Outline

- Inference in temporal models
- Markov assumptions
- Hidden Markov Models
Dynamic processes

We have seen that we can representing knowledge in a static world in Bayesian networks.

Now, we will look at ways of modelling processes over time, taking into consideration:

- Uncertainty.
- Dependencies.
Example

Anna works for Beth.

Anna has noticed that whether Beth accepts her suggestions for improvements to the working environment depends heavily on Beth's mood:

- If Beth is happy, she always accepts Anna's suggestions
- If Beth is angry, she never accepts Anna's suggestions

The problem is that Beth has a very good poker face: it is impossible to know what her mood is like from looking at her.

Luckily, over time, Anna has made some interesting observations. In order to be able to put her suggestions forward at strategic times, Anna wants to use these observations to build a model to help her predict what mood Beth is in.
Example

Anna's observations are the following

- Beth is either angry the whole day or happy the whole day
- If Beth is having a bad (angry) day, she is more likely to wear a red sweater
- If Beth was angry yesterday, she is less likely to be angry today
- If Beth is angry, she is more likely to slam her office door shut when she arrives in the morning
Example

Anna starts out using the following random variables:

- $A$ represents whether Beth is angry today. It takes the values $a$ (angry) and $\neg a$ (happy)
- $AY$ represents whether Beth was angry yesterday. It takes the values $ay$ and $\neg ay$
- $R$ represents whether Beth has a red sweater on. It takes the values $r$ and $\neg r$
- $S$ represents whether Beth slams her door in the morning. It takes the values $s$ and $\neg s$
Example

Anna summarizes the conditional probabilities she estimates in conditional probability tables as follows:

\[
\begin{array}{c|c}
P(A \mid Ya) & P(A \mid \neg ya) \\
\hline
Ya & P(a) \\
Ya & 1/5 \\
\neg ya & 1/3 \\
\end{array}
\]

\[
\begin{array}{c|c}
P(R \mid A) & P(S \mid A) \\
\hline
A & P(r) \\
a & 2/3 \\
\neg a & 1/4 \\
\end{array}
\]

But how to make this model talk about more than one day?
Example

Anna decides to use **three unique random variables** for each day:

\[ A_0, A_1, A_2, \ldots \text{ will represent Beth's mood} \]

\[ R_0, R_1, R_2, \ldots \text{ will represent Beth's clothing} \]

\[ S_0, S_1, S_2, \ldots \text{ will represent Beth's door handling} \]
Modelling the example

As far as Anna's model is concerned, Beth's mood depends only on her mood the previous day:

\[ P(A_t \mid A_{0:t-1}) = P(A_t \mid A_{t-1}) \]

\( X_{a:b} \) denotes the set of variable from \( X_a \) to \( X_b \)

Similarly, her choice of clothing and her handling of doors depend only on her mood today:

\[ P(R_t \mid A_{0:t}, R_{0:t-1}, S_{0:t}) = P(R_t \mid A_t) \]
\[ P(S_t \mid A_{0:t}, R_{0:t}, S_{0:t-1}) = P(S_t \mid A_t) \]
Modelling the example

The conditional probabilities can now be reformulated:

\[
\begin{array}{c|c}
A_t & P(A_t | A_{t-1}) \\
\hline
A_{t-1} & P(a_t) \\
\hline
a_{t-1} & 1/5 \\
\neg a_{t-1} & 1/3 \\
\end{array}
\]

\[
\begin{array}{c|c}
A_t & P(R_t | A_t) \\
\hline
A_t & P(r_t) \\
\hline
a_t & 2/3 \\
\neg a_t & 1/4 \\
\end{array}
\]

\[
\begin{array}{c|c}
A_t & P(S_t | A_t) \\
\hline
A_t & P(s_t) \\
\hline
a_t & 1/2 \\
\neg a_t & 1/4 \\
\end{array}
\]

This gives Anna the following formula for the full joint distribution over all variables up to day \( t \):

\[
P(A_{0:t}, R_{1:t}, S_{1:t}) = P(A_0) \prod_{i=1}^{t} P(A_i | A_{i-1})P(R_i | A_i)P(S_i | A_i)
\]
Modelling the example

Anna, industrious as ever, decides to illustrate her model as a Bayesian network:

![Bayesian network diagram]

- $A_0$ to $A_1$ to $A_2$ to $A_3$ to $\ldots$
- $R_1$ to $R_2$ to $R_3$
- $S_1$ to $S_2$ to $S_3$
What has Anna achieved so far?

