

Hidden Markov models (HMMs)

Timo Adam

4th Summer school on modern topics in time series analysis

Klagenfurt, Austria

September 17, 2024

Preliminary remarks

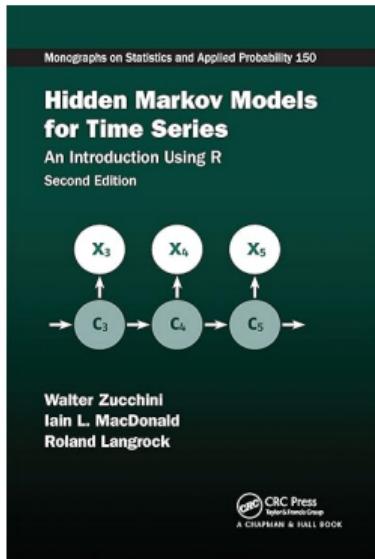
My background:



- 2016–2020: PhD in economics at Bielefeld University, Germany (flexible extensions of HMMs, applications in ecology and economics).
- 2020–2022: postdoc at the University of St Andrews, UK (method development, ecological applications of HMMs).
- 2022–2023: postdoc at the University of Copenhagen, Denmark (method development, ecological applications of HMMs).
- Since 2023: assistant professor of statistical modelling at Bielefeld University, Germany.

Preliminary remarks

Main reference — if you want to dive deeper into HMMs — Zucchini *et al.* (2016)¹:



¹Zucchini, W., MacDonald, I., and Langrock, R. (2016): *Hidden Markov models for time series: an introduction with R* (2nd edition). Chapman and Hall/CRC.

Preliminary remarks

Throughout this session, we'll use **R/RStudio**²:



All **R scripts** are available on **GitHub**:

[https://github.com/timoadam/mtsa/tree/main.](https://github.com/timoadam/mtsa/tree/main)

²If you haven't installed RStudio yet, you can download it here: <https://posit.co/download/rstudio-desktop>.

Preliminary remarks

Throughout this session, we'll use the (new) R package LaMa³ (**L**atent **M**arkov model likelihood evaluation in C++, Koslik, 2024), which provides various tools that are useful for HMMs⁴:



To install the package, run:

```
install.packages("LaMa") # install package  
library(LaMa) # load package
```

³Koslik, J.-O. (2024): LaMa: fast numerical maximum likelihood estimation for latent Markov models. *R package*.

⁴See https://janoleko.github.io/files/vignettes/LaMa/Intro_to_LaMa.pdf for an introduction to the package.

Outline

1 Introduction to HMMs (13:30–15:00)

- 1.1 Model formulation and dependence structure
- 1.2 Likelihood evaluation and model fitting
- 1.3 Model selection, model checking, and state decoding

2 Extensions of the basic HMM (15:30–16:45)

- 2.1 Multivariate HMMs
- 2.2 Covariate-dependent state processes

3 Summary, questions, etc. (16:45–17:00)

Outline

1 Introduction to HMMs (13:30–15:00)

1.1 Model formulation and dependence structure

1.2 Likelihood evaluation and model fitting

1.3 Model selection, model checking, and state decoding

■ Extensions of the basic HMM (15:30–16:45)

2.1 Multivariate HMMs

2.2 Covariate-dependent state processes

■ Summary, questions, etc. (16:45–17:00)

1.1 Model formulation and dependence structure

What **types of data** can we model using HMMs? Some examples:



Stock returns



Elephant movements



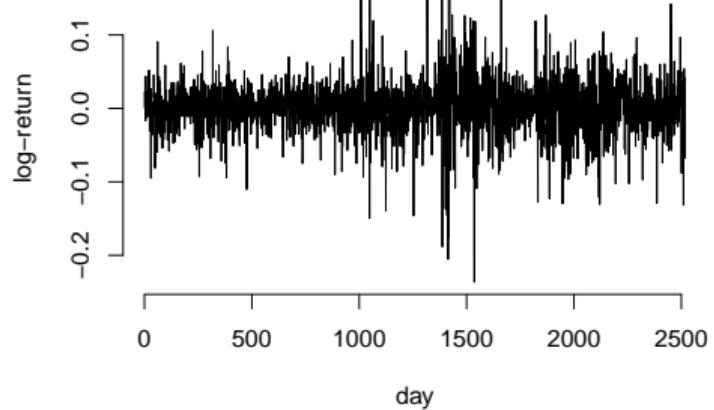
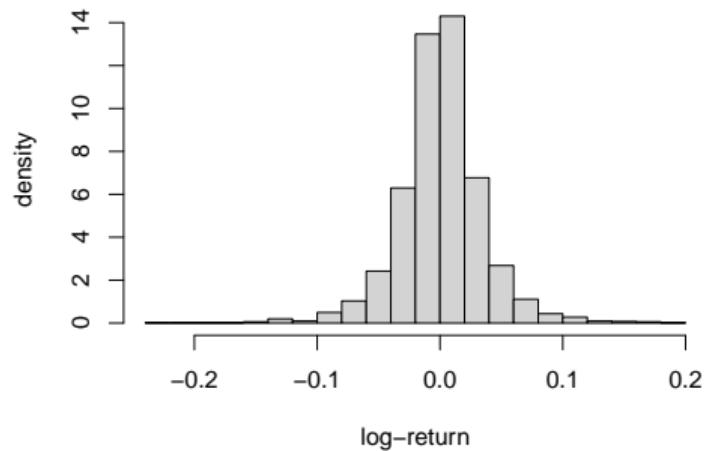
Narwhal dives



Eruption times

1.1 Model formulation and dependence structure

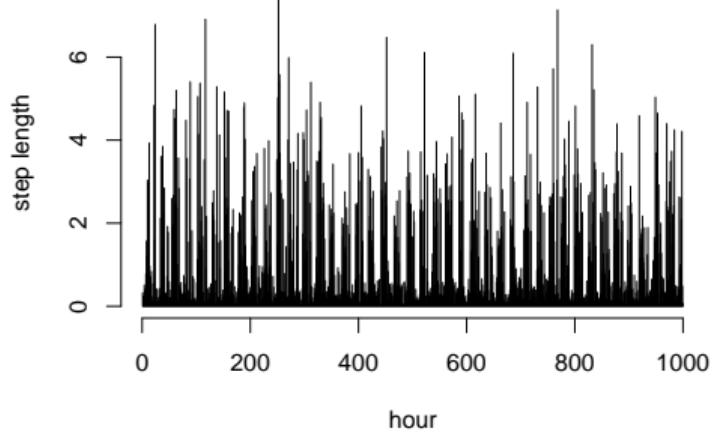
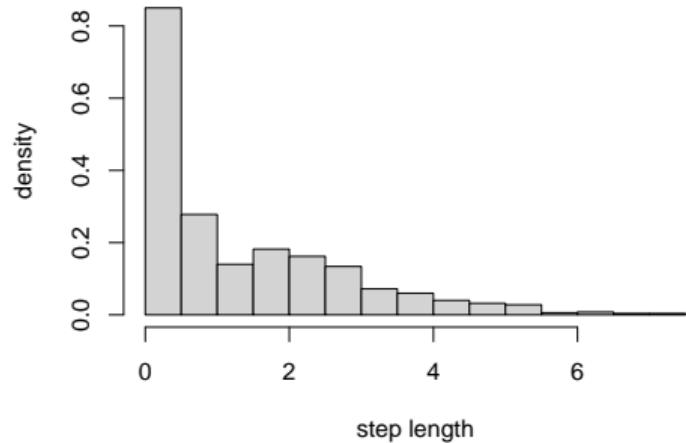
Example 1 — daily log-returns of the Tesla stock:



Key features: periods of low and high volatility ↗ state-switching over time.

1.1 Model formulation and dependence structure

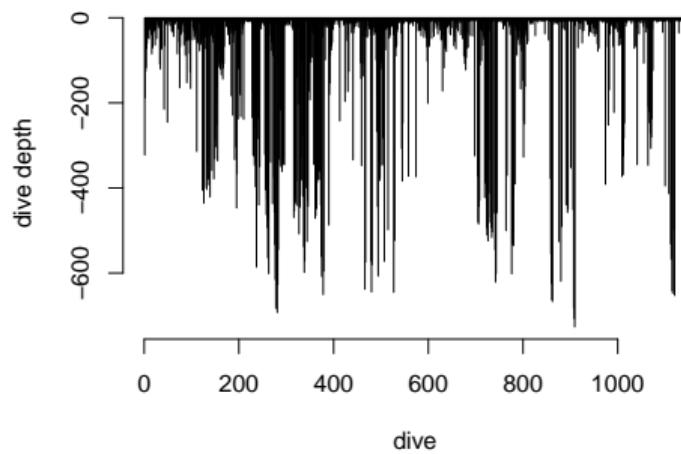
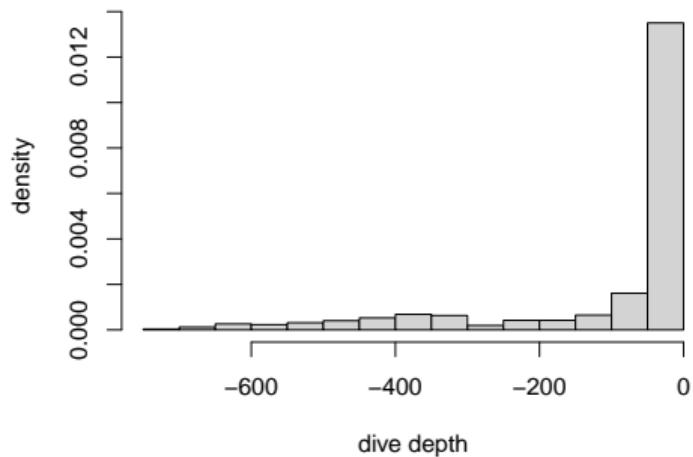
Example 2 — hourly distances travelled by an elephant:



Key features: periods of short and long steps ↗ state-switching over time.

1.1 Model formulation and dependence structure

Example 3 — dive depths of a narwhal:



Key features: bouts of shallow and deep dives ↗ state-switching over time.

1.1 Model formulation and dependence structure

From the statistical perspective...

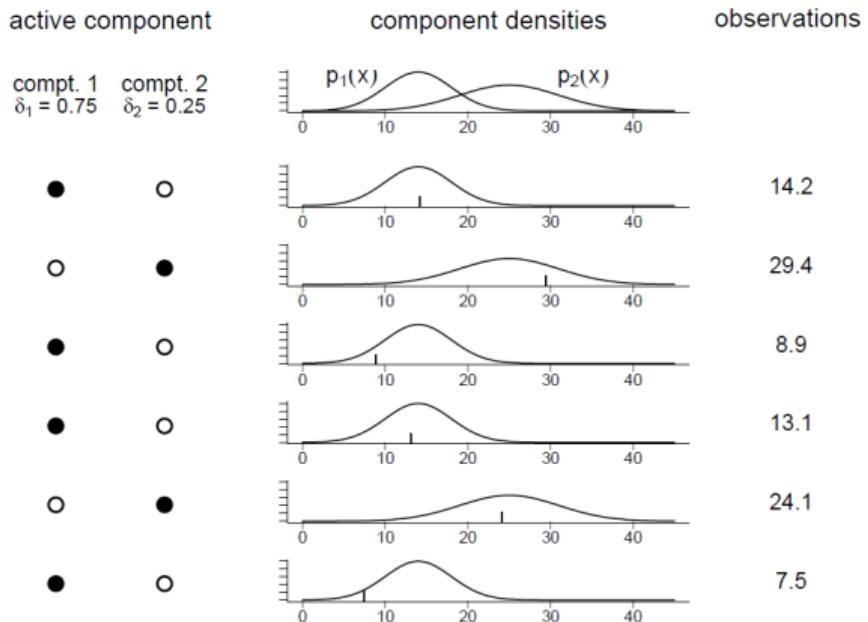
- ...the **state** (of a market, an animal, etc.) is a **latent variable** (we know that it's influencing the market or movement metrics that we observe, but we can't directly observe it).
- It often makes sense to assume that there are **finitely many** such states.
- The probability of any such state being active at time $t + 1$ should be high when the state is already active at time t .

