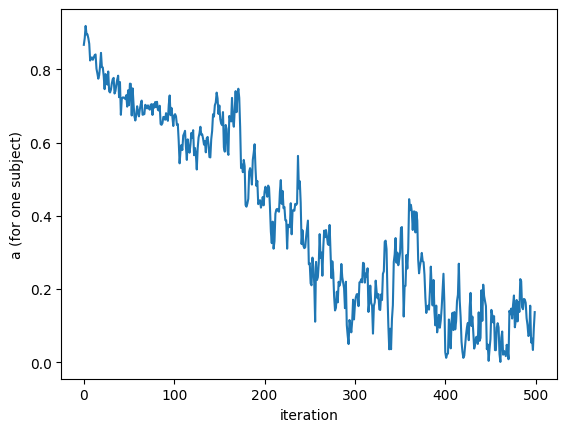
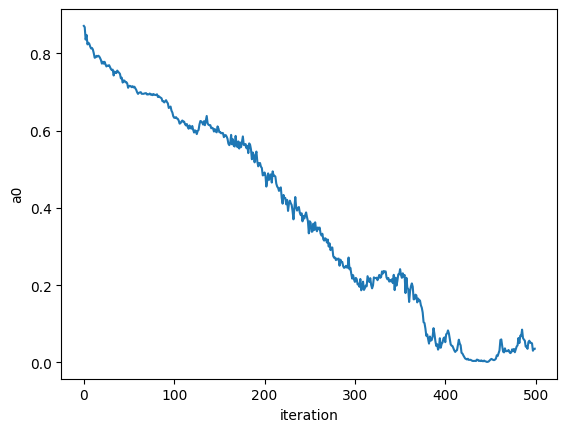
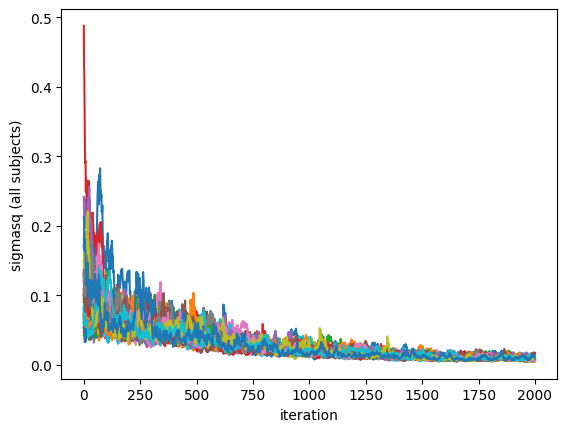
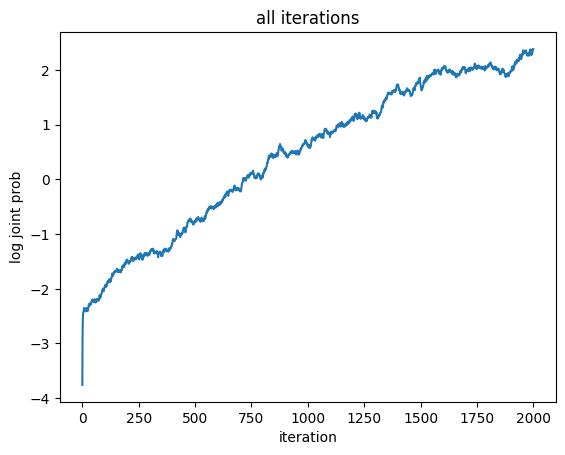
The hMFC is a complex model with many global and local parameters to be estimated. On simulated data the model works great. However, fitting hMFC on empirical data sometimes results in quirky model behavior. In this tutorial we’ll discuss how to recognize some potential quirky behavior and how to solve it. At the heart of our solution will often be fixing some hyperparameter such that they are not estimated.