• She has described the world, seen over time, as a number of snapshots (or time slices).
• Each snapshot contains some random variables that describe the likelihoods of various states of the world at that time.
• Some of the variables \((R_t, S_t)\) are observable.
• Some of the variables \((A_t)\) are unobservable or hidden.

In Anna’s case, the snapshots represent days, but this is highly application-dependent.

There are also tools for modeling continuous time, but they are not covered in this course.
State and evidence variables and Markov assumptions

- The **unobservable variables** in a model such as Anna's are called **state variables**.
- The **observable variables** are called **evidence variables**.

In Anna's model, the state variables depend only on the state variables in the previous snapshot:

\[ P(A_t \mid A_{0:t-1}) = P(A_t \mid A_{t-1}) \]

- This is rather strict, but in general, we want the current state variables to depend only on a **fixed number of previous time slices**. This is the **Markov assumption**.
- Processes that satisfy the Markov assumption are called **Markov chains** or **Markov processes** after the Russian mathematician Andrei Markov.
- Notice that the Markov assumption is often used **even when it doesn't hold**.
Stationary processes

• In Anna's model, every $A_t$ depends on the previous state $A_{t-1}$ in the same way.

• This means that her model is a stationary process.

• The laws that govern change in the model do not themselves change over time.

• These conditional probabilities define our transition model.
Sensor Markov assumption

Anna's model also satisfies the sensor Markov assumption: The evidence variables depend only on the state variables from the same time slice.

\[ P(R_t \mid A_{0:t}, R_{1:t-1}, S_{1:t}) = P(R_t \mid A_t) \]

Every \( R_t \) and \( S_t \) depend on their corresponding \( A_t \) in the same way. These conditional probabilities define our sensor model.
Inference

Given a temporal model, there are four basic inference tasks:

**Filtering.** Given all the evidence to date, what is the probability distributions for our current state variables? In Anna's case, this amounts to calculating

\[ P(A_t | r_{1:t}, s_{1:t}) \]

**Prediction.** Given all the evidence to date, what is the probability distribution for the state variables in some future snapshot? For Anna, this is

\[ P(A_{t+k} | r_{1:t}, s_{1:t}) \]
Inference

**Smoothing.** Given all the evidence to date, what is the probability distribution for the state variables in some past snapshot? For Anna:

\[ P(A_{t-k} \mid r_{1:t}, s_{1:t}) \]

**Most likely explanation.** Given all the evidence to date, what is the most likely sequence of actual state values to date? For Anna:

\[ \operatorname{argmax}_{a_{1:t}} P(a_{1:t} \mid r_{1:t}, s_{1:t}) \]
Filtering and prediction

Consider the filtering question:

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

Suddenly computing this distribution after \( t \) timesteps is very heavy.

Instead, we aim at doing updates: We continuously keep track of the distribution over the state variables, and updating it snapshot by snapshot. We want to compute

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

for some appropriate function \( f \).
Example

Is Beth angry on day 0?

Here, Anna can only use her best guess. Let's say she estimates
\[ P(A_0) = \langle 0.4, 0.6 \rangle \]
On day 1, Beth wears red, but does not slam her door, so
\[ R_1 = r_1 \text{ and } S_1 = \neg s_1. \]

We can first compute the distribution over \( A_1 \), which is only based on \( A_0 \):

\[
P(A_1) = P(A_1 \mid a_0)P(a_0) + P(A_1 \mid \neg a_0)P(\neg a_0) =
\langle 1/5, 4/5 \rangle \frac{2}{5} + \langle 1/3, 2/3 \rangle \frac{3}{5} = \langle 2/25, 8/25 \rangle + \langle 3/15, 6/15 \rangle =
\langle 42/150, 108/150 \rangle = \langle 0.28, 0.72 \rangle
\]
Example

