t \)]

\[ = a P(e_{t+1} | X_{t+1}) \sum_{x_t} P(X_{t+1} | x_t) P(x_t | e_t) \]  
[Cond. Indep.]
1. Filtering

• Goal: Devise a recursive state estimation algorithm.

\[ P(X_{t+1} | e_{1:t+1}) = f(e_{t+1}, P(X_t | e_{1:t})) \quad \text{[Define } f()] \]

\[ P(X_{t+1} | e_{1:t+1}) = P(X_{t+1} | e_{1:t}, e_{t+1}) \]

\[ = a P(e_{t+1} | X_{t+1}, e_{1:t}) P(X_{t+1} | e_{1:t}) \quad \text{[Baye's rule]} \]

\[ = a P(e_{t+1} | X_{t+1}) P(X_{t+1} | e_{1:t}) \quad \text{[Cond. Indep.]} \]

\[ = a P(e_{t+1} | X_{t+1}) \sum_{x_t} P(X_{t+1} | x_t, e_{1:t}) P(x_t | e_{1:t}) \quad \text{[Sum out } X_t\text{]} \]

\[ = a P(e_{t+1} | X_{t+1}) \sum_{x_t} P(X_{t+1} | x_t) P(x_t | e_{1:t}) \quad \text{[Cond. Indep.]} \]

Filtering = normalization, estimation, prediction, recurrence.

• Notation: \( f_{1:t+1} = \text{FORWARD}(f_{1:t}, e_{t+1}) \), where \( f_{1:t} = P(X_t | e_{1:t}) \)

• Time and space constant (independent of \( t \)).
Aim: 

I.e., 

\[
P(X_{t+1}) = \sum_{X_t} P(X_{t+1} | X_t) P(X_t)
\]

and

\[
P(X_t | X_{t+1})
\]
**Filtering example**

Transition model:

<table>
<thead>
<tr>
<th>$R_{t-1}$</th>
<th>$P(R_t)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t$</td>
<td>0.7</td>
</tr>
<tr>
<td>$f$</td>
<td>0.3</td>
</tr>
</tbody>
</table>

Sensor model:

<table>
<thead>
<tr>
<th>$R_t$</th>
<th>$P(U_t)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t$</td>
<td>0.9</td>
</tr>
<tr>
<td>$f$</td>
<td>0.2</td>
</tr>
</tbody>
</table>

True: 0.500  
False: 0.500

$P(R_1 | U_1=1) = ?$

More generally: $P(R_t | U_t=1) = ?$ for $t>1$

**Smoothing example**

Transition model:

<table>
<thead>
<tr>
<th>$R_{t-1}$</th>
<th>$P(R_t)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t$</td>
<td>0.7</td>
</tr>
<tr>
<td>$f$</td>
<td>0.3</td>
</tr>
</tbody>
</table>

Sensor model:

<table>
<thead>
<tr>
<th>$R_t$</th>
<th>$P(U_t)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t$</td>
<td>0.9</td>
</tr>
<tr>
<td>$f$</td>
<td>0.2</td>
</tr>
</tbody>
</table>

True: 0.500  
False: 0.500

$P(R_2 | U_1=1) = ?$

More generally: $P(R_t | U_t=1) = ?$ for $t>1$
2. Prediction

• Notice simple prediction to \( k \) steps in the future (without evidence):
\[
P(X_{t+k} | X_t) = P(X_{t+k} | X_{t+k-1}) P(X_{t+k-1} | X_{t+k-2}) \ldots P(X_{t+1} | X_t)
\]

• To estimate \( P(X_{t+k} | e_{1:t}) \) for \( k > 0 \) (=predict \( t+k \) steps in the future, using evidence \( e_{1:t} \)):
  - Use filtering for time steps \( t \) to \( t \), then apply simple prediction as above for remaining \( k \) steps.
  - Need to sum over all possible assignments of intermediates \( X_{t+1} \) to \( X_{t+k-1} \).
3. Smoothing

Compare: Prediction: $P(X_{t+1} | e_t)$ for $k > 0$

Smoothing: $P(X_t | e_t)$ for $0 < k < t$  

Why is this useful?

---

3. Smoothing

Why? Estimate of $P r(X_t | ...)$ is informed by evidence before and after.

- Divide evidence $e_{1:t}$ into $e_{t:k}$, $e_{k+1:t}$:

  $P(X_t | e_{t:k}) = P(X_t | e_{i:t}, e_{i+1:t})$

  $= a P(X_t | e_{i:t}) P(e_{i+1:t} | X_t, e_{i:t})$

  $= a P(X_t | e_{i:t}) P(e_{i+1:t} | X_t)$

  $= a f_i b_{i+1}$  

  Forward message, $f_{i:t}$, computed as in filtering.

  - Backward message $b_{k+1}$ computed by backwards recursion.

  $b_{k+1} = P(e_{k+1:t} | X_t) = a \sum_{i=k+1}^{t} P(e_{k+1:t} | X_t, x_{i+1}) P(x_{i+1} | X_t)$

    $= a \sum_{i=k+1}^{t} P(e_{k+1:t} | X_t, x_{i+1}) P(x_{i+1} | x_{i+1}) P(x_{i+1} | X_t)$
Smoothing example

COMP-424: Artificial intelligence 27 Joelle Pineau
Aim: I.e.,

\[ P_{t+1} = \alpha \]

\[ t \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]

\[ = \]

\[ (\ldots) \]

\[ \ldots \]