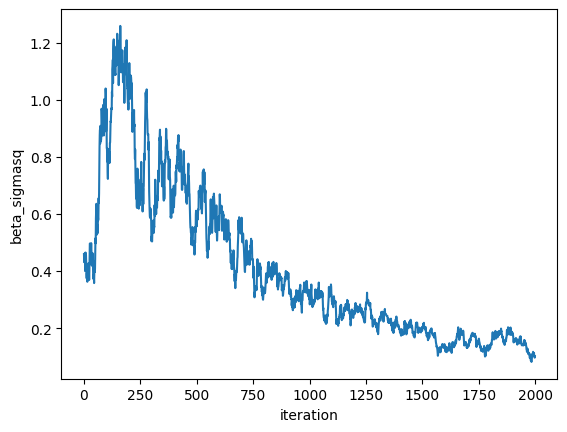
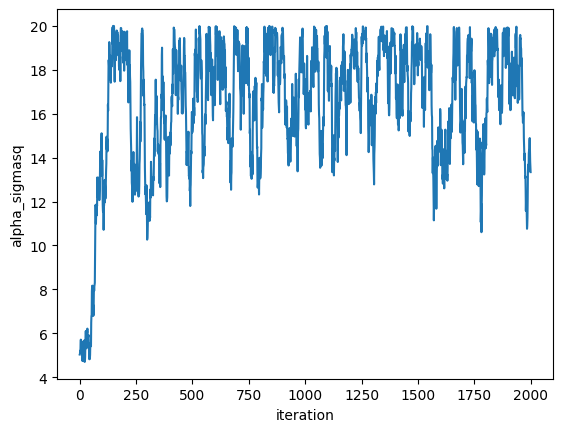
**If a0 and all individual a’s go down over iterations:**



* **Reason**: the centering of the criterion fluctuations only happens afterwards and is not incorporated in the posterior directly. This means that before the centering our estimated fluctuations can be shifted all the way up or down (i.e., it doesn’t have a mean of zero). This non-zero mean can be very unlikely for the sampler since we don’t allow an AR intercept. Without an AR intercept the time series has by definition a mean of zero (mean = intercept/(1-ar)). The model tries to deal with this *unlikeliness* by pushing all a’s to zero (and thus also a0) and thus just estimating a flat line. This problem is more likely to occur with smaller datasets (<1000 trials) because with more trials, the fluctuations are more likely to have a mean of zero.
* **Solution**: Remove the mean-centering when estimating the criterion fluctuations (insert code line of hmfc.py)

**If log prob keeps on increasing over many iterations, sigma\_sq for all subjects goes to zero, alpha is pushing against upper bound, and beta is going to zero:**

****

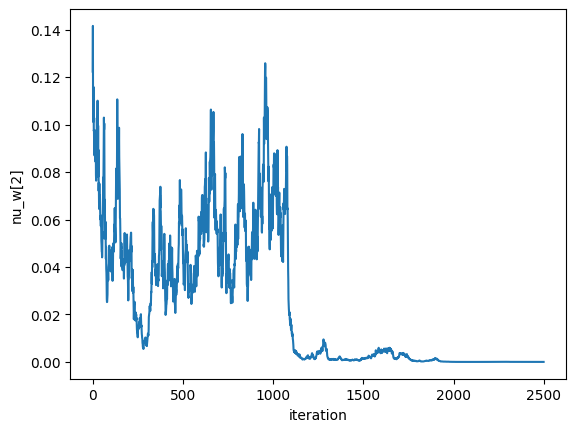
****

* **Reason**: model is making all the sigma\_sq very small. By doing so it is making the slow drifts deterministic (added AR noise is so small that . By doing so the inverse gamma distribution is become very narrow and steep. As a consequence, the log joint prob keeps on increasing.
* **Solution**: fix alpha and beta (don’t estimate them) (insert code line of hmfc.py)
* Note: it is recommended to do a sensitivity analysis: check how much final estimates are influenced by fixed alpha and beta values

**If model ‘breaks’ (log prob makes weird jump and then stops before the specified number of iterations is reached):**

* **Reason**: nu\_w for a certain variable might go to zero over iterations. If this happens, then the variance over subjects cannot be calculate anymore. This results in an error and the model fitting stops.
* **Solution**: lower bound for nu\_w is now specified in the model and this problem should not occur anymore! (insert code line of hmfc.py)

A graph with numbers and lines

Description automatically generated

**Something on increasing the upper bound for alpha and beta?**