Anna can now compute how likely Beth is to be angry on day 1:

\[
P(A_1 \mid r_1, \neg s_1) = \alpha P(r_1, \neg s_1 \mid A_1)P(A_1) =
\]
\[
\alpha P(r_1 \mid A_1)P(\neg s_1 \mid A_1)P(A_1) =
\]
\[
\alpha \langle 2/3, 1/4 \rangle \langle 1/2, 3/4 \rangle \langle 0.28, 0.72 \rangle
\]
\[
\alpha \langle 0.28/3, 2.16/16 \rangle \approx \langle 0.41, 0.59 \rangle
\]

How do we generalize this to work for any \( A_t \)?
Bayes' rule with background evidence

We can formulate Bayes' rule for distributions with some given background evidence:

\[ P(X \mid Y, e) = \frac{P(Y \mid X, e)P(X \mid e)}{P(Y \mid e)} \]
Computing an update rule

\[
P(X_{t+1} \mid e_{1:t+1}) = P(X_{t+1} \mid e_{t+1}, e_{1:t})
\]
\[
= \alpha P(e_{t+1} \mid X_{t+1}, e_{1:t}) P(X_{t+1} \mid e_{1:t}) \quad (\text{Bayes' rule})
\]
\[
= \alpha P(e_{t+1} \mid X_{t+1}) P(X_{t+1} \mid e_{1:t}) \quad (\text{sensor Markov assumption})
\]

We also have

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

Which, thank’s to the Markov assumption, becomes

\[
P(X_{t+1} \mid e_{1:t}) = \sum_{x_t} P(X_{t+1} \mid x_t) P(x_t \mid e_{1:t})
\]
Computing an update rule

Putting this all together we get

\[ P(X_{t+1} \mid e_{1:t+1}) = \alpha P(e_{t+1} \mid X_{t+1}) \sum_{x_t} P(X_{t+1} \mid x_t) P(x_t \mid e_{1:t}) \]

- We have \( P(e_{t+1} \mid X_{t+1}) \) from our sensor model
- We have \( P(X_{t+1} \mid x_t) \) from our transition model
- We assume that we have have \( P(x_t \mid e_{1:t}) \) from the previous snapshot.

Thus we have a recursive update rule.
Example

\[ P(X_{t+1} \mid e_{1:t+1}) = \alpha P(e_{t+1} \mid X_{t+1}) \sum_{x_t} P(X_{t+1} \mid x_t) P(x_t \mid e_{1:t}) \]

What does this formula translate to for Anna?

\[ P(A_{t+1} \mid r_{1:t+1}, s_{1:t+1}) = \alpha P(r_{t+1} \mid A_{t+1}) P(s_{t+1} \mid A_{t+1}) \sum_{a_t \in \{a_t, \neg a_t\}} P(A_{t+1} \mid a_t) P(a_t \mid r_{1:t}, s_{1:t}) \]
Example

Assume that on day 2, Beth is dressed in red and slams her door. Let's try to compute how likely it is that she is angry.

\[
P(A_{t+1} \mid r_{1:t+1}, s_{1:t+1}) = \alpha P(r_{t+1} \mid A_{t+1}) P(s_{t+1} \mid A_{t+1}) \sum_{a_t \in \{a_t, \neg a_t\}} P(A_{t+1} \mid a_t) P(a_t \mid r_{1:t}, s_{1:t})
\]

In our case, this becomes

\[
P(A_2 \mid r_{1:2}, s_{1:2}) = \alpha P(r_2 \mid A_2) P(s_2 \mid A_2) \sum_{a_1 \in \{a_1, \neg a_1\}} P(A_2 \mid a_1) P(a_1 \mid r_1, \neg s_1)
\]
Example

\[
P(A_2 \ | \ r_{1:2}, s_{1:2}) = \alpha P(r_2 \ | \ A_2) P(s_2 \ | \ A_2) \sum_{a_1 \in \{a_1, \neg a_1\}} P(A_2 \ | \ a_1) P(a_1 \ | \ r_1, \neg s_1)
\]

We look at the parts of this expression one at a time:

\(P(r_2 \ | \ A_2)\) we can get from our sensor model:

\[
\begin{array}{c|c}
  P(R_t \ | \ A_t) \\
  \hline
  a_t & P(r_t) \\
  \hline
  t & 2/3 \\
  f & 1/4
\end{array}
\]

