Fitting an HMM

After the data has been collected...

Fitting an HMM to data requires specification of:

- the number of states
- the forms of the state-dependent distributions
- whether to estimate the initial distribution (or not!)
  can be taken to the be stationary distribution, which is solution to: $\boldsymbol{\delta} = \boldsymbol{\delta\Gamma}$

R Packages to Fit an HMM

General Purpose:

depmixS4: https://cran.r-project.org/web/packages/depmixS4/index.html

HiddenMarkov: https://cran.r-project.org/web/packages/HiddenMarkov/index.html

Specific:

moveHMM: https://cran.r-project.org/web/packages/moveHMM/index.html

HMMcopula: https://cran.r-project.org/web/packages/HMMcopula/HMMcopula.pdf

Fitting an HMM from scratch

Let's go behind the scenes and figure out how to fit an HMM to data without any package.

First, we need to write out the likelihood function. For inference, two likelihood functions are commonly used. Let $\boldsymbol{\theta}$ represent all parameters requiring estimation in an HMM.

Complete-data likelihood: $CD\mathcal{L} = f(\boldsymbol{\theta}|\boldsymbol{x}, \boldsymbol{c})$

Likelihood: $\mathcal{L} = f(\boldsymbol{\theta}|\boldsymbol{x})$

Likelihood function

We will focus on use of the (marginal) likelihood function for inference. This means that in practice, we need to sum over all possible state sequences to marginalize out the state process. This means we need to compute:

$$\mathcal{L} = \sum_{C_1=n}^{N} \cdots \sum_{C_T=n}^{N} \left[\delta_1 f(X_1|C_1) \prod_{t=2}^{T} \gamma_{C_{t-1},C_t} f(X_t|C_t) \right]$$

where $\gamma_{C_{t-1},C_t} = \Pr(C_t|C_{t-1})$. Is this possible? (Spoiler: Yes!)

Use of the likelihood function directly allows for inference in other manners aside from using the expectation-maximization algorithm (Baum-Welch) for HMMs.

Forward Algorithm

The so-called **forward algorithm** is a recursive scheme for efficient summation over all possible state sequences and estimation of the likelihood function.

We make use of the forward variables, $\{\boldsymbol{\alpha}_t\}_{t=1}^{T}$ where $\boldsymbol{\alpha}_t = (\alpha_{t,1}, \ldots, \alpha_{t,N})$

Beginning at t=1, $\boldsymbol{\alpha}_1 = \boldsymbol{\delta}^T \mathbf{P}(x_1)$, then for t=2, …., T, we have $\boldsymbol{\alpha}_t = \boldsymbol{\alpha}_{t-1} \boldsymbol{\Gamma} \mathbf{P}(x_t)$

for $\mathbf{P}(x_t) = diag\{f(x_t|C_t = 1), \ldots, f(x_t|C_t = N)\}$

The likelihood function from the previous slide can be computed as: $\mathcal{L} = \sum_{n=1}^{N} \alpha_{T,n}$

The likelihood as a matrix product: $\mathcal{L} = \boldsymbol{\delta}^T \mathbf{P}(x_1) \boldsymbol{\Gamma} \mathbf{P}(x_2) \cdots \boldsymbol{\Gamma} \mathbf{P}(x_T) \mathbf{1}$

Maximization of the likelihood function

Inferential approach #1:

1. Find the set of parameters $\hat{\boldsymbol{\theta}}$ that maximizes the likelihood function for the observations $x_1, x_2, \ldots, x_T$
   a. Make use of the forward algorithm for likelihood evaluation
   b. Map bounded parameters to unbounded spaces
2. Quantify uncertainty around $\hat{\boldsymbol{\theta}}$
3. Estimate the underlying state sequence $\hat{c}_1, \hat{c}_2, \ldots, \hat{c}_T$

Bayesian inference

Inferential approach #2:

1. Set a joint prior distribution $\pi(\boldsymbol{\theta})$ for parameters $\boldsymbol{\theta}$
    a. Make use of the forward algorithm for likelihood evaluation
    b. Map bounded parameters to unbounded spaces
2. Compute the posterior distribution, $p(\boldsymbol{\theta}) \propto f(\boldsymbol{x}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})$
3. Estimate the underlying state sequence using $p(\boldsymbol{\theta})$

I use the software Stan for Bayesian inference: https://mc-stan.org

Estimating the model in practice

Identifiability of an HMM requires that:

- the state-dependent distributions are 'distinct'
- the t.p.m. is ergodic and full rank

In practice:

- the state-dependent distributions should be *very* distinct -- the less overlap the better
- more persistence in the states the more easily identified the model is! that is, the diagonal entries of Γ should be high (anecdotally > 0.8)

Let's fit some HMMs