Mixture models as building blocks of HMMs:

- In an N -state HMM, every observation will be assumed to have been generated by one of N component distributions.
- Thus, an HMM is a special type of a **mixture model**, where several component distributions are mixed together to build a new distribution.
- In the following, we will first discuss **independent** mixture models, which will later be extended to **dependent** mixture models — the latter will be HMMs.

1.1 Model formulation and dependence structure — mixture models

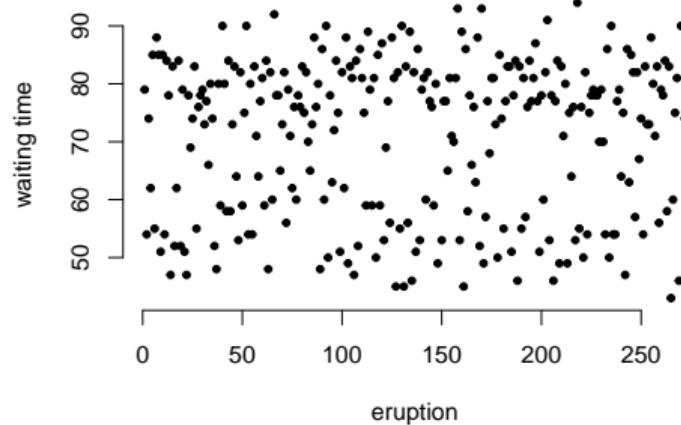
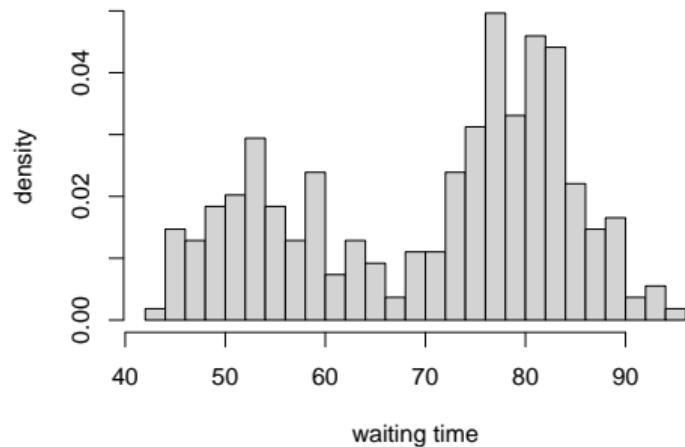
Illustration of the idea behind independent mixture models⁵:



⁵Illustration from Zucchini *et al.* (2016).

1.1 Model formulation and dependence structure

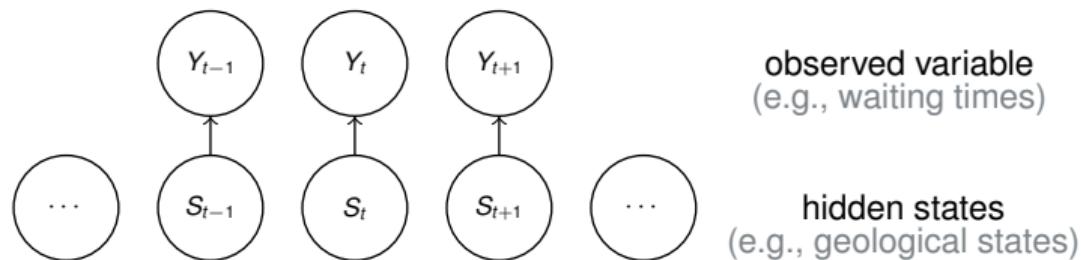
Example 4 — eruption waiting times of the Old Faithful geyser in Yellowstone National Park, USA:



Key features: short and long waiting times \rightsquigarrow state-switching over time.

1.1 Model formulation and dependence structure — mixture models

Dependence structure of an independent mixture model:



- N component distributions, which are characterised by density functions f_1, f_2, \dots, f_N .
- $S_t \in \{1, \dots, N\}$ selects which component distribution is active at time t .
- $S_t \perp S_{t^*}$ (independent) for all $t \neq t^*$.
- In particular, $\Pr(S_t = j | S_{t-1} = i) = \pi_j$ (the fact that state i is active at time $t - 1$ has no consequence on the distribution of states at time t).

1.1 Model formulation and dependence structure — mixture models

An independent mixture model for the Old Faithful geyser data:

- A 2-state Gaussian independent mixture model seems to be a natural choice:

$$Y_t \sim \begin{cases} \mathcal{N}(\mu_1, \sigma_1^2) & \text{when in state 1} \\ \mathcal{N}(\mu_2, \sigma_2^2) & \text{when in state 2,} \end{cases}$$

where $\Pr(\text{in state 1}) = \pi_1$, $\Pr(\text{in state 2}) = \pi_2$, and $\pi_1 + \pi_2 = 1$.

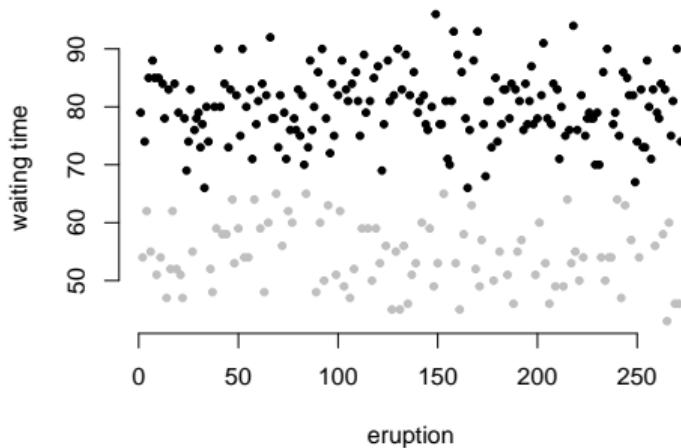
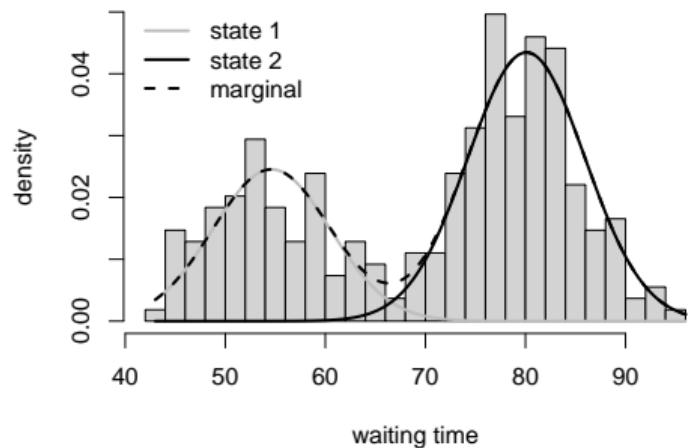
- With f_1 and f_2 denoting the densities of the normal distribution in states 1 and 2, respectively, the distribution of a random waiting time y_t is obtained as

$$\begin{aligned} f(y_t) &= \Pr(\text{in state 1}) \cdot f(y_t | \text{in state 1}) + \Pr(\text{in state 2}) \cdot f(y_t | \text{in state 2}) \\ &= \pi_1 \cdot f_1(y_t) + \pi_2 \cdot f_2(y_t). \end{aligned}$$

- $f(y_t)$ is a weighted sum of the densities f_1 and f_2 .

1.1 Model formulation and dependence structure — mixture models

A 2-state Gaussian independent mixture model fitted to the Old Faithful geyser data:



Density of the fitted mixture model:

$$\begin{aligned}f(y_t) &= \hat{\pi}_1 \cdot \hat{f}_1(y_t) + \hat{\pi}_2 \cdot \hat{f}_2(y_t) \\&= 0.361 \cdot f_{\mathcal{N}(54.6, 5.9^2)}(y_t) + 0.639 \cdot f_{\mathcal{N}(80.1, 5.9^2)}(y_t).\end{aligned}$$

1.1 Model formulation and dependence structure

Does the independence assumption for the states make sense?

- In some applications, it does, but often it doesn't (in stock market data, animal movement data, etc., there is often strong serial correlation).
- In particular, it is often the case that there is some inertia in systems (a tendency to remain in a given state).
- Thus, we'd like to have a model where the component distribution selected at time $t + 1$ is likely to be the same as the component distribution selected at time t .
- An independent mixture model doesn't account for this serial correlation in the state process.
- This naturally leads to the consideration of **Markov chains** for the state process.

1.1 Model formulation and dependence structure — Markov chains

A **Markov chain** is a sequence of random variables S_1, S_2, \dots, S_T such that

- $S_t \in \{1, \dots, N\}$ for all t (there are N **states**)
- the **Markov property** holds:

$$\Pr(S_{t+1} = s_{t+1} | S_t = s_t, \dots, S_1 = s_1) = \Pr(S_{t+1} = s_{t+1} | S_t = s_t).$$

- The Markov property means that the state at time t completely determines the probabilities of the different states at time $t + 1$ (given the present, the future is independent of the past).
- This dependence structure is mathematically convenient and often plausible.

1.1 Model formulation and dependence structure — Markov chains

- A Markov chain is characterised by specifying the **initial state distribution**,

$$\boldsymbol{\delta}^{(1)} = (\delta_1^{(1)}, \dots, \delta_N^{(1)}), \quad \text{where } \delta_i^{(1)} = \Pr(S_1 = i),$$

and the (one-step) **transition probability matrix** (t.p.m.),

$$\boldsymbol{\Gamma}^{(t)} = \begin{pmatrix} \gamma_{11}^{(t)} & \cdots & \gamma_{1N}^{(t)} \\ \vdots & \ddots & \vdots \\ \gamma_{N1}^{(t)} & \cdots & \gamma_{NN}^{(t)} \end{pmatrix}, \quad \text{where } \gamma_{ij}^{(t)} = \Pr(S_{t+1} = j \mid S_t = i).$$

- **Restrictions** on the parameters: $\gamma_{ij} \in [0, 1]$ for all i, j (the transition probabilities are actually probabilities) and $\sum_{j=1}^N \gamma_{ij} = 1$ for all i (we need to go to some state).

1.1 Model formulation and dependence structure — Markov chains

- A Markov chain is called **homogeneous** if $\gamma_{ij}^{(t)} = \gamma_{ij}$ for all t — in case of homogeneity (which we assume for basic HMMs), the t.p.m. is

$$\boldsymbol{\Gamma} = \begin{pmatrix} \gamma_{11} & \cdots & \gamma_{1N} \\ \vdots & \ddots & \vdots \\ \gamma_{N1} & \cdots & \gamma_{NN} \end{pmatrix}, \quad \gamma_{ij} = \Pr(S_{t+1} = j | S_t = i).$$

- The **stationary distribution** of a (homogeneous) Markov chain, $\boldsymbol{\delta}$, is the solution to

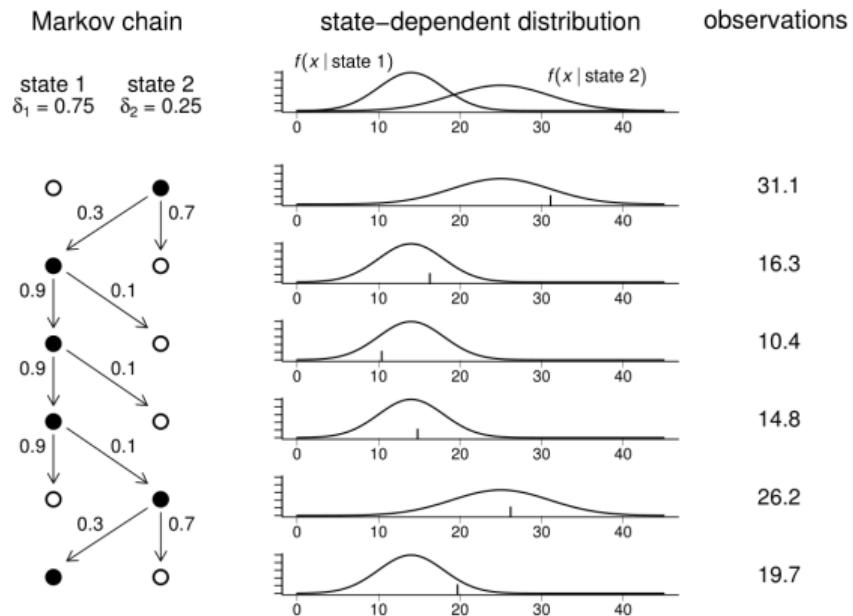
$$\boldsymbol{\delta}\boldsymbol{\Gamma} = \boldsymbol{\delta} \quad \text{subject to} \quad \sum_{i=1}^N \delta_i = 1,$$

which can be obtained by solving the equation system

$$\left(\begin{pmatrix} 1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1 \end{pmatrix} - \boldsymbol{\Gamma} + \begin{pmatrix} 1 & \cdots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \cdots & 1 \end{pmatrix}^\top \begin{pmatrix} \delta_1 \\ \vdots \\ \delta_N \end{pmatrix} \right) = \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}.$$

1.1 Model formulation and dependence structure — HMMs

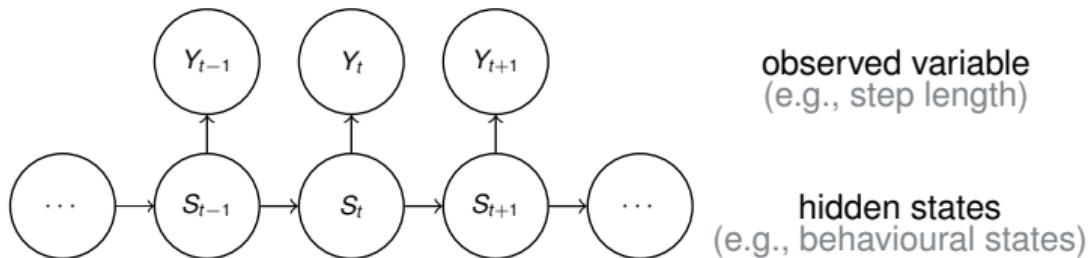
Illustration of the idea behind HMMs⁶:



⁶Illustration from Zucchini *et al.* (2016).

