**Latent Neural Networks**

*Hidden Markovian Models (HMMs) basics*

An v.

A trial is a trajectory of observation: which we obtain through the following steps:

1. Sample the initial hidden states using the initial distributions π. We can get, for instance, .
2. Sample the following hidden states using the transition matrix. E.g., if we sample
3. Similarly, we repeat the process for
4. Finally, using the emission matrix, , we sample the observations from

We would like to define a teacher HMM with fixed parameters , which will be our “ground truth”. Afterwards, we define student(s) HMM(s) with random initial parameters. We will use the emissions we got from the trial as our data to train the students to find the optimal parameters which maximize the (log) likelihood or minimize the negative likelihood.

In our case, the likelihood is the probability: where are the emissions we obtained from our trial.

Since the Markovian property applies here, only cares about .

Co-smoothing = ?

Filtering = ?

Decoding (and cross-decoding) = ?

…

#TODOs:

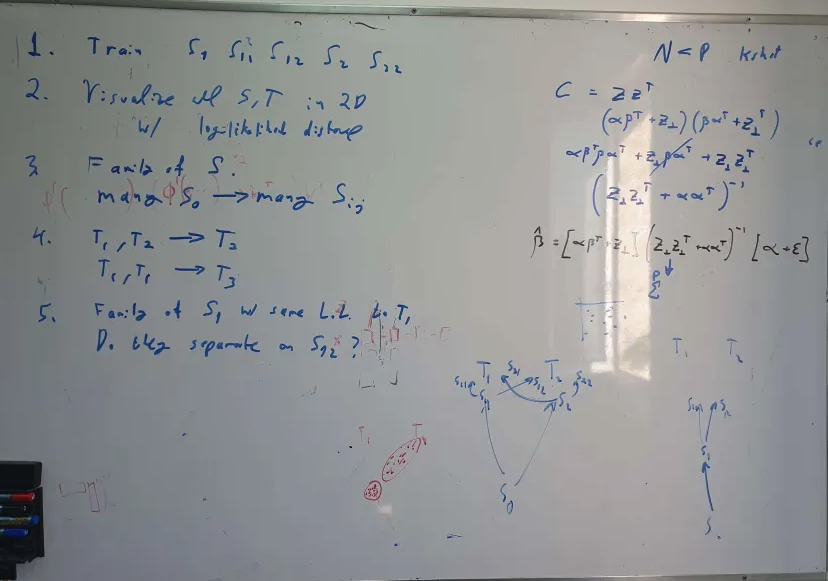
* Make the teacher dynamics a ring.
* Implement decoding comparisons.
* Omri talked about- cohona map, emap, other ditance matrix ?
* HMM that creates HMMs

Tasks/milestones:

1. Make sure the results from the current models make sense - Likelihood converges as we extend trials or increase the learning rate, teacher likelihood given the teacher’s emissions test dataset should give the best performance, etc.
2. Given a teacher model , i.e. our “ground truth”, we will create a series of models from it () by perturbating it. Such that = + perturbation, = + perturbation, and so on… Thus, represent how the brain changes as it learns. We want to start by training several students, respectively on .
3. Visualize the , likelihoods on the emissions of in a 2D plot w/ likelihood distance.
4. Generalize the training to

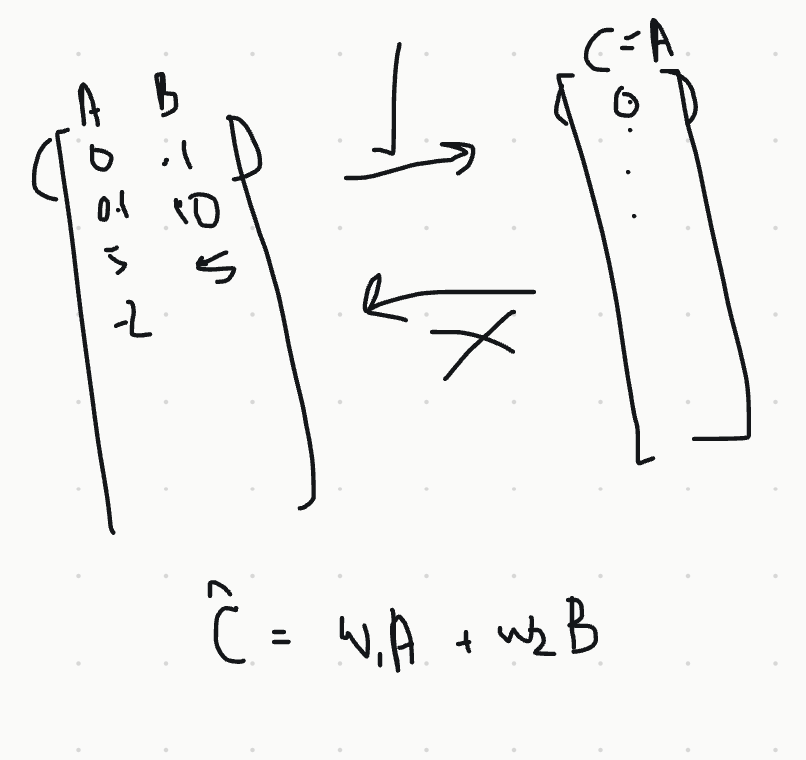
Goal:

Test whether the learning process, i.e. a series of models originating from perturbations of a model, could help discriminate “bad” models. Would the results be supported by the cross-decoded method mentioned in the paper (K. Dabholkar, O. Barak; 2024)?



*Decoding*

Given 2 models, A and B, if we decode from A 🡪 B we attempt to see how well the states of A can adjust themselves to B, i.e. how well can A decode B. For instance, if A has more states than B, we check whether a subset of its state can form the same structure B has. B on the other hand, cannot have enough information about A, because it has less states. In this example, this decoding score from A🡪B will be lower from B🡪A:



In the image we can see that the left model can decode C by having w1=1, w2=0, but C cannot decode the left model properly because it has only 1 column.

It is done by training a logistic regression with the features being A’s prediction probabilities and the labels B’s prediction probabilities. The error is the result. The lower the error the better the decoding. If the error is low for both ways, A->B and B->A, it indicates that the latent structure (states/variables) are similar to one another.