# Introduction to Time Series[1]

David Barber, Dmitry Adamskiy

University College London

[1] These slides accompany the book *Bayesian Reasoning and Machine Learning*. The book and demos can be downloaded from `www.cs.ucl.ac.uk/staff/D.Barber/brml`. Feedback and corrections are also available on the site. Feel free to adapt these slides for your own purposes, but please include a link the above website.

# Markov Models

# Time-Series

A time-series is an ordered sequence:

$$x_{a:b} = \{x_a, x_{a+1}, \ldots, x_b\}$$

So that one can consider the 'past' and 'future' in the sequence. The $x$ can be either discrete or continuous.

---

## Biology

Gene sequences. Emphasis is on understanding sequences, filling in missing values, clustering sequences, detecting patterns. Hidden Markov Models are one of the key tools in this area.

---

## Finance

Price movement prediction.

---

## Planning

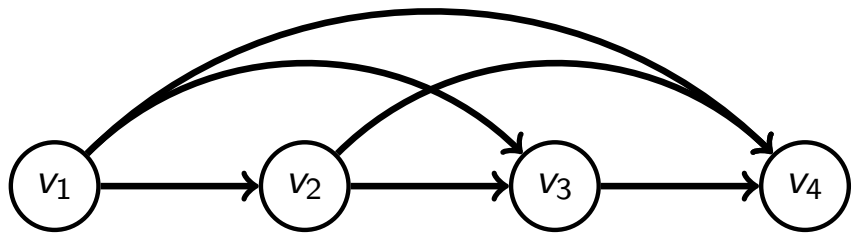
Forecasting – eg how many newspaper to deliver to retailers.

# Markov Models

For timeseries data $v_1, \ldots, v_T$, we need a model $p(v_{1:T})$. For causal consistency, it is meaningful to consider the decomposition

$$p(v_{1:T}) = \prod_{t=1}^{T} p(v_t | v_{1:t-1})$$

with the convention $p(v_t | v_{1:t-1}) = p(v_1)$ for $t = 1$.




---

## Independence assumptions

It is often natural to assume that the influence of the immediate past is more relevant than the remote past and in Markov models only a limited number of previous observations are required to predict the future.

# Markov Chain

Only the recent past is relevant:

$$p(v_t|v_1, \ldots, v_{t-1}) = p(v_t|v_{t-L}, \ldots, v_{t-1})$$

where $L \geq 1$ is the order of the Markov chain

$$p(v_{1:T}) = p(v_1)p(v_2|v_1)p(v_3|v_2) \ldots p(v_T|v_{T-1})$$

For a stationary Markov chain the transitions $p(v_t = s'|v_{t-1} = s) = f(s', s)$ are time-independent ('homogeneous').

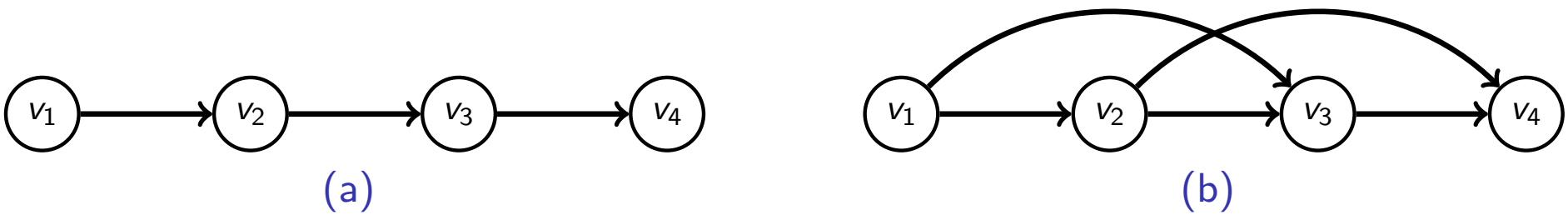



Figure: **(a)**: First order Markov chain. **(b)**: Second order Markov chain.

# Fitting Markov models (discrete variables)

## Single series

- Fitting a first-order stationary Markov chain by Maximum Likelihood corresponds to setting the transitions by counting the number of observed transitions in the sequence:

$$p(v_\tau = i | v_{\tau-1} = j) \propto \sum_{t=2}^{T} \mathbb{I}\left[v_t = i, v_{t-1} = j\right]$$

- The Maximum Likelihood setting for the initial first timestep distribution is $p(v_1 = i) \propto \sum_n \mathbb{I}\left[v_1^n = i\right]$.