1.1 Model formulation and dependence structure — HMMs

Incorporating serial correlation — from (independent) mixture models to HMMs:



- N component distributions, which are characterised by density functions f_1, f_2, \dots, f_N .
- $S_t \in \{1, \dots, N\}$ selects which component distribution is active at time t .
- S_t is a **Markov chain**, such that $S_t \not\perp S_{t^*}$ (not independent) for all $t \neq t^*$.
- In particular, $\Pr(S_t = j | S_{t-1} = i) = \gamma_{ij}$ (the fact that state i is active at time $t - 1$ does have a consequence on the distribution of states at time t).

1.1 Model formulation and dependence structure — HMMs

An N -state **HMM** is a doubly stochastic process in discrete time, with

- an unobserved **state process** S_1, S_2, \dots, S_T taking values in $\{1, \dots, N\}$,
- an observed **state-dependent process** Y_1, Y_2, \dots, Y_T ,

such that

- $f(s_t | s_1, \dots, s_{t-1}) = f(s_t | s_{t-1})$ (Markov property)
- $f(y_t | s_1, \dots, s_t, y_1, \dots, y_{t-1}) = f(y_t | s_t)$ (conditional independence assumption).

- This is the most basic HMM — various extensions of this dependence structure are possible (more on these later in this session).
- To get some intuition about Markov chains and the data-generating process assumed, let's simulate from an HMM!

1.1 Model formulation and dependence structure in R

Example in R — simulating from an HMM:



https://github.com/timoadam/mtsa/blob/main/1_Introduction_to_HMMs.R.

Outline

1 Introduction to HMMs (13:30–15:00)

1.1 Model formulation and dependence structure

1.2 Likelihood evaluation and model fitting

1.3 Model selection, model checking, and state decoding

■ Extensions of the basic HMM (15:30–16:45)

2.1 Multivariate HMMs

2.2 Covariate-dependent state processes

■ Summary, questions, etc. (16:45–17:00)

1.2 Likelihood evaluation and model fitting

To fit HMMs to data (i.e., estimating its parameters), we'll use **maximum likelihood** (ML) estimation:

- Idea: good parameter estimates make the observed data look plausible (select those parameters for which the model has the highest likelihood of having generated the observed data).
- Given observations y_1, \dots, y_T , we regard the joint density of all observations as a function of the parameter vector of interest, θ ,

$$\mathcal{L}(\theta) = f_\theta(y_1, \dots, y_T).$$

- The ML **estimate** (MLE) is the parameter vector θ that maximises the likelihood function $\mathcal{L}(\theta)$.
- Since the logarithm is strictly monotone⁷, we have that

$$\theta \text{ maximises } \mathcal{L}(\theta) \iff \theta \text{ maximises } \ell(\theta) = \log \mathcal{L}(\theta).$$

- Maximising $\ell(\theta)$ is often easier in practice, and is in fact necessary for HMMs due to numerical underflow (see later slides).

⁷That is, $z < z^* \Leftrightarrow \log(z) < \log(z^*)$.

1.2 Likelihood evaluation and model fitting

Naive (brute force) approach to evaluate the likelihood:

- Separate consideration of all possible state sequences that may have given rise to the observations (for example, with $T = 500$ and $N = 2$, there are $N^T = 500^2 = 250,000$ possible state sequences).

Smart approach to evaluate the likelihood:

- Exploit the dependence structure of HMMs, step-by-step traversing along the time series and updating the likelihood at every step ( forward algorithm).
- We'll first discuss the (completely useless) naive approach...
- ...but only to emphasise how very neat the forward algorithm is!

1.2 Likelihood evaluation and model fitting — naive approach

Naive approach to calculating the likelihood:

$$\mathcal{L}(\theta) = f_{\theta}(y_1, \dots, y_T)$$

$$= \sum_{s_1=1}^N \dots \sum_{s_T=1}^N f(y_1, \dots, y_T, s_1, \dots, s_T) \quad (\text{total probability})$$

$$= \sum_{s_1=1}^N \dots \sum_{s_T=1}^N f(y_1, \dots, y_T | s_1, \dots, s_T) \cdot f(s_1, \dots, s_T) \quad (\text{conditional probability})$$

$$= \sum_{s_1=1}^N \dots \sum_{s_T=1}^N \left(\prod_{t=1}^T f(y_t | s_t) \right) \left(f(s_1) \prod_{t=2}^T f(s_t | s_{t-1}) \right) \quad (\text{dependence assumptions})$$

$$= \sum_{s_1=1}^N \dots \sum_{s_T=1}^N \left(\prod_{t=1}^T f_{s_t}(y_t) \right) \left(\delta_{s_1} \prod_{t=2}^T \gamma_{s_{t-1}, s_t} \right).$$

Simple structure, and all components are directly available — but N^T summands render the calculation infeasible in most cases.

1.2 Likelihood evaluation and model fitting — naive approach

Naive approach — **wrong** conclusions (Albert, 1991, and Morales *et al.*, 2004):

We need an alternative to maximising directly since the amount of computation required to evaluate this likelihood becomes enormous for even moderately sized $[T]$ ⁸.

As the number of observations and behavioural states increases, it becomes infeasible to evaluate all possible forms [of the state vector]⁹.

- Even in more recent literature, lots of similar statements can be found.
- These often serve as a motivation for using either expectation-maximisation (EM)-based or Markov chain Monte Carlo (MCMC)-based estimation approaches, which tend to be complicated and/or slow.
- However, the above statements are simply wrong!

⁸Albert, P.S. (1991): A two-state Markov mixture model for a time series of epileptic seizure counts. *Biometrics*, 47(4), 1371–1381.

⁹Morales, J.M., Haydon, D.T., Frair, J., Holsinger, K.E., and Fryxell, J.M. (2004): Extracting more out of relocation data: building movement models as mixtures of random walks. *Ecology*, 85(9), 2436–2445.

1.2 Likelihood evaluation and model fitting — forward algorithm

Forward algorithm — motivation:

- The problem with the naive approach is that many calculations are redundant.
- For example, for $T = 20$ and $N = 2$, separately calculating the likelihood contribution of the two state sequences

1 1 1 2 2 2 2 1 1 1 1 1 1 2 2 2 2 1 1 1
and 1 1 1 2 2 2 2 1 1 1 1 1 1 2 2 2 2 1 1 2

is extremely inefficient.

- Intuitively, what we'd like to do instead is the following: calculate likelihood contribution of the states at the first 19 occasions **only once** — then **fork off** into the two possible states at time $t = 20$.
- This is essentially what the forward algorithm does.

1.2 Likelihood evaluation and model fitting — forward algorithm

Consider the so-called **forward variables**,

$$\alpha_t = (\alpha_t(1), \dots, \alpha_t(N)), \quad \text{where} \quad \alpha_t(j) = f(y_1, \dots, y_t, s_t = j).$$

At time t , the forward variables contain information on

- the likelihood of the observations up to time t
- the probabilities of being in the different states at time t .

The **forward algorithm** is an efficient recursive scheme for calculating the forward variables:

$$\begin{aligned}\alpha_1 &= \delta^{(1)} \mathbf{P}(y_1) \\ \alpha_t &= \alpha_{t-1} \boldsymbol{\Gamma} \mathbf{P}(y_t) \quad \text{for } t = 2, \dots, T,\end{aligned}$$

with $\mathbf{P}(y_t) = \text{diag}(f_1(y_t), \dots, f_N(y_t))$, initial distribution $\delta^{(1)}$, and t.p.m. $\boldsymbol{\Gamma}$.

1.2 Likelihood evaluation and model fitting — forward algorithm

Proof of the recursion (in class):

1.2 Likelihood evaluation and model fitting — forward algorithm

Proof of the recursion (in class):

1.2 Likelihood evaluation and model fitting — forward algorithm

- The forward algorithm can be applied to first calculate α_1 , then α_2 based on α_1 , then α_3 based on α_2 , etc., until we arrive at α_T .
- The **likelihood** can then be obtained as $\mathcal{L}(\theta) = \sum_{j=1}^N f(y_1, \dots, y_T, s_T = j) = \sum_{j=1}^N \alpha_T(j)$ or, in closed-form matrix expression with $\mathbf{1} = (1, \dots, 1) \in \mathbb{R}^N$, as

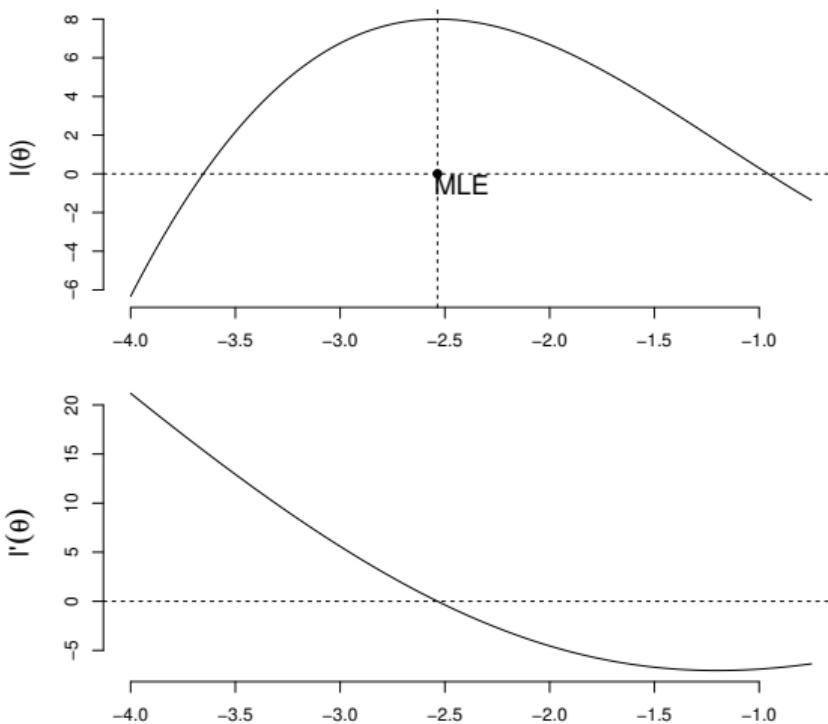
$$\mathcal{L}(\theta) = \delta^{(1)} \mathbf{P}(y_1) \boldsymbol{\Gamma} \mathbf{P}(y_2) \boldsymbol{\Gamma} \mathbf{P}(y_3) \dots \boldsymbol{\Gamma} \mathbf{P}(y_T) \mathbf{1}^\top.$$

- Computational cost of evaluating $\mathcal{L}(\theta)$ is **linear** in the number of observations T .
- In practice, this means that the likelihood can be evaluated in a fraction of a second even for T in the thousands and a moderate number of states N .
- This opens up the way for a **numerical maximisation** of the likelihood.

