Baum-Welch implementation

Carlota Carbajo moral s202424
Celia Burgos Sequeros s202423
Christian Holm Johansen s202770
Isabel Diaz Pines-Cort s202406

Hidden markov model

A HMM is a statistical model that describes a system that is a Markov process.

- It can exist in N unobservable **hidden states**.
- It behaves differently depending on the hidden state it is in.

Applications in **bioinformatics**:

- Sequence alignment
- Protein structure prediction
- Identification of functional motifs

Hidden markov model

A HMM is a statistical model that describes a system that is a Markov process.

- It can exist in N unobservable **hidden states**.
- It behaves differently depending on the hidden state it is in.

When transition probabilities between states and different state behaviours are unknown...
The **Baum-Welch algorithm** is used to learn them from data.

Applications in **bioinformatics**:

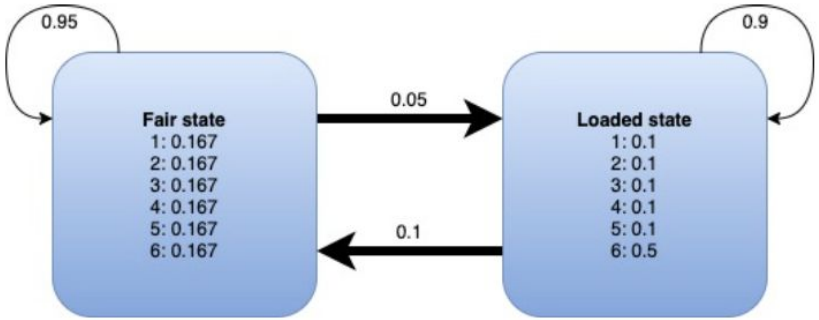
- Sequence alignment
- Protein structure prediction
- Identification of functional motifs

The Unfair Casino Problem

2 hidden states:

- Fair die
- Loaded die

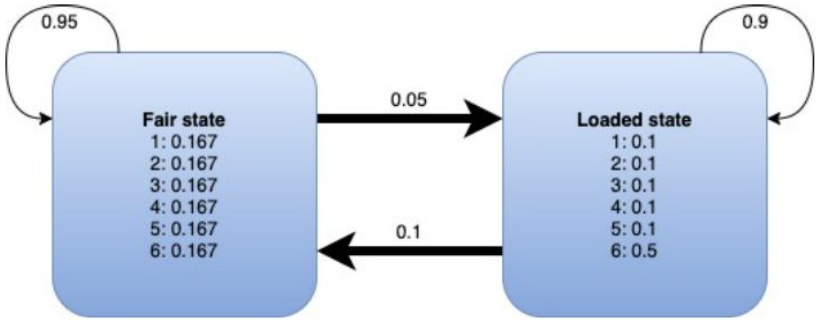
Dice can be exchanged between rolls with low probability



Sequence generation

- Synthetic sequences generated by model

- 100 sequences for training and 100 for testing

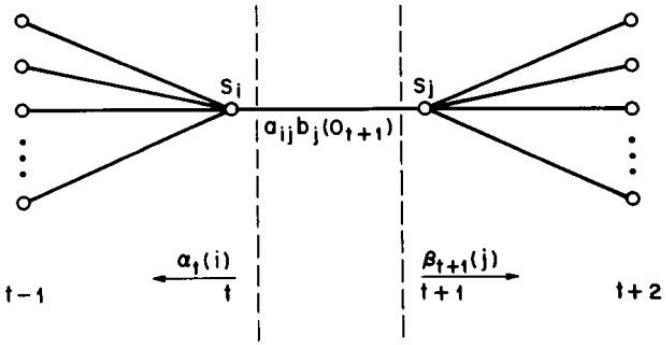


Forward and backward algorithm

- Need measure for probability of specific transition
  At each timepoint

- Forward and backward algorithms allow for efficient computing

$$\xi_{kl}(i) = \frac{\alpha_k(i) a_{kl} e_l(x_{i+1}) \beta_l(i+1)}{\sum_k \sum_l \alpha_k(i) a_{kl} e_l(x_{i+1}) \beta_l(i+1)}$$



(Rabiner 1989)

Baum-Welch algorithm

- $\gamma_k(i)$ is percentage of time in state k at time i

$$\gamma_k(i) = \sum_l \xi_{kl}(i) \qquad \gamma_k(T) = \frac{\alpha_k(T)\beta_K(T)}{\sum_l \alpha_l(T)\beta_l(T)}$$