---

## Multiple series

For a set of timeseries, $v_{1:T_n}^n, n = 1, \ldots, N$, the transition is given by counting all transitions across time and datapoints.

# Rock Paper Scissors

- Two people game: each player plays either Rock, Paper or Scissors.
- Paper beats Rock, Scissors beats Paper, Rock beats Scissors.
- Let's use the encoding $Rock = 1$, $Scissors = 2$, $Paper = 3$.

---

## First Order Markov Model

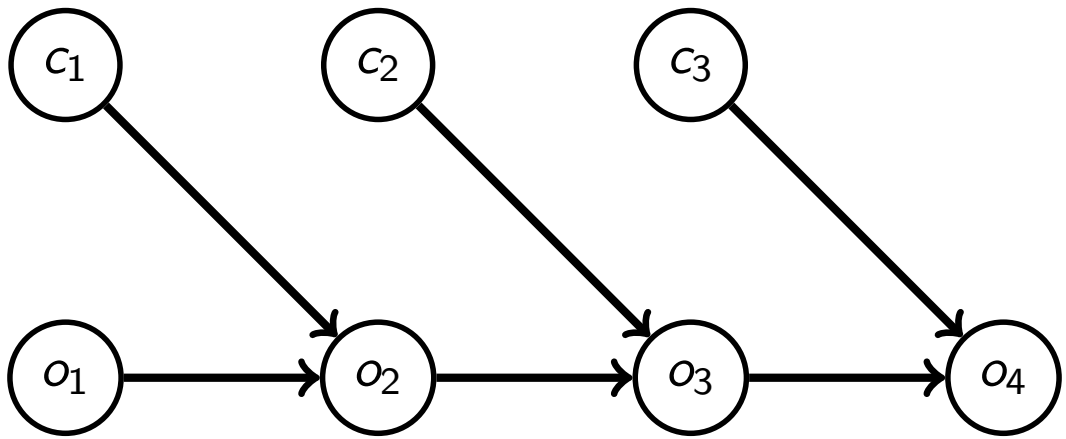
- $o_t \in \{1, 2, 3\}$ : human opponent play at time $t$.
- $c_t \in \{1, 2, 3\}$ : human opponent play at time $t$.
- The computer assumes the human moves based on what the human did on the last move.


`demoRockPaperScissorsMarkovHuman.m`

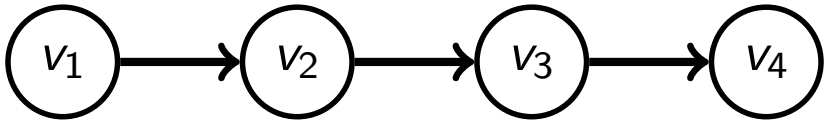
# Rock Paper Scissors

## First Order Markov Model with Computer past move

- The computer assumes the human moves based on what the human did on the last move and also on what the computer did on the last move.



# Markov Chains

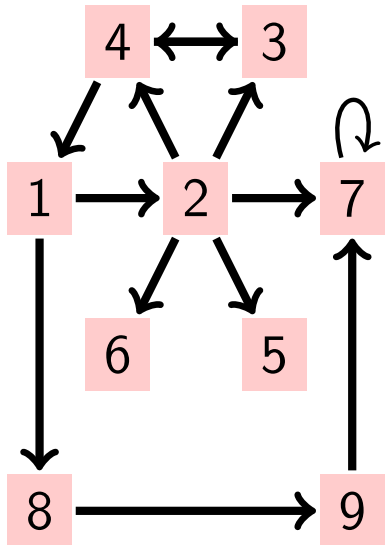



$$p(v_1, \ldots, v_T) = \underbrace{p(v_1)}_{\text{initial}} \prod_{t=2}^{T} \underbrace{p(v_t|v_{t-1})}_{\text{Transition}}$$

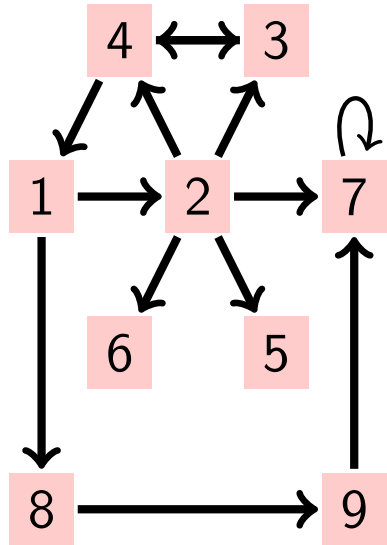
---

## State transition diagram

Nodes represent states of the variable $v$ and arcs non-zero elements of the transition $p(v_t|v_{t-1})$