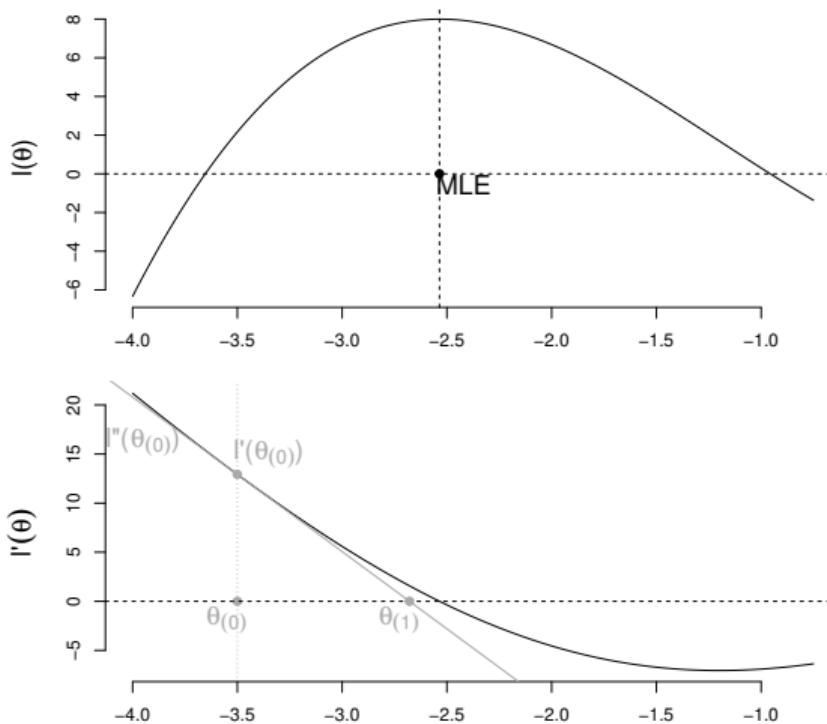
1.2 Likelihood evaluation and model fitting

- Like for most other statistical models, there is (in general) no closed-form (analytical) solution for the MLE of an HMM.
- Instead, **numerical algorithms** are used to find the MLE — general workflow:
 - Guess the value of the parameter vector as $\theta_{(0)}$ (initial value).
 - Obtain improved guess $\theta_{(1)}$ based on $\theta_{(0)}$.
 - Obtain improved guess $\theta_{(2)}$ based on $\theta_{(1)}$.
 - ...
 - Terminate algorithm when changes in $\ell(\theta)$ are negligible.
- On the next slides, we illustrate the **Newton-Raphson method**, which is a very popular method for likelihood maximisation in HMMs.

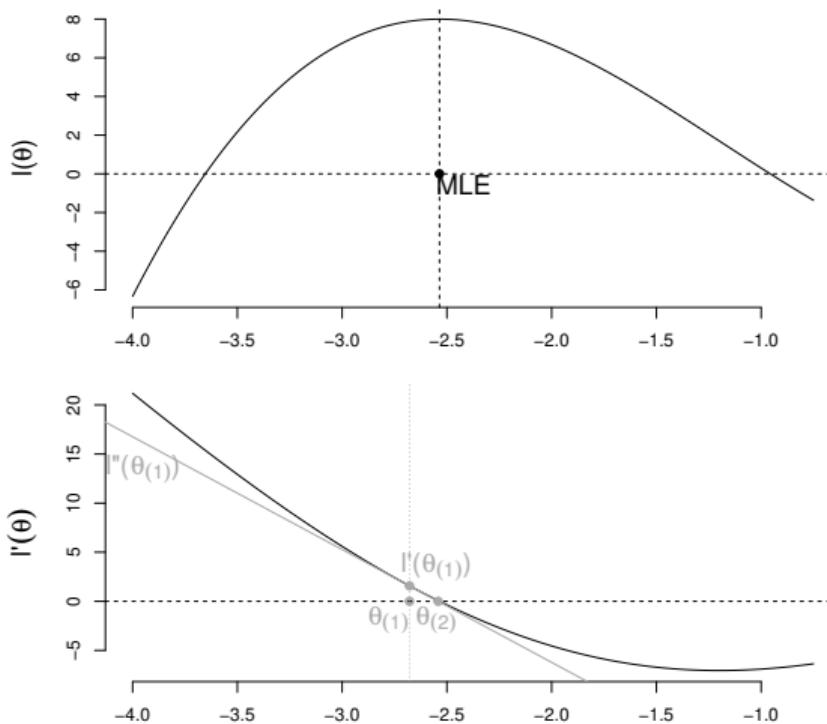
1.2 Likelihood evaluation and model fitting



1.2 Likelihood evaluation and model fitting



1.2 Likelihood evaluation and model fitting



1.2 Likelihood evaluation and model fitting

Numerical maximisation of the (log-) likelihood:

- Newton-Raphson is one of various numerical optimisation schemes that can be used to find the MLE of an HMM.
- In R, the routines `nlm()`, `optim()`, `optimx()`, `nloptr()`, and `nlminb()` provide several options (several optimisers that can be used).
- The different optimisers all have their strengths and weaknesses, and the performance depends on the problem at hand (Nash, 2014¹⁰).

¹⁰Nash, J.C. (2014): On best practice optimization methods in R. *Journal of Statistical Software*, 60(2). 1–14.

1.2 Likelihood evaluation and model fitting

Technical issues related to the maximisation:

- Numerical **underflow (overflow)**: for large T , the likelihood becomes very small (or very large) and is rounded to 0 (or returned as Inf) in R.
- **Local maxima** of the likelihood: the algorithm may converge to a local rather than the global maximum of the likelihood.
- Parameter **constraints**: the transition probabilities (and possibly other parameters) are constrained parameters.

We won't discuss these in detail (not enough time) — the LaMa package takes care of numerical under- and overflow by computing the log-likelihood.

1.2 Likelihood evaluation and model fitting — parameter constraints

Option 1 — constrained optimisers:

- Use a numerical maximisation routine which allows to **specify constraints** (e.g., `nlsminb()` in R).

Option 2 — parameter transformations:

- Consider one-to-one transformation $g(\theta) = \theta^*$ such that θ^* is **unconstrained**.
- Inverse transformation: $\theta = g^{-1}(\theta^*)$.
- Maximise the log-likelihood $\ell(\theta) = \ell(g^{-1}(\theta^*))$ with respect to θ^* .
- This is equivalent to maximising $\ell(\theta)$ with respect to θ (since by searching in the parameter space of θ^* we effectively search in the entire parameter space of θ).
- Unconstrained optimisers like `nls()` in R can then be used.

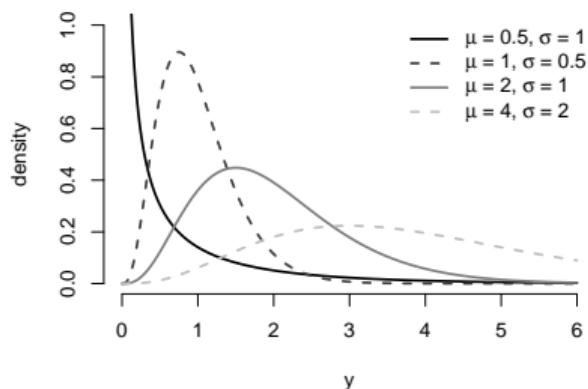
We'll use option 2 — our experience with option 1 hasn't been that positive!

1.2 Model fitting in R

- A random variable $Y > 0$ follows a **gamma distribution** if its density is given by

$$f(y) = \frac{\beta^\alpha \cdot y^{\alpha-1} \cdot \exp(-\beta \cdot y)}{\Gamma(\alpha)},$$

where α (scale) and β (shape) > 0 .



- If $Y \sim \text{gamma}(\alpha, \beta)$, then $\mu = E(Y) = \alpha \cdot \beta$ and $\sigma^2 = \text{Var}(Y) = \alpha \cdot \beta^2$.

1.2 Model fitting in R

Example in R — fitting an HMM to the elephant data:



https://github.com/timoadam/mtsa/blob/main/1_Introduction_to_HMMs.R.

Outline

1 Introduction to HMMs (13:30–15:00)

1.1 Model formulation and dependence structure

1.2 Likelihood evaluation and model fitting

1.3 Model selection, model checking, and state decoding

■ Extensions of the basic HMM (15:30–16:45)

2.1 Multivariate HMMs

2.2 Covariate-dependent state processes

■ Summary, questions, etc. (16:45–17:00)

1.3 Model selection and model checking

We now know how to fit an HMM to data. But how do we find the “right” HMM? In order to do so, we need to be able to...

1. ...select the best HMM from a set of candidate models (e.g., 2-state vs. 3-state or normal vs. Student’s t-component distributions) ↵ **model selection**
2. ...decide for a selected model whether it is a good model or not (analogous for example to a residual analysis in a linear model) ↵ **model checking**.

But is any real process actually an HMM?

- In practice we consider an HMM to be no more than an approximate representation of a real data-generating process ↵ “*All models are wrong...*” (George Box).
- “[...] *There is no need to ask the question ‘Is the model true?’.* [...] *The only question of interest is ‘is the model illuminating and useful?’*” (George Box).

1.3 Model selection and model checking

In practice, the search for the “right” HMM is usually an **iterative process**:

1. Formulate one or more candidate models.
 2. Fit candidate models to the data.
 3. Select among the candidate models.
 4. Check if the selected model fits the data well.
 5. Perhaps the fit is not satisfactory ↵ go back to 1. and formulate new candidate models that try to address the problems identified in 4..
 6. Select among the new candidate models.
- ...
123. At last, identification of a model that fits the data well — or maybe give up trying to model the data using HMMs...

1.3 Model selection

Building a set of candidate models most importantly involves choosing the **number of states**¹¹ and the component **distributions** — the latter are typically determined by the type of data:

- Positive continuous data (e.g., step lengths) \rightsquigarrow exponential or gamma distribution.
- Other continuous data (e.g., stock returns) \rightsquigarrow normal or Student's t-distribution.
- Unbounded counts (e.g., earthquake counts) \rightsquigarrow Poisson or negative binomial.
- Bounded counts (e.g., experimental data) \rightsquigarrow binomial distribution.
- Binary data (e.g., event data) \rightsquigarrow Bernoulli distribution.
- Categorical data (e.g., survey answers) \rightsquigarrow multinomial distribution.
- ...

¹¹Unless of course the number of states is known or simply fixed a priori.

1.3 Model selection

Akaike information criterion (AIC) — select the model that is closest to the true model (the least bad model):

- Estimate discrepancies between true model and candidate models.
- Select the model with the **smallest** (estimated) **discrepancy**.

Bayesian information criterion (BIC) — select the model that is most likely to be the true one:

- Quantify prior beliefs, then calculate posterior probability of each candidate model (given the data).
- Select the model with the **highest posterior probability**.

Cross-validation (won't be covered in detail):

- Split sample into calibration and validation subsamples, fit candidate models to calibration data, then evaluate using validation data.

1.3 Model selection — information criteria

- Model selection: trade-off between flexibility and parsimony/stability.

$$\text{AIC} = -2 \cdot \log \mathcal{L}(\theta) + 2 \cdot \text{number of parameters}$$

$$\text{BIC} = -2 \cdot \log \mathcal{L}(\theta) + \log(T) \cdot \text{number of parameters}$$

- In either case, choose model with **smallest** value of the criterion applied.
- Both criteria **reward goodness of fit** while **penalising complexity**.
- AIC favours more complex models due to the smaller penalty (for $T \geq 8$).
- Both provide a relative comparison of models from a suite of candidate models \rightsquigarrow the selected model may still be a bad one!

1.3 Model selection — information criteria

For something as complex as animal movement, models will always be simplistic¹². Misspecification of the model is often compensated by additional states, which “mop up” the neglected structure caused by

- inadequate (parametric) state-dependent distributions
- outliers
- temporal variation
- individual heterogeneity
- violations of Markov property or conditional independence assumption
- ...

For HMMs, both AIC and BIC **tend to favour models with “too many” states** ↵ don't blindly trust these criteria, and instead only use them as guidance.