- This can then be used to re-estimate the emission and transition probabilities

$$\hat{a}_{kl} = \frac{\sum_r^R \sum_{i=1}^{T-1} \xi_{kl}(i)}{\sum_r^R \sum_{i=1}^{T-1} \gamma_k(i)}$$

$$\hat{e}_k(s) = \frac{\sum_r^R \sum_{i=1, X_i=v_s}^{T} \gamma_k(i)}{\sum_r^R \sum_{i=1}^{T} \gamma_k(i)}$$

Performance evaluation

- In terms of the model parameters:

$$RMSE_i = \sqrt{(true - predicted)^2}$$

- In terms of the predictive ability:

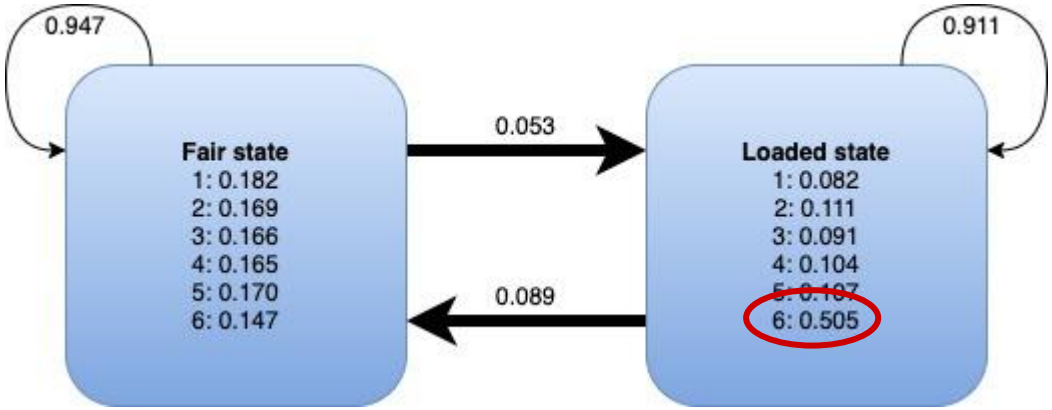
Viterbi algorithm

$$P_l(i+1) = p_l(i+1) \cdot \max_k (P_k(i) \cdot a_{kl})$$

Posterior decoding

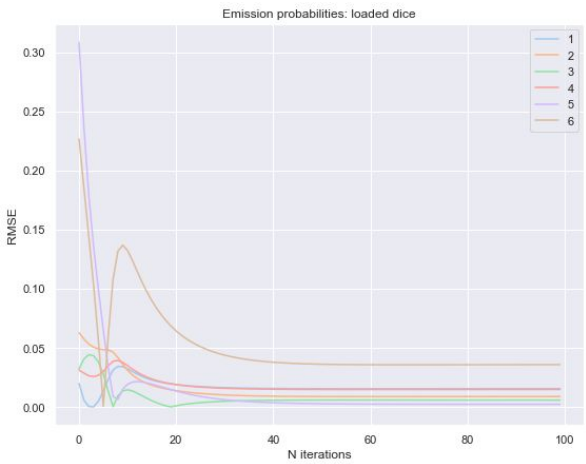
$$P(\pi_i = k|x) = \frac{P(x, \pi_i = k)}{P(x)} = \frac{\alpha_k(i)\beta_k(i)}{P(x)}$$