This means that \(P(r_2 \ | \ A_2) = \langle 2/3, 1/4 \rangle\)

In a similar way, we get \(P(s_2 \ | \ A_2) = \langle 1/2, 1/4 \rangle\)
Example

$$\sum_{a_1 \in \{a_1, \neg a_1\}} P(A_2 \mid a_1) P(a_1 \mid r_1, \neg s_1) =$$

$$P(A_2 \mid a_1) P(a_1 \mid r_1, \neg s_1) + P(A_2 \mid \neg a_1) P(\neg a_1 \mid r_1, \neg s_1)$$

We have

- $$P(A_2 \mid a_1) = \langle 1/5, 4/5 \rangle$$
- $$P(a_1 \mid r_1, \neg s_1) \approx 0.28$$
- $$P(A_2 \mid \neg a_1) = \langle 1/3, 2/3 \rangle$$
- $$P(\neg a_1 \mid r_1, \neg s_1) \approx 0.72$$
Example

\[ P(A_2 \mid r_{1:2}, s_{1:2}) = \]

\[ \alpha P(r_2 \mid A_2) P(s_2 \mid A_2) \sum_{a_1 \in \{a_1, \neg a_1\}} P(A_2 \mid a_1) P(a_1 \mid r_1, \neg s_1) \]

Summing up, we get

\[ P(A_2 \mid r_{1:2}, s_{1:2}) = \]

\[ \alpha \langle 2/3, 1/4 \rangle \langle 1/2, 1/4 \rangle \left[ \langle 1/5, 4/5 \rangle 0.28 + \langle 1/3, 2/3 \rangle 0.72 \right] \approx \]

\[ \alpha \langle 0.099, 0.044 \rangle \approx \langle 0.69, 0.31 \rangle \]
Complexity

How much computation is needed for each update of course depends on the number of variables and their ranges.

The important thing to notice is that each update step takes more or less the same amount of resources.

In other words, the update effort does not grow with $t$. 
Hidden Markov Models (HMMs)

Unbeknownst to herself, Anna has actually entered the higher realms of probabilistic modelling:

She has created a Hidden Markov Model.

The defining feature of such models is that they have exactly one state variable per snapshot and that this variable is discrete.
Example

Once she has realized this, she quickly realizes that she can speed up her computations.

In fact, her transition model, $P(A_t | A_{t-1})$ can be expressed as a simple matrix:

\[
\begin{array}{c|c}
A_t & P(A_t | A_{t-1}) \\
\hline
a_{t-1} & P(a_t) \\
\hline
t & 1/5 \\
f & 1/3 \\
\end{array}
\]

becomes

\[
\begin{pmatrix}
1/5 & 4/5 \\
1/3 & 2/3
\end{pmatrix}
\]

For example

\[
P(A_1 | A_0) = \langle 2/5, 3/5 \rangle \begin{pmatrix}
1/5 & 4/5 \\
1/3 & 2/3
\end{pmatrix} = \langle 0.28, 0.72 \rangle
\]
Advantage of HMM on Sequential Data

- **Natural model structure**: doubly stochastic process
  - transition parameters model *temporal* variability
  - output distribution model *spatial* variability
- **Efficient and good modeling tool** for
  - sequences with temporal constraints
  - spatial variability along the sequence
  - real world complex processes
- **Efficient evaluation**, decoding and training algorithms
  - Mathematically strong
  - Computationally efficient
- **Proven technology**!
  - Successful stories in many applications
Tools for HHM

- **HMM toolbox for Matlab**
  - Developed by Kevin Murphy
  - Freely downloadable SW written in Matlab (Hmm... Matlab is not free!)
  - Easy-to-use: flexible data structure and fast prototyping by Matlab
  - Somewhat slow performance due to Matlab

- **HTK (Hidden Markov toolkit)**
  - Developed by Speech Vision and Robotics Group of Cambridge University
  - Freely downloadable SW written in C
  - Useful for speech recognition research: comprehensive set of programs for training, recognizing and analyzing speech signals
  - Powerful and comprehensive, but somewhat complicate and heavy package
  - Download: [http://htk.eng.cam.ac.uk/](http://htk.eng.cam.ac.uk/)
Sources of this Lecture


• H. Björklund, slides from previous courses.