

# Learning Lab

This lab should be done in pairs. Try these experiments using both versions of your MCMC code.

## Faculty data

I have updated the specification given below. I have tried to be more clear about what plots are required. I still want you to experiment. Find something you think is interesting to show.

Experiment with parameter learning in the faculty data from the [mcmc lab](#):

- Add hyper-parameters and learn them one-by-one. Compare your results as you go.
- How does learning the hyper-parameters affect the posterior distributions for the mean and variance? Compare the posterior distributions you obtain with learned hyper-parameters to the posterior distributions you obtained when the prior parameters (hyper-parameters) were hard coded.

In the end you should have four graphs for the mean and variance (hint: one joint mean variance graph is better than two independent ones) and one for the hard coded hyper-parameters. You will also need a few graphs of the distributions over the hyper parameters for each case (hint: one graph with multiple lines is easier to draw conclusions from than several independent graphs). You will need to check the mixing with mixing graphs, but you do not need to show them to me.

## Alarm Model

Extend the [Alarm](#) model from the [mcmc lab](#) to allow for multiple observations and parameter learning. However we will modify it as follows:

- $P(b)=0.2$  and  $P(e)=0.3$  (think of them as lifetime probabilities or the author has moved to a very unstable neighborhood...)
- $P(a|\sim b, \sim e) = 0.2$  (bad alarm too...)
- $P(j|\sim a) = 0.2$  and  $P(m|\sim a) = 0.3$  (but friendlier neighbors!)

In this case generate your own test data (hint: generating data is like inference, just the knows and unknowns trade places, you should be able to use your mcmc code!). Try the following:

- Add three hyper-parameters to learn, then try learning with 1/2 of the parameters, then try learning them all. The point is to get a feel for how adding hyper-parameters affects your network. Compare your results as you go, and compare your results to the known true parameters.
- Experiment with the amount of data you give your parameter learning version of the network. Note that your objective here is to see how data affect the learning. Obviously more data allows you to better learn the parameters. Unfortunately more data will take longer too. Do not get carried away with the amount of data or you will spend the rest of the

semester on this! You might want to start with just 100 sets of observations, you may need to go even lower. Show me a few plots so I can see a change in the learning as you changed the amount of data.

- Experiment with the true parameters. For example, go back to the original parameters ( $P(b) = 0.001$ , etc.), does this make the system harder or easier to learn? What in general makes a net harder to learn? (a few more plots)
- Add a hyper-hyper-parameter or two. How does this affect your results. (one or two more plots)

Putting it all together:

You do not need to include the hyper-hyper parameters for the following part:

- Try adding some observations with missing data (removed by hand if that is easiest for you). For example, suppose for some rows in your input data, you do not know if John Called or not, in others we do not know if there was a burglary or not. Again, remove a large enough percentage of the data so I can see a change in the learning. (a few plots)
- Add inference: Given a batch of your data, assume that Mary has just called. What is the posterior probability of a burglary given your training data and the fact that Mary called. (one more plot)
- Change the parameters and generate some data (with missing data). Post you data on the wiki. In your report tell ME what the parameters are so I can compare the true values with the values learned by your classmates (ON THE BURGLARY AND ALARM NODES ONLY).
- Re-run your learner (with all hyper parameters, but no hyper-hyper-parameters) with data from another group that has been posted on the wiki. Report on your results by plotting the distributions of the hyper parameters ON THE BURGLARY AND ALARM NODES ONLY.

## Report

In general I would like to see a lot of graphs and not much writing. What writing you do should be pointing out the interesting things about the graphs and drawing conclusions.

Note that you can go back to working in pairs for this lab. Since you will each have your own mcmc code, you might want to try running a few of your experiments on both versions and comparing the results.