¹²This is not a problem per se, since the aim of statistical modelling usually is to capture only the key patterns in real data.

1.3 Model checking

Main options to check if a fitted HMM is adequate:

- **Graphical comparison** of marginal distribution under the fitted HMM and the empirical distribution of the data, to check adequacy of state-dependent distributions.
- **Simulate data from the fitted model**, then compare the patterns found in the simulated data with those of the real data¹³ — informal but useful strategy!
- **Pseudo-residual check** — comprehensive formal check of the model.

¹³Patterns to look for: marginal distribution, autocorrelation, etc.

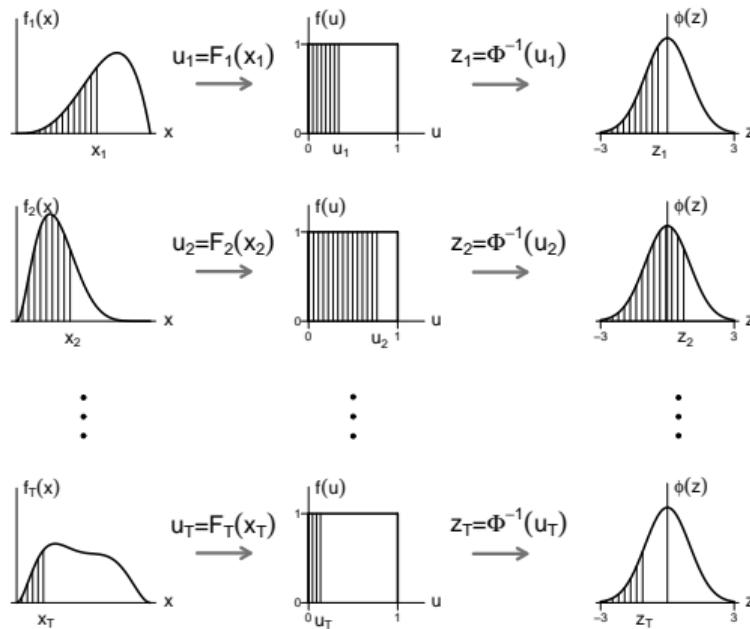
1.3 Model checking — pseudo-residuals

Pseudo-residuals:

- Problem: each Y_t has a different distribution, making it difficult to assess which observations are extreme **relative to the model**.
- Solution: consider the so-called pseudo-residuals — if the fitted model is correct, then these follow a standard normal distribution.
- We can use quantile-quantile-plots and/or normality tests to check the adequacy of the state-dependent distributions: any indication of non-normality \rightsquigarrow indication of a lack of fit.
- Strong residual autocorrelation \rightsquigarrow correlation structure not fully captured.

1.3 Model checking — pseudo-residuals

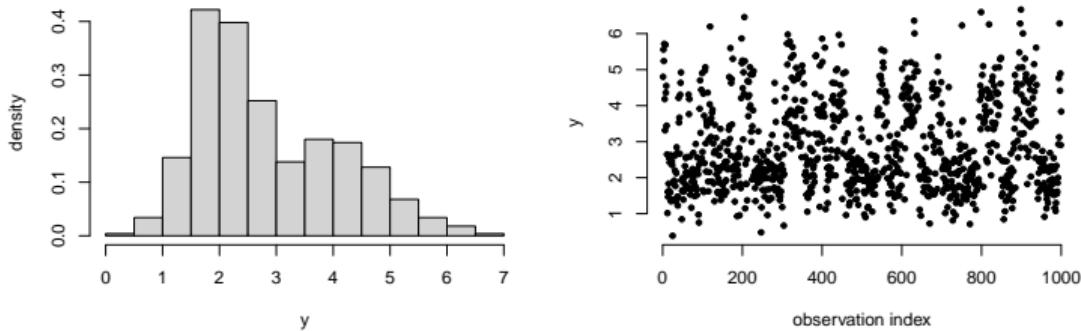
Construction of pseudo-residuals¹⁴:



¹⁴ Illustration from Zucchini *et al.* (2016).

1.3 Model checking — pseudo-residuals

Pseudo-residual check for HMMs — illustration using simulated data (a 2-state normal HMM was used to generate these data):

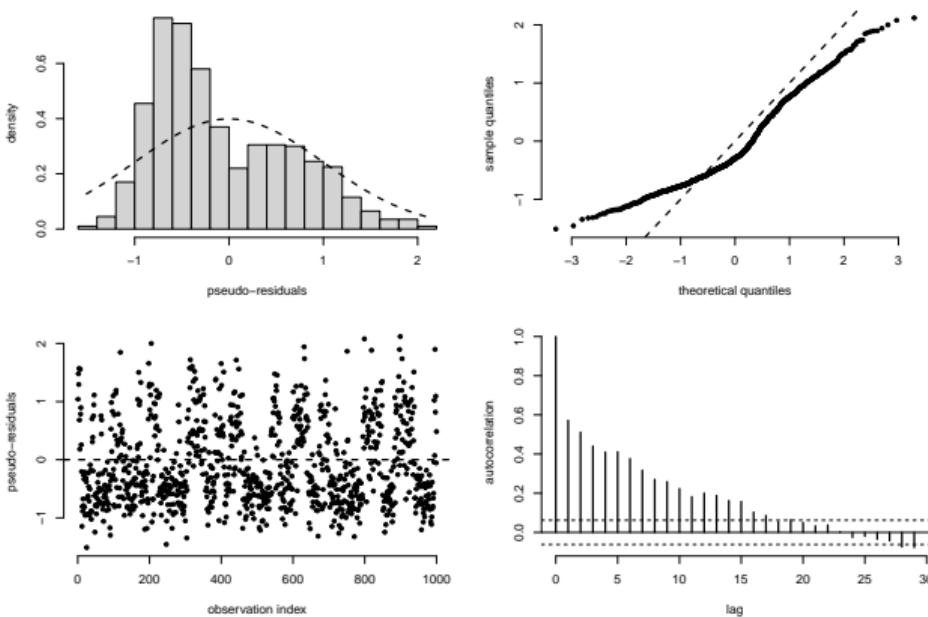


On the next slides, pseudo-residuals will be shown as obtained when fitting

- a simple normal distribution (the observations are assumed to be i.i.d. normally distributed)
- a 2-state Gaussian independent mixture model (serial correlation not modelled)
- a 2-state normal HMM (the correct model).

1.3 Model checking — pseudo-residuals

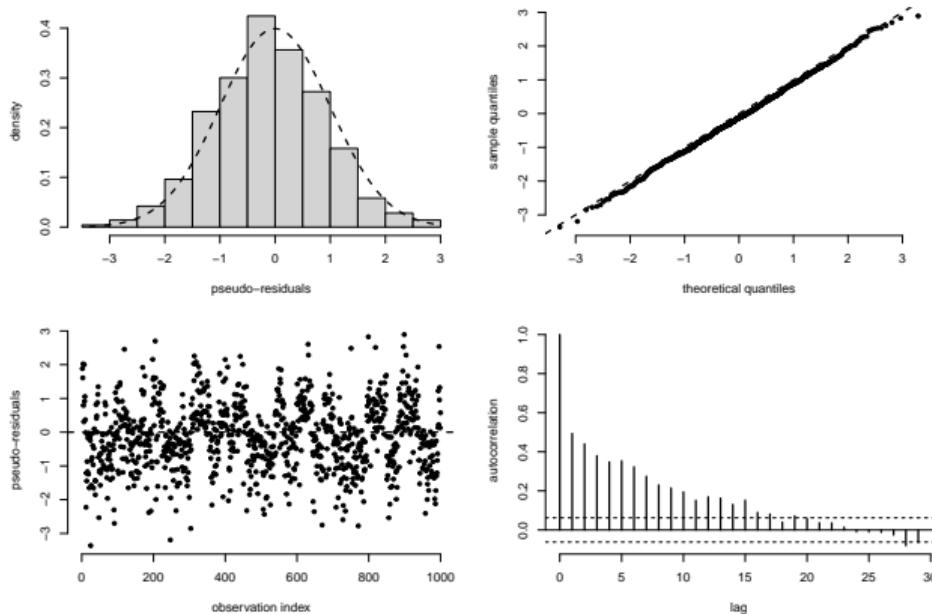
Pseudo-residuals obtained under the fitted **simple normal distribution**:



~ The marginal distribution and the correlation structure are not captured well.

1.3 Model checking — pseudo-residuals

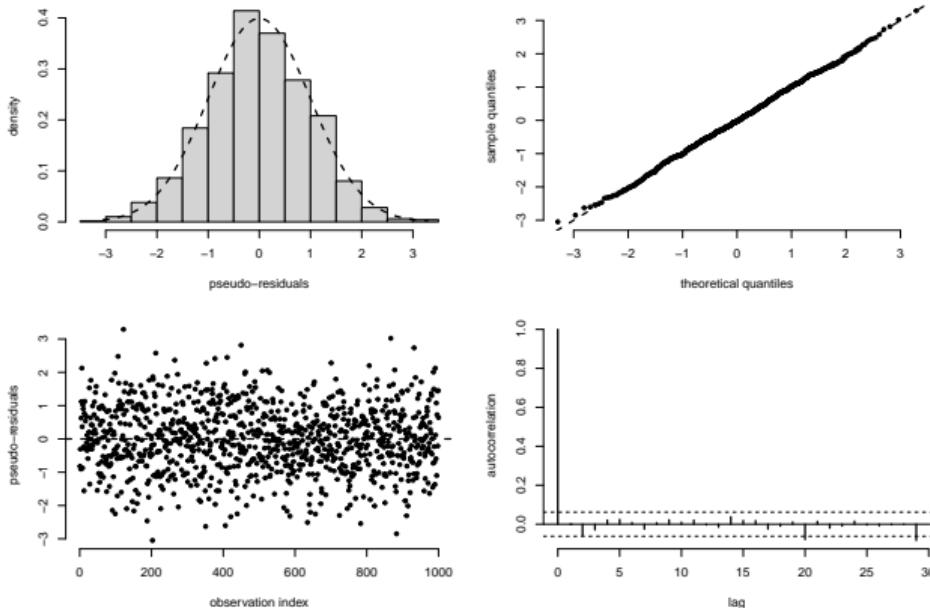
Pseudo-residuals obtained under the fitted 2-state Gaussian independent **mixture model**:



~ The marginal distribution is captured well — but not the dependence structure.

1.3 Model checking — pseudo-residuals

Pseudo-residuals obtained under the fitted 2-state **normal HMM**:



Both the marginal distribution and the correlation structure are captured well.

1.3 State decoding

Given a fitted model, it is often of interest to decode the hidden states underlying the observations — this can be done using local and global decoding. **Local decoding:**

- Consider $\Pr(S_t = i | y_1, \dots, y_T)$ — the most probable state at time t is the maximum of the above over $i = 1, \dots, N$ (looks at each time point in isolation).

Global decoding:

- Consider $\Pr(S_1 = i_1, \dots, S_T = i_T | y_1, \dots, y_T)$ — the most probable state sequence is the maximum of the above over $(i_1, \dots, i_T) \in \{1, \dots, N\}^T$ (looks at the sequence as a whole).
- This can't be done via brute force (considering all possible state sequences — there are too many).
- Instead, the so-called **Viterbi algorithm** is used — this again involves the application of a recursive algorithm.

1.3 State decoding

Local vs. global decoding:

- Usually the outcome is either identical or at least very similar.
- Most people tend to use Viterbi rather than local decoding, simply because it usually makes sense to look at the state sequence as a whole.
- However, local decoding comes with one crucial advantage, namely an uncertainty quantification in the decoded state sequence.
- This is relevant for asymmetric costs of incorrectly decoding a state: suppose the HMM is a model for the disease progression of a patient...
- ...and doctors intervene when the HMM indicates a worsening disease status.
- If local decoding then says 60 % “stable”, 40 % “deteriorating”, then surely we shouldn’t rely on the patient being in the most likely (stable) state!

1.3 Model selection, model checking, and state decoding in R

Example in R — model selection, model checking, and state decoding for the elephant data:



https://github.com/timoadam/mtsa/blob/main/1_Introduction_to_HMMs.R.