# Most probable and shortest paths




- The shortest (unweighted) path from state 1 to state 7 is $1 - 2 - 7$.
- The most probable path from state 1 to state 7 is $1 - 8 - 9 - 7$ (assuming uniform transition probabilities). The latter path is longer but more probable since for the path $1 - 2 - 7$, the probability of exiting state 2 into state 7 is $1/5$.

# Equilibrium distribution

- It is interesting to know how the marginal $p(x_t)$ evolves through time:

$$p(x_t = i) = \sum_j \underbrace{p(x_t = i | x_{t-1} = j)}_{M_{ij}} p(x_{t-1} = j)$$

- $p(x_t = i)$ is the frequency that we visit state $i$ at time $t$, given we started from $p(x_1)$ and randomly drew samples from the transition $p(x_\tau | x_{\tau-1})$.
- As we repeatedly sample a new state from the chain, the distribution at time $t$, for an initial distribution $\mathbf{p}_1(i)$ is

$$\mathbf{p}_t = \mathbf{M}^{t-1} \mathbf{p}_1$$

If, for $t \to \infty$, $\mathbf{p}_\infty$ is independent of the initial distribution $\mathbf{p}_1$, then $\mathbf{p}_\infty$ is called the equilibrium distribution of the chain:

$$\mathbf{p}_\infty = \mathbf{M} \mathbf{p}_\infty$$

- The equil. distribution is proportional to the eigenvector with unit eigenvalue of the transition matrix. `demoMarkovConvergence.m`

# PageRank

Define the matrix

$$A_{ij} = \begin{cases} 1 & \text{if website } j \text{ has a hyperlink to website } i \\ 0 & \text{otherwise} \end{cases}$$

From this we can define a Markov transition matrix with elements

$$M_{ij} = \frac{A_{ij}}{\sum_{i'} A_{i'j}}$$

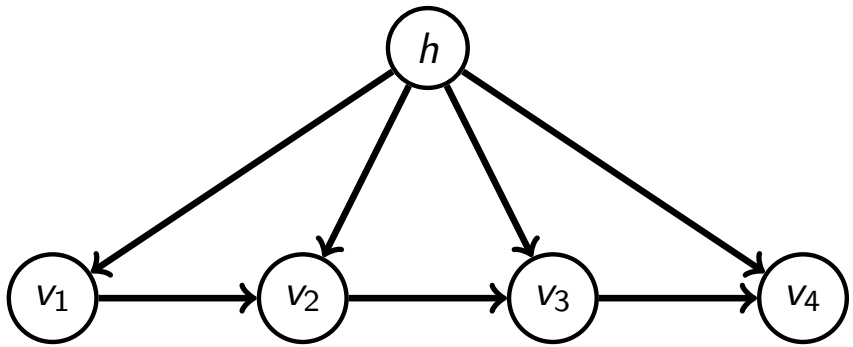
- If we jump from website to website, the equilibrium distribution component $p_\infty(i)$ is the relative number of times we will visit website $i$. This has a natural interpretation as the 'importance' of website $i$.
- For each website $i$ a list of words associated with that website is collected. After doing this for all websites, one can make an 'inverse' list of which websites contain word $w$. When a user searches for word $w$, the list of websites that contain word is then returned, ranked according to the importance of the site.

# Gene Clustering

- Consider the 20 fictitious gene sequences below presented in an arbitrarily chosen order.
- Each sequence consists of 20 symbols from the set $\{A, C, G, T\}$.
- The task is to try to cluster these sequences into two groups, based on the (perhaps biologically unrealistic) assumption that gene sequences in the same cluster follow a stationary Markov chain.