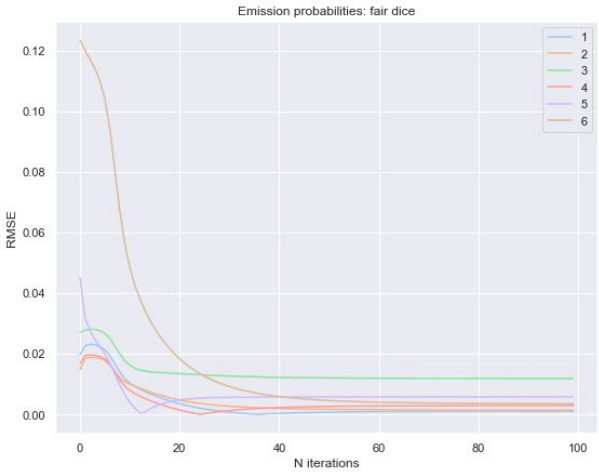
Estimated model

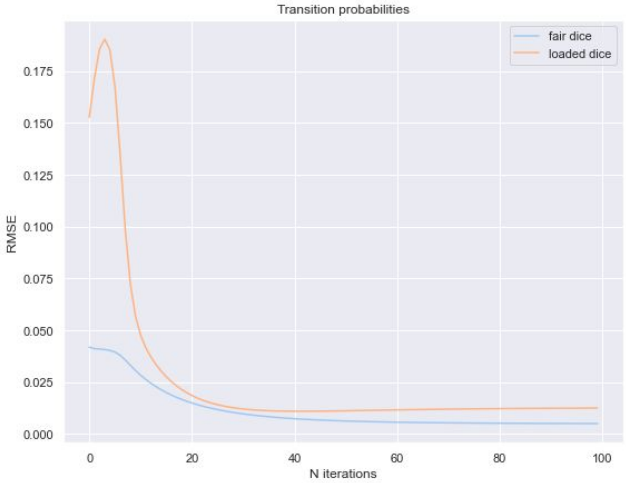


The estimated model shares structure with the true model.

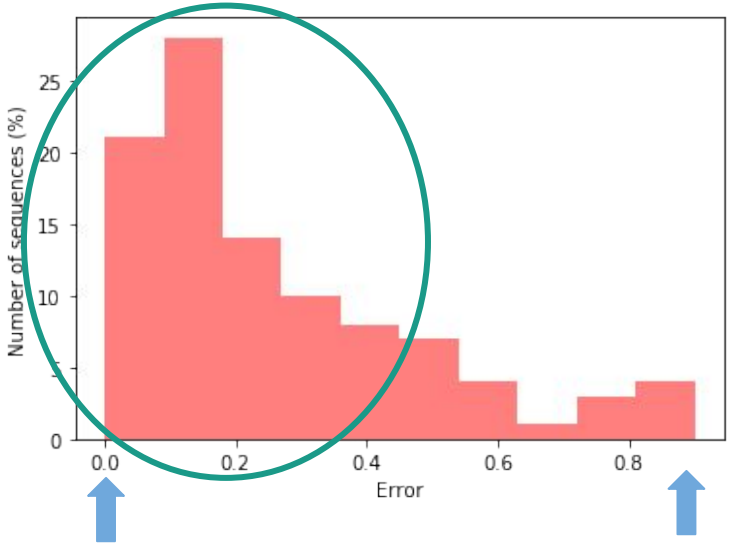
Re-estimating the model using Baum-Welch







Predictive ability of the model - Viterbi decoding



Perfect prediction

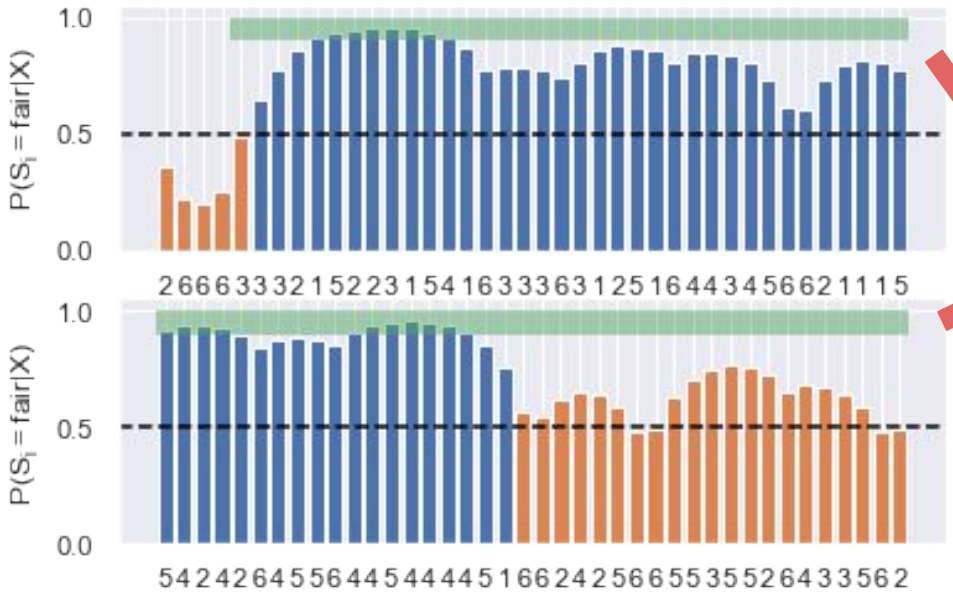
90% of time-steps are assigned the wrong state