Outline

1 Introduction to HMMs (13:30–15:00)

- 1.1 Model formulation and dependence structure
- 1.2 Likelihood evaluation and model fitting
- 1.3 Model selection, model checking, and state decoding

2 Extensions of the basic HMM (15:30–16:45)

- 2.1 Multivariate HMMs
- 2.2 Covariate-dependent state processes

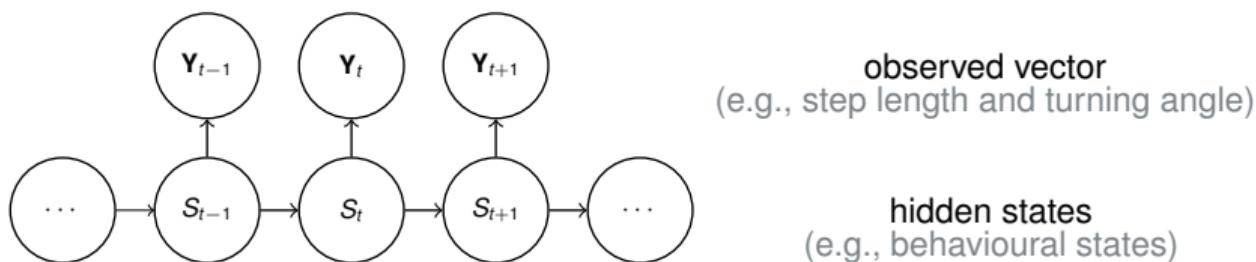
3 Summary, questions, etc. (16:45–17:00)

2.1 Multivariate HMMs

- HMMs are often used to model **multivariate** time series, where the observations are **vectors** $\mathbf{Y}_1, \mathbf{Y}_2, \dots, \mathbf{Y}_T$, where

$$\mathbf{Y}_1 = \begin{pmatrix} Y_{11} \\ Y_{12} \\ \vdots \\ Y_{1K} \end{pmatrix}, \mathbf{Y}_2 = \begin{pmatrix} Y_{21} \\ Y_{22} \\ \vdots \\ Y_{2K} \end{pmatrix}, \dots, \mathbf{Y}_T = \begin{pmatrix} Y_{T1} \\ Y_{T2} \\ \vdots \\ Y_{TK} \end{pmatrix}.$$

- In many such scenarios it makes sense to assume these K variables to be **driven by the same underlying state process** S_t :



- Given conditional independence and the Markov property, the **likelihood** form remains **unchanged**.

2.1 Multivariate HMMs — dependence assumptions

For multivariate HMMs, there are two important dependence assumptions:

Longitudinal conditional independence: conditional on the states, the vectors $\mathbf{Y}_1, \dots, \mathbf{Y}_T$ are mutually independent \rightsquigarrow we have to select a class of multivariate distributions, with some density $f(y_{t1}, \dots, y_{tK})^{15}$.

Contemporaneous conditional independence: conditional on the states, all components X_{tk} , $t = 1, \dots, T$, $k = 1, \dots, K$ are independent. Under that assumption, the multivariate state-dependent distribution is a product of K univariate distributions:

$$f(y_{t1}, \dots, y_{tK} | S_t = j) = f(y_{t1} | S_t = j) \cdot \dots \cdot f(y_{tK} | S_t = j)$$

\rightsquigarrow we have to select classes of univariate distributions, with densities $f(y_{t1}), \dots, f(y_{tK})$.

¹⁵However, at any time t , the components of the vector \mathbf{Y}_t may be correlated.

2.1 Multivariate HMMs — dependence assumptions

- A possible model that assumes **longitudinal conditional independence**:

$$\mathbf{Y}_t | S_t = j \sim \mathcal{N}_2(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)$$

(a multivariate normal distribution models the correlation of the components of the vector \mathbf{Y}_t).

- A possible model that assumes **contemporaneous conditional independence**:

$$Y_{tk} | S_t = j \sim \mathcal{N}(\mu_{jk}, \sigma_{jk}^2), \quad Y_{t1} \perp Y_{t2} | S_t$$

(a nested, less flexible special case of the model above where the components of the vector \mathbf{Y}_t are uncorrelated).

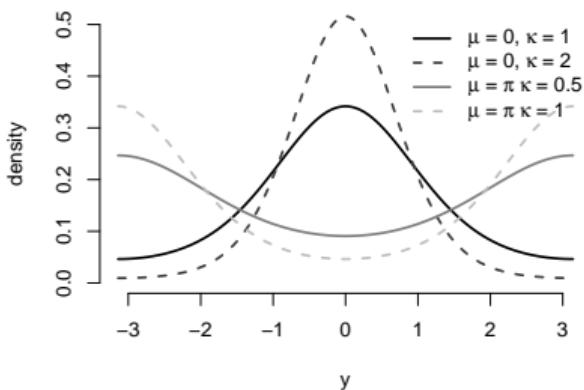
- Models involving the **stronger** assumption of contemporaneous conditional independence are easier to formulate and implement (formulating a joint distribution can be hard).
- Crucially, even under contemporaneous conditional independence, the variables are conditionally independent, given the states — but not independent: **the Markov chain induces serial dependence.**

2.1 Multivariate HMMs in R

- A random variable $Y \in [-\pi, \pi]$ follows a **von Mises distribution** if its density is given by

$$f(y) = \frac{1}{2 \cdot \pi \cdot I_0(\kappa)} \cdot \exp(\kappa \cdot \cos(y - \mu)),$$

where I_0 is the modified Bessel function of the first kind of order 0, μ (mean) $\in [-\pi, \pi]$, and κ (concentration) > 0 .



- Key properties: the von Mises distribution has support $[-\pi, \pi]$ — it “lives on the circle” — and the density satisfies the desirable property $f(-\pi) = f(\pi)$ for any μ and κ .

2.1 Multivariate HMMs in R

Example in R — modelling elephant steps and turns using multivariate HMMs:



https://github.com/timoadam/mtsa/blob/main/2_Extensions_of_the_basic_HMM.R.

2.1 Multivariate HMMs — blue whale example

Another example from the literature (DeRuiter *et al.*, 2017¹⁶):

On behalf of Springer Nature
2017, Vol 11, No. 1, 362–392
DOI 10.1007/s10463-017-1020-2
© Institute of Mathematical Statistics, 2017

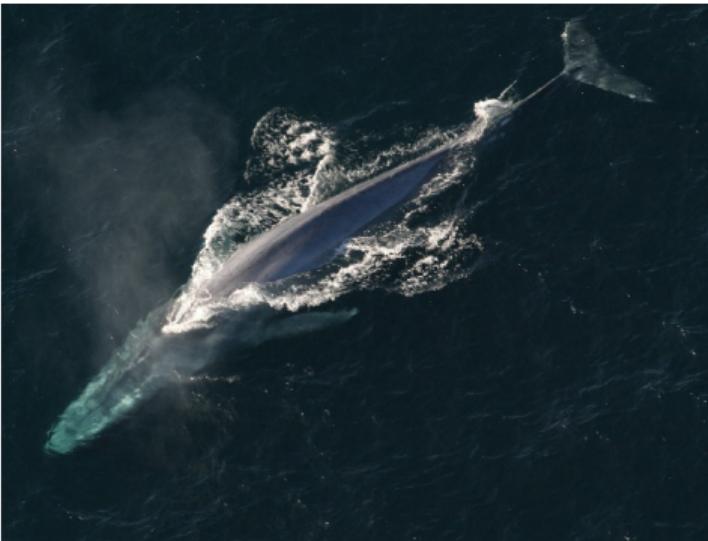
A MULTIVARIATE MIXED HIDDEN MARKOV MODEL FOR BLUE WHALE BEHAVIOUR AND RESPONSES TO SOUND EXPOSURE¹

BY STACY L. DERUITER^{2,3}, ROLAND LANGROCK^{1,3}, TOMAS SKIRBUTAS⁴,
JEREMY A. GOLDBOGEN⁵, JOHN CALAMBOKIDIS⁶,
ARI S. FRIEDELAENDER^{1,*} AND BRANDON L. SOUTHALL⁷

Gulf of California², University of St Andrews³, Bielefeld University¹,
Stanford University⁶, Cascadia Research Collective⁵,
Oregon State University⁷ and SEA, Inc.⁴

Characterization of multivariate time series of behaviour data from animal-borne sensors is challenging. Biologists require methods to objectively quantify baseline behaviour, and then assess behaviour changes in response to environmental stimuli. Here, we apply hidden Markov models (HMMs) to analyse blue whale behaviour. We identify three states underlying latent states corresponding to three main underlying behaviour states: shallow feeding, travelling, and deep feeding. The model formulation accounts for inter-whale differences via a computationally efficient cluster membership approach. We also account for individual effects on the dependence on between-state transition probabilities. We identify clear differences in blue whale disturbance response depending on the behavioural context during exposure, with whales in travel or deep feeding behaviour showing greater avoidance compared with earlier stages of shallow feeding. In addition, the HMM approach provides a more nuanced characterization of behaviour changes.

1. Introduction. Measuring and describing behaviour changes by marine mammals in response to disturbance is a key step in understanding the potential health effects of the types of human disturbance on marine mammals [Ellison *et al.* (2012), Sharma *et al.* (2015)]. The effects of mid-frequency military sonars, airguns used for geological exploration, and shipping noise are of particular concern [National Research Council (2005), Southall *et al.* (2007)]. We focus on potential behaviour changes in response to military mid-frequency active sonar sounds (MFAS, broadly defined) and pseudo-random noise (PRN) in the same frequency range. Here, mid-frequency is broadly 1–10 kHz, but the narrower frequency band containing the primary energy of the MFAS and PRN stimuli is between 3–4.5 kHz. Previous work has suggested that blue whales respond simi-



Received February 2015; revised December 2016.

¹Sponsored by the U.S. Office of Naval Research Grant N00014-12-1-0204, under the project entitled Multi-scale Ocean acoustic Human effects Analysis (MOCHA).

Key words and phrases. Forward algorithm, hidden Markov model, multivariate time series, numerical maximum likelihood, random effects, blue whales.

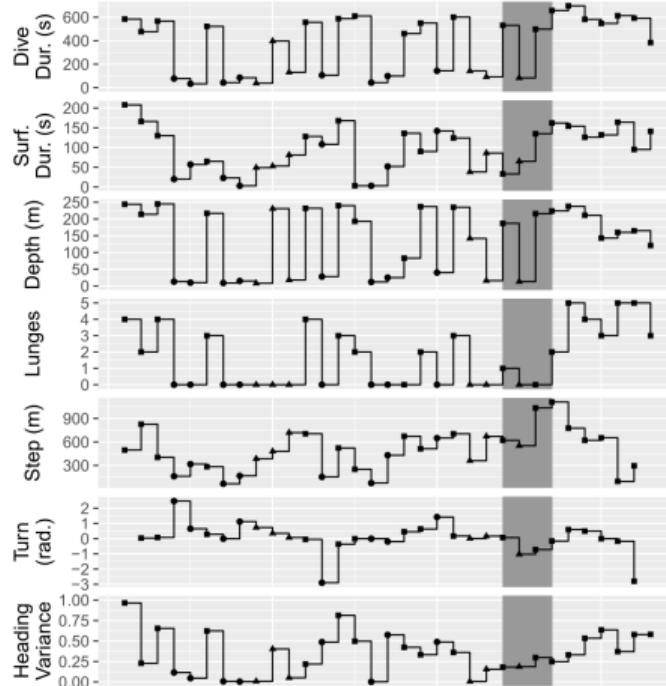
362

¹⁶ DeRuiter, S.L., Langrock, R., Skirbutas, T., Goldbogen, J.A., Calambokidis, J., Friedlaender, A.S., and Southall, B.L. (2017): A multivariate mixed hidden Markov model for blue whale behaviour and responses to sound exposure. *The Annals of Applied Statistics*, 11(1), 362–392.

2.1 Multivariate HMMs — blue whale example

Multivariate series of variables observed on a **dive-by-dive** basis — variables measured include

- dive duration $\in (0, \infty)$
- surface duration $\in (0, \infty)$
- maximum depth $\in (0, \infty)$
- number of lunges $\in \{0, 1, 2, \dots\}$
- step length $\in [0, \infty)$
- turning angle $\in [-\pi, \pi)$
- variability of whale heading $\in [0, 1]$.