| | | | |
|---|---|---|---|
| CATAGGCATTCTATGTGCTG | CCAGTTACGGACGCCGAAAG | TGGAACCTTAAAAAAAAAAA | GTCTCCTGCCCTCTCTGAAC |
| GTGCCTGGACCTGAAAAGCC | CGGCCGCGCCTCCGGGAACG | AAAGTGCTCTGAAAACTCAC | ACATGAACTACATAGTATAA |
| GTTGGTCAGCACACGGACTG | CCTCCCCTCCCCTTTCCTGC | CACTACGGCTACCTGGGCAA | CGGTCCGTCCGAGGCACTC |
| TAAGTGTCCTCTGCTCCTAA | CACCATCACCCTTGCTAAGG | AAAGAACTCCCCTCCCTGCC | CAAATGCCTCACGCGTCTCA |
| GCCAAGCAGGGTCTCAACTT | CATGGACTGCTCCACAAAGG | AAAAAAACGAAAAACCTAAG | GCGTAAAAAAAGTCCTGGGT |

# Mixture of Markov models




- The discrete hidden variable $\mathrm{dom}(h) = \{1, \ldots, H\}$ indexes the Markov chain

$$\prod_t p(v_t | v_{t-1}, h)$$

- Such models can be useful as simple sequence clustering tools.

# Mixture of Markov models

Given a set of sequences $\mathcal{V} = \{v_{1:T}^n, n = 1, \ldots, N\}$, how might we cluster them?

- We can define a mixture model for a single sequence $v_{1:T}$.
- Here we assume each component model is first order Markov

$$p(v_{1:T}) = \sum_{h=1}^{H} p(h)p(v_{1:T}|h) = \sum_{h=1}^{H} p(h) \prod_{t=1}^{T} p(v_t|v_{t-1}, h)$$

- Clustering can then be achieved by finding the maximum likelihood parameters $p(h)$, $p(v_t|v_{t-1}, h)$ and subsequently assigning the clusters according to $p(h|v_{1:T}^n)$.
- We will learn later in the course how to learn parameters in these 'latent variable' models. A common approach is the 'EM' algorithm which we will discuss later.

# Clustering Genes

- After running the EM maximum likelihood algorithm, we can then assign each of the sequences by examining $p(h = 1|v_{1:T}^n)$.
- If this posterior probability is greater than 0.5, we assign it to cluster 1, otherwise to cluster 2.
- Using this procedure, we find the following clusters:

| | |
|---|---|
| CATAGGCATTCTATGTGCTG | TGGAACCTTAAAAAAAAAAA |
| CCAGTTACGGACGCCGAAAG | GTCTCCTGCCCTCTCTGAAC |
| CGGCCGCGCCTCCGGGAACG | GTGCCTGGACCTGAAAAGCC |
| ACATGAACTACATAGTATAA | AAAGTGCTCTGAAAACTCAC |
| GTTGGTCAGCACACGGACTG | CCTCCCCTCCCCTTTCCTGC |
| CACTACGGCTACCTGGGCAA | TAAGTGTCCTCTGCTCCTAA |
| CGGTCCGTCCGAGGCACTCG | AAAGAACTCCCCTCCCTGCC |
| CACCATCACCCTTGCTAAGG | AAAAAAACGAAAAACCTAAG |
| CAAATGCCTCACGCGTCTCA | GCGTAAAAAAAGTCCTGGGT |
| GCCAAGCAGGGTCTCAACTT | |
| CATGGACTGCTCCACAAAGG | |

where sequences in the first column are assigned to cluster 1, and sequences in the second column to cluster 2.

`demoMixMarkov.m`

# Hidden Markov Models

# Hidden Markov Models

The HMM defines a Markov chain on hidden variables $h_{1:T}$. The observed variables depend on the hidden variables through an emission $p(v_t|h_t)$. This defines a joint distribution

$$p(h_{1:T}, v_{1:T}) = p(v_1|h_1)p(h_1)\prod_{t=2}^{T} p(v_t|h_t)p(h_t|h_{t-1})$$

$p(h_t|h_{t-1})$ and $p(v_t|h_t)$ are constant through time.

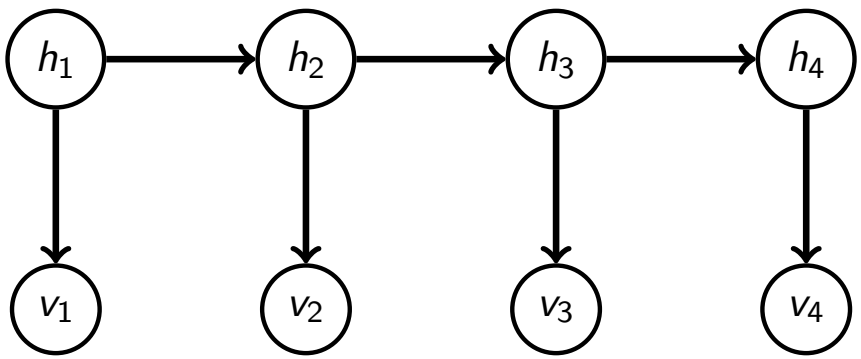



Figure: A first order hidden Markov model with 'hidden' variables $\mathrm{dom}(h_t) = \{1, \ldots, H\}$, $t = 1 : T$. The 'visible' variables $v_t$ can be either discrete or continuous.