Most probable sequence of underlying states for the test sequences

30% <0.1,
59% <0.25
85% <0.5

Given that there are two hidden states in the system, that an error > 0.5 represents a prediction is worse than random

Predictive ability of the model - Posterior decoding



Posterior probabilities of being in the fair state

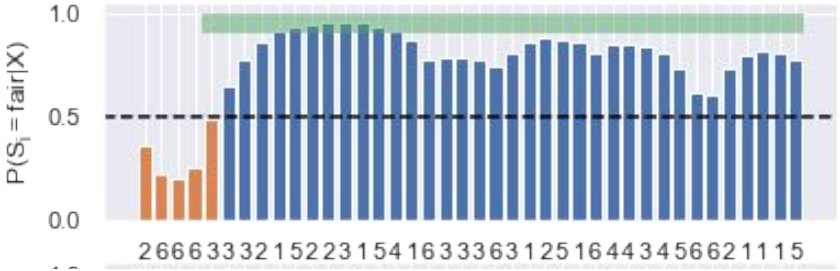
Viterbi decoding can accurately predict states when their probabilities are close to 1 or 0, but has difficulty doing so when they are closer to 0.5

Discussion - Model generation and predictive power

The Baum-Welch algorithm can estimate the HMM parameters almost perfectly

Types of sequences that give rise to errors:

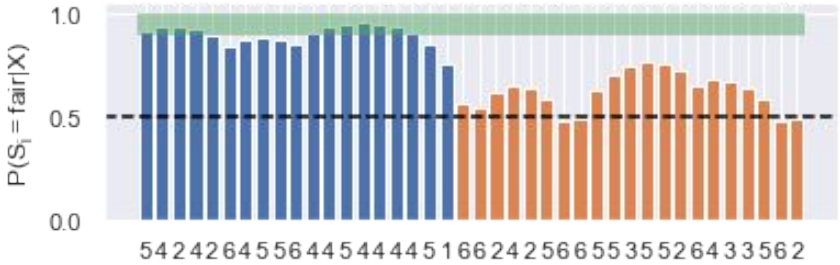
- Regions of transition between states present ambiguous probabilities which lead to errors in state selection



Discussion - Model generation and predictive power

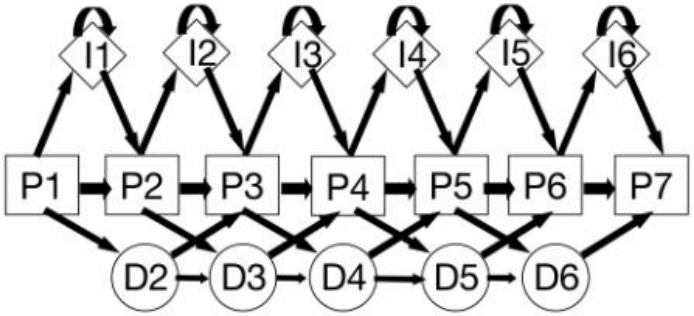
- Rare events → Two consecutive state transitions occur

- Long mis-predicted regions



Discussion

Regarding motif finding, the HMM can be seen as an extension of the PSSM, where it is possible to also model insertions and deletions



Profile-HMM

(Lund, Ole et al. 2018)

This more advanced type of HMM can be modelled using a multiple alignment and from this calculate the probabilities of each of the states

However, the Baum-Welch algorithm allows one to generate the HMM with just the raw sequences and then train the model using them.

Conclusions

1. When transition and emission probabilities of a HMM are unknown, the Baum-Welch algorithm can accurately learn these parameters from data.
2. With the HMM built, it is possible to predict the hidden states of new data with varying but overall good accuracy.

   While the Viterbi algorithm can trace the most likely state path given a sequence of emissions, with posterior decoding we get more detailed insight on the step probabilities and the reliability of the prediction.
3. HMMs and Baum-Welch training are great tools for biological sequence analysis, as they don't require labeled data and can model insertions and deletions in a way that traditional weight matrices can't.