2.1 Multivariate HMMs — blue whale example

- Assuming (only) longitudinal conditional independence — and hence using seven-dimensional state-dependent distribution — is infeasible here (it's just too complicated).
- Instead, we assume **contemporaneous conditional independence**,

$$f(y_{t1}, \dots, y_{t7} | S_t = j) = f(y_{t1} | S_t = j) \cdot \dots \cdot f(y_{t7} | S_t = j),$$

such that each variable has its own (state-specific) univariate distribution.

- $Y_{t1} | S_t = j \sim \text{gamma}(\mu_j^{\text{dive duration}}, \sigma_j^{\text{dive duration}})$
- $Y_{t2} | S_t = j \sim \text{gamma}(\mu_j^{\text{surface duration}}, \sigma_j^{\text{surface duration}})$
- $Y_{t3} | S_t = j \sim \text{gamma}(\mu_j^{\text{depth}}, \sigma_j^{\text{depth}})$
- $Y_{t4} | S_t = j \sim \text{Poisson}(\lambda_j)$
- $Y_{t5} | S_t = j \sim \text{gamma}(\mu_j^{\text{step length}}, \sigma_j^{\text{step length}})$
- $Y_{t6} | S_t = j \sim \text{von Mises}(\mu_j^{\text{turning angle}}, \kappa_j)$
- $Y_{t7} | S_t = j \sim \text{beta}(a_j, b_j)$.

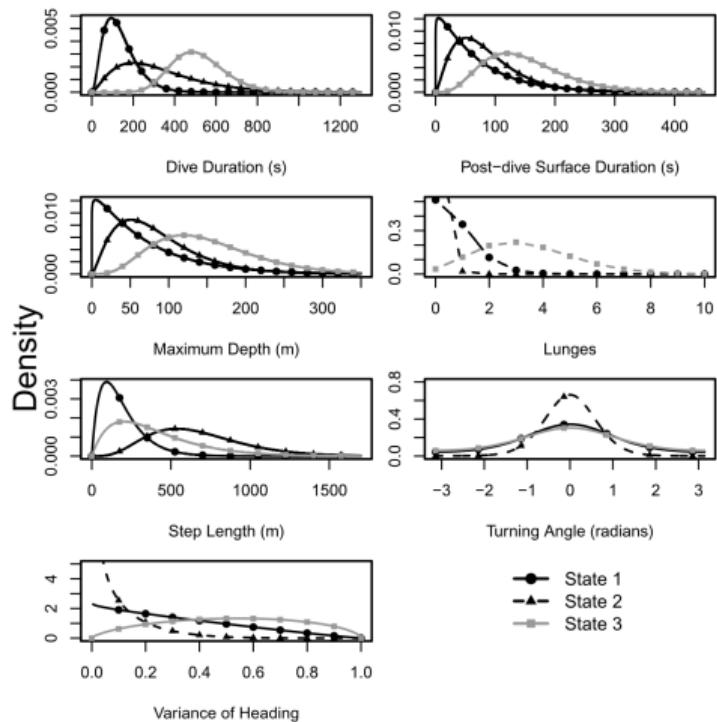
2.1 Multivariate HMMs — blue whale example

Estimated parameters of the state-dependent distributions:

Variable	State 1	State 2	State 3
Duration (gamma)	$\mu = 140$ (132, 148) $\sigma = 80$ (73, 88)	$\mu = 334$ (305, 367) $\sigma = 212$ (187, 241)	$\mu = 516$ (503, 530) $\sigma = 130$ (120, 140)
Surface Duration (gamma)	$\mu = 70$ (64, 77) $\sigma = 68$ (61, 76)	$\mu = 86$ (78, 95) $\sigma = 55$ (47, 65)	$\mu = 151$ (144, 158) $\sigma = 69$ (63, 75)
Maximum Depth (gamma)	$\mu = 32$ (30, 35) $\sigma = 24$ (21, 26)	$\mu = 68$ (59, 79) $\sigma = 65$ (55, 77)	$\mu = 170$ (164, 176) $\sigma = 60$ (56, 65)
Step Length (gamma)	$\mu = 189$ (175, 204) $\sigma = 134$ (121, 149)	$\mu = 675$ (629, 725) $\sigma = 305$ (268, 348)	$\mu = 406$ (376, 439) $\sigma = 287$ (258, 318)
Turning Angle (von Mises)	$\mu == 0$ $\kappa = 1.0$ (0.9, 1.2)	$\mu == 0$ $\kappa = 3.1$ (2.5, 3.7)	$\mu == 0$ $\kappa = 0.8$ (0.6, 1.0)
Heading Variance (beta)	$a = 1.0$ (0.8, 1.1) $b = 2.1$ (1.8, 2.4)	$a = 0.5$ (0.4, 0.6) $b = 5.4$ (4.2, 7.1)	$a = 1.7$ (1.5, 1.9) $b = 1.6$ (1.4, 1.8)
Number of Lunges (Poisson)	$\lambda = 0.7$ (0.6, 0.8)	$\lambda = 0.0$ (0.0, 0.1)	$\lambda = 3.4$ (3.2, 3.6)

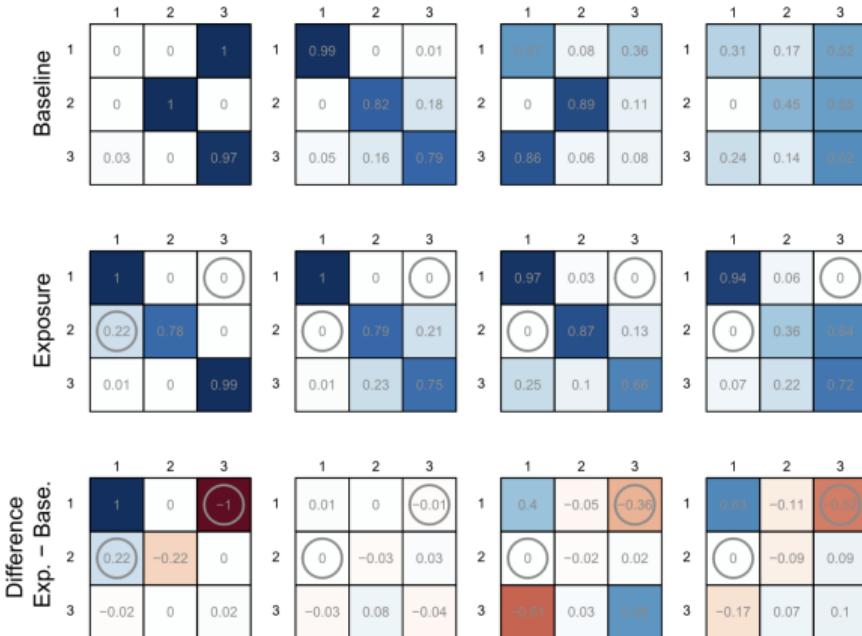
2.1 Multivariate HMMs — blue whale example

Estimated state-dependent distributions:



2.1 Multivariate HMMs — blue whale example

Estimated t.p.m.s (without sonar exposure, with sonar exposure, and difference) in four different contexts:



Outline

1 Introduction to HMMs (13:30–15:00)

- 1.1 Model formulation and dependence structure
- 1.2 Likelihood evaluation and model fitting
- 1.3 Model selection, model checking, and state decoding

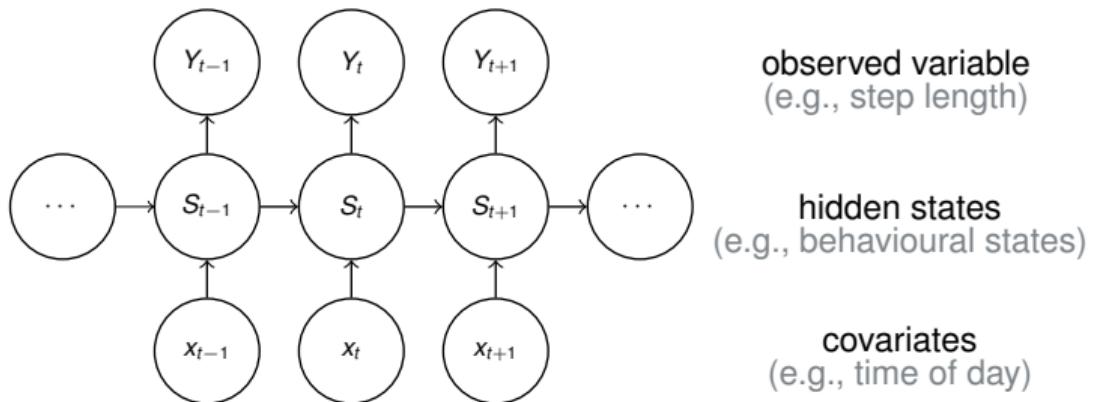
2 Extensions of the basic HMM (15:30–16:45)

- 2.1 Multivariate HMMs
- 2.2 Covariate-dependent state processes**

Summary, questions, etc. (16:45–17:00)

2.1 Covariate-dependent state processes

- Idea: express state transition probabilities as functions of covariates to infer how state switching depends on external (or internal) factors.



- Incorporating covariates into the state process is straightforward — **the likelihood form remains unchanged** (all that changes is that the t.p.m. will depend on covariates).

2.1 Covariate-dependent state processes

- Consider again the 2-state HMM, but now with x_t affecting the transitions,

$$\Gamma^{(t)} = \begin{pmatrix} 1 - \gamma_{12}^{(t)} & \gamma_{12}^{(t)} \\ \gamma_{21}^{(t)} & 1 - \gamma_{21}^{(t)} \end{pmatrix},$$

where, for $i \neq j$,

$$\gamma_{ij}^{(t)} = \text{logit}^{-1}(\beta_0^{(ij)} + \beta_1^{(ij)} \cdot x_t) = \frac{\exp(\beta_0^{(ij)} + \beta_1^{(ij)} \cdot x_t)}{\exp(\beta_0^{(ij)} + \beta_1^{(ij)} \cdot x_t) + 1}.$$

- The parameters to be estimated for the state process are now $\beta_0^{(ij)}$ and $\beta_1^{(ij)}$ instead of γ_{ij} — this essentially is a logistic regression within the HMM.
- Extension to multiple covariates, quadratic effects, interactions, etc. is straightforward.
- For more than two states, the multinomial logit function can be used to ensure that the rows of the t.p.m. sum up to one.

2.1 Covariate-dependent state processes — seasonality

Special case — **seasonality**:

- In many real data applications — not just in ecology! — there is within-day or within-year variation.
- In that case, (some) model parameters ought to be modelled as functions of the covariate time of day.
- However, predictors ought to return to where they started after completing a full cycle — this can be achieved using **trigonometric functions** (with period equal to day, year, or whatever the assumed cycle length).
- For example, for 24 hour cycles, consider the 2-state HMM as before, but now with

$$\begin{aligned}\gamma_{ij}^{(t)} &= \text{logit}^{-1} \left(\beta_0^{(ij)} + \beta_1^{(ij)} \cdot \sin \left(\frac{2 \cdot \pi \cdot x_t}{24} \right) + \beta_2^{(ij)} \cdot \cos \left(\frac{2 \cdot \pi \cdot x_t}{24} \right) \right) \\ &= \frac{\exp \left(\beta_0^{(ij)} + \beta_1^{(ij)} \cdot \sin \left(\frac{2 \cdot \pi \cdot x_t}{24} \right) + \beta_2^{(ij)} \cdot \cos \left(\frac{2 \cdot \pi \cdot x_t}{24} \right) \right)}{\exp \left(\beta_0^{(ij)} + \beta_1^{(ij)} \cdot \sin \left(\frac{2 \cdot \pi \cdot x_t}{24} \right) + \beta_2^{(ij)} \cdot \cos \left(\frac{2 \cdot \pi \cdot x_t}{24} \right) \right) + 1}.\end{aligned}$$

2.2 Covariate-dependent state processes in R

Example in R — covariate-dependent state processes for the elephant data:



https://github.com/timoadam/mtsa/blob/main/2_Extensions_of_the_basic_HMM.R.