*Probably the most common timeseries model in all of engineering/biology/physical science.*

# HMM parameters

## Transition Distribution

For a stationary HMM the transition distribution $p(h_{t+1}|h_t)$ is defined by the $H \times H$ transition matrix

$$A_{i',i} = p(h_{t+1} = i'|h_t = i)$$

and an initial distribution

$$a_i = p(h_1 = i).$$

---

## Emission Distribution

For a stationary HMM and emission distribution $p(v_t|h_t)$ with discrete states $v_t \in \{1, \ldots, V\}$, we define a $V \times H$ emission matrix

$$B_{i,j} = p(v_t = i|h_t = j)$$

For continuous outputs, $h_t$ selects one of $H$ possible output distributions $p(v_t|h_t)$, $h_t \in \{1, \ldots, H\}$.

# The classical inference problems

| | | | |
|---|---|---|---|
| **Filtering** | (Inferring the present) | $p(h_t \mid v_{1:t})$ | |
| **Prediction** | (Inferring the future) | $p(h_t \mid v_{1:s})$ | $t > s$ |
| **Smoothing** | (Inferring the past) | $p(h_t \mid v_{1:u})$ | $t < u$ |
| **Likelihood** | | $p(v_{1:T})$ | |
| **Most likely path** | (Viterbi alignment) | $\operatorname{argmax}_{h_{1:T}} p(h_{1:T} \mid v_{1:T})$ | |

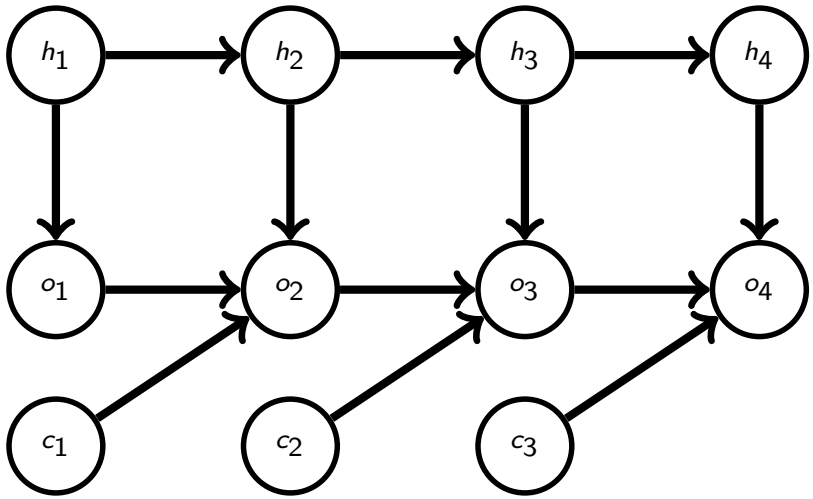
For prediction, one is also often interested in $p(v_t \mid v_{1:s})$ for $t > s$.

---

## Uses of the HMM

- Biology: gene sequence analysis
- Computer Vision: tracking of people in videos
- Signal Processing: cleaning up noise corrupted music signals
- Speech Recognition (dominant approach until recently)
- Engineering: the famous 'Kalman Filter' is a special case of a HMM (with continuous variables)
- Weather Forecasting
- Financial prediction, product purchase prediction, modelling the economy
- Military: tracking ballistic objects
- ... and many more ...

# Rock Paper Scissors, again!




`demoHMMRockPaperScissors.m`

- As we gather information about the plays the human and computer makes, we can calculate the filtered distribution $p(h_t|o_{1:t-1}, c_{1:t-1})$ of the likely strategy that the human is currently playing.
- Strategies are: random, do what you did last time, do what computer did last time, do something different to your and computer last time, cycle around rock paper scissors.
- We can use this to then predict what move the human is likely to make at the next timestep.