2.2 Covariate-dependent state processes — grey seal example

Another example from the literature (van Beest *et al.*, 2019¹⁷):

Received: 31 December 2018
Accepted: 23 March 2019
Published online: 06 April 2019

OPEN **Classifying grey seal behaviour in relation to environmental variability and commercial fishing activity - a multivariate hidden Markov model**

Fiona M. van Beest¹*, Elske Mews², Sander Elkenkamp³, Patrick Schuhmann⁴, Dimitri Tsolak⁵, Tijn Wobbe⁶, Valérie Bartardie⁷, Fransois Bastardie⁸, René Dietz⁹, Christiaan van Dorrien¹⁰, Morten Tange Olsen¹¹, Jónas Nielson¹² & Rens Langrock¹³

Abstract Grey seals are highly mobile marine mammals whose behaviour is influenced by anthropogenic activity and environmental conditions is important to guide marine conservation. We investigate the relationship between environmental variables and grey seal behaviour using a multivariate hidden Markov model. Data were collected from 100 GPS-collared grey seals in the North Sea. We used multiple environmental covariates to predict the probability of being at sea or resting. We also considered the effect of commercial fishing activity on the probability of being at sea. We found significant changes in movement behaviour of grey seals at sea. Sea depth, dive duration, surface distance, temperature and wind speed were all significant covariates. The probability of being at sea increased with increasing water temperature and decreased with increasing wind speed. The probability of being at sea decreased with increasing sea depth. Dive duration had a pronounced effect on state occupancy. The probability of being resting was higher at sites from which seals were more likely to be disturbed. The probability of being at sea decreased with increasing surface distance, indicating very only 1% of their time of use highlighting its importance compared to distance in sea depth. The probability of being resting increased with increasing surface distance. In conclusion, our study provides a scientific basis for developing management strategies that consider ecological requirements of grey seals to ensure their long-term survival.

Keywords grey seal, behavioural ecology, environmental variability, commercial fishing, hidden Markov model, multivariate analysis, marine mammal, North Sea

¹ Department of Mathematics, University of Amsterdam, Amsterdam, The Netherlands
² Department of Business Administration and Economics, Bielefeld University, Bielefeld, Germany
³ Department of Mathematics, University of Amsterdam, Amsterdam, The Netherlands
⁴ Department of Mathematics, University of Amsterdam, Amsterdam, The Netherlands
⁵ Department of Environmental Research and Monitoring, British Museum of Natural History, London, United Kingdom
⁶ Department of Mathematics, University of Amsterdam, Amsterdam, The Netherlands
⁷ Department of Mathematics, University of Amsterdam, Amsterdam, The Netherlands
⁸ Department of Mathematics, University of Amsterdam, Amsterdam, The Netherlands
⁹ Department of Mathematics, University of Amsterdam, Amsterdam, The Netherlands
¹⁰ Department of Mathematics, University of Amsterdam, Amsterdam, The Netherlands
¹¹ Department of Mathematics, University of Amsterdam, Amsterdam, The Netherlands
¹² Department of Mathematics, University of Amsterdam, Amsterdam, The Netherlands
¹³ Department of Mathematics, University of Amsterdam, Amsterdam, The Netherlands

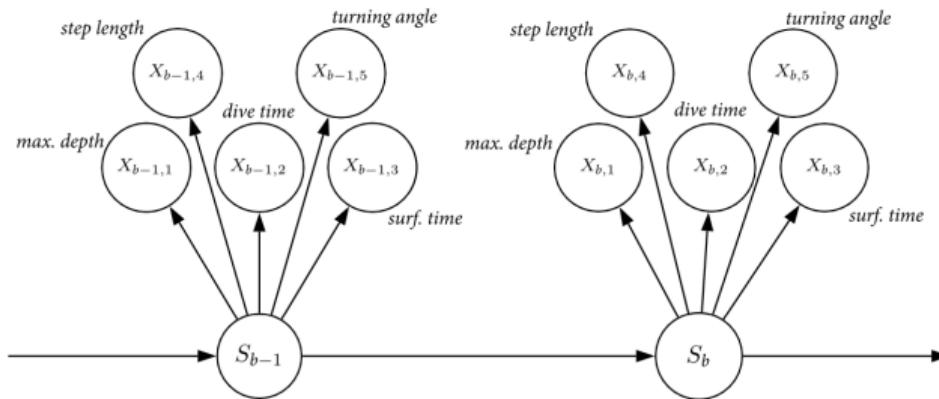
* To whom correspondence should be addressed. E-mail: f.vanbeest@math.uva.nl



¹⁷van Beest, F.M., Mews, S., Elkenkamp, S., Schuhmann, P., Tsolak, D., Wobbe, T., Bartolino, V., Bastardie, F., Dietz, R., von Dorrien, C., Galatius, A., Karlsson, O., McConnell, B., Nabe-Nielsen, J., Tange Olsen, M., Teilmann, J., and Langrock, R. (2019). Classifying grey seal behaviour in relation to environmental variability and commercial fishing activity — a multivariate hidden Markov model. *Scientific Reports*, 9(1), 5642.

2.2 Covariate-dependent state processes — grey seal example

Variables included in the state-dependent process:

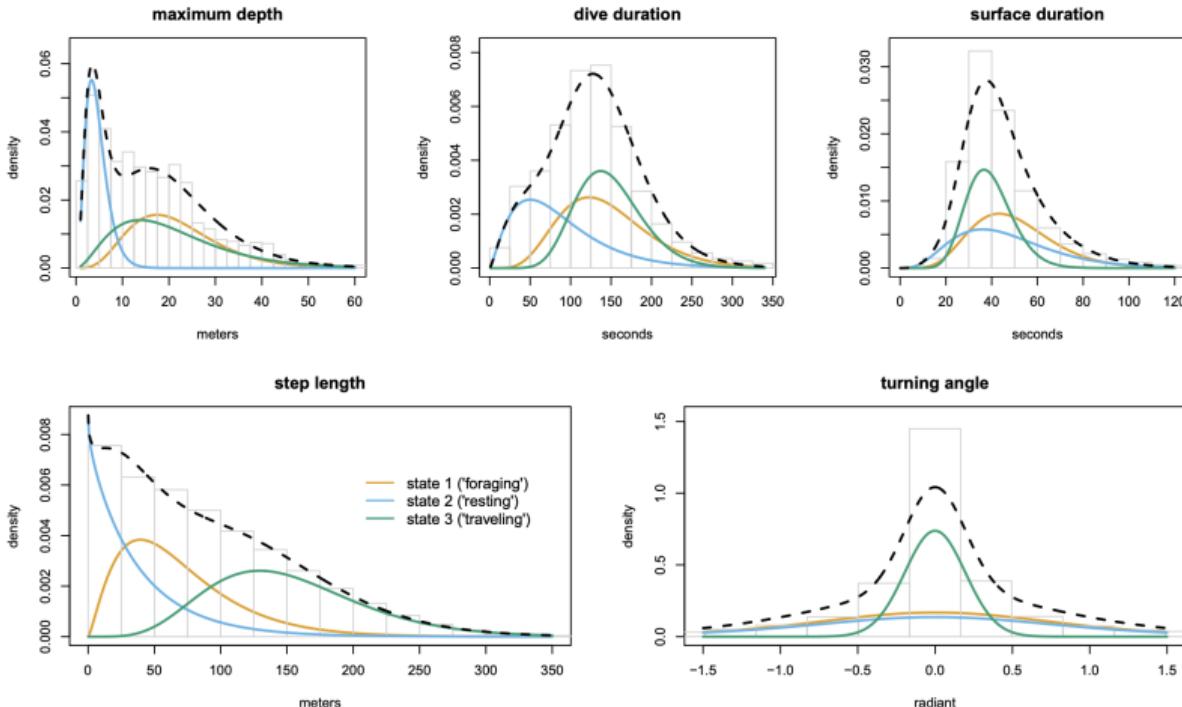


Covariates included in the state process:

$$\begin{aligned}\gamma_{ij}^{(t)} = \text{logit}^{-1}(&\beta_0 + \beta_1 \cdot \text{bathymetry}_t + \beta_2 \cdot \text{sea surface temperature}_t + \beta_3 \cdot \text{sea surface salinity}_t \\ &+ \beta_4 \cdot \text{seabed slope}_t + \beta_5 \cdot \text{distance to active fishing net}_t).\end{aligned}$$

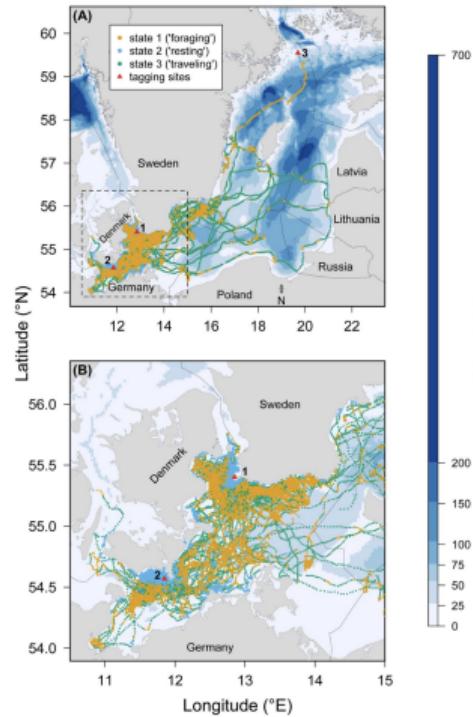
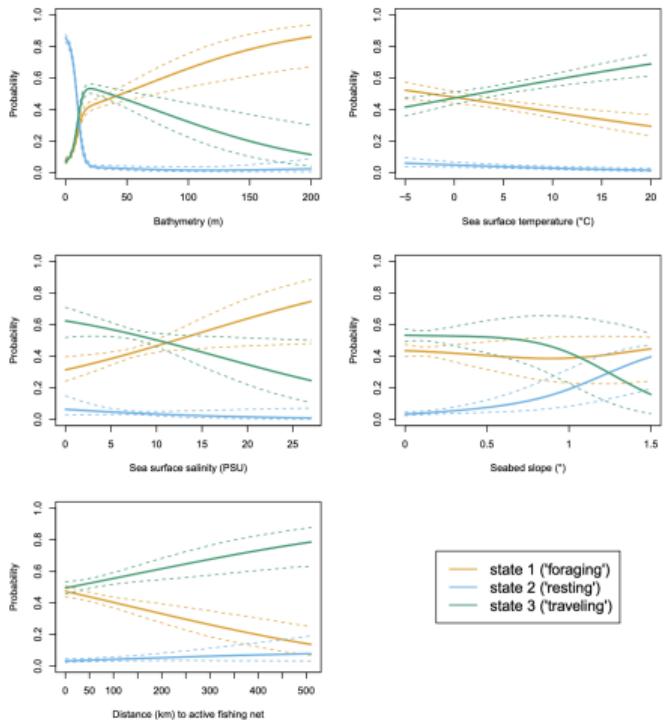
2.2 Covariate-dependent state processes — grey seal example

Estimated state-dependent distributions:



2.2 Covariate-dependent state processes — grey seal example

Stationary distribution as function of the covariates and Viterbi-decoded states:



Outline

■ Introduction to HMMs (13:30–15:00)

- 1.1 Model formulation and dependence structure
- 1.2 Likelihood evaluation and model fitting
- 1.3 Model selection, model checking, and state decoding

■ Extensions of the basic HMM (15:30–16:45)

- 2.1 Multivariate HMMs
- 2.2 Covariate-dependent state processes

3 Summary, questions, etc. (16:45–17:00)

3. Summary, questions, etc.

What have we learned in this session?

- A basic HMM comprises two stochastic processes: a hidden state process and an observed state-dependent process.
- The likelihood of an HMM can be evaluated using the forward algorithm and maximised using Newton-Raphson-type optimisation routines.
- Multivariate HMMs can be used to model multiple variables that are driven by the same state process.
- In multivariate HMMs, we often assume contemporaneous conditional independence.
- Covariates can be incorporated into the state process, where they affect the transition probabilities.
- Trigonometric functions can be used to model seasonality.

3. Summary, questions, etc.

This session was **too short** to cover the following topics:

- Markov-switching regression models (incorporating covariates into the state-dependent process).
- Non-standard dependence structures (higher-order Markov, semi-Markov, feedback effects, within-state dependence of observations, etc.).
- Hierarchical HMMs (for multi-scale data).
- Random effects (when jointly modelling multiple time series).
- HMMs with continuous-valued state processes.
- Continuous-time HMMs.
- Non-parametric HMMs (based on splines).
- ...

Anything you want to discuss — just **contact me**: [timo.adam@uni-bielefeld.de!](mailto:timo.adam@uni-bielefeld.de)

3. Summary, questions, etc.

That's it — thanks for your attention!