Project 2 Part 2: Bayesian Inference for Nonlinear Models

Due November 9, 2021 by 11:59 pm on Canvas

In this project we will consider the topic of parameter inference in dynamical systems.

The deliverable for this project is a report detailing your approach and answers to all of the questions in each section.

Please submit a single PDF file that contains your text and figures. All code should be in the *appendix*, unless you want to provide a BRIEF 5 line snippet that contains only the algorithm you are describing within the text. In the latter case, make absolutely sure that this algorithm follows the narrative of the text. For every figure that you obtain, please make sure that the axes are labeled, any relevant legend is shown, and a caption describes what is being plotted.

The preference is that a formal typed report is submitted. However, a scanned written report that is legible will be accepted. Illegible reports or reports that are primarily/entirely code with no text will not be evaluated.

The same guidelines outlined in Project 1 and the Syllabus apply here.

1 Delayed Rejection Adaptive Metropolis

In this section you will create the Delayed Rejection Adaptive Metropolis (DRAM) algorithm.

1. Implement the Adaptive Metropolis (AM) algorithm with the efficient covariance updating scheme.
2. Implement a delayed rejection algorithm with two levels. Assume that it is initialized with a high level proposal with covariance C and the second level ℓ reduces this covariance by γ. For instance if $\gamma = \frac{1}{2}$ and we have a two level scheme then the covariances of each proposal are C and $\frac{1}{2}C$.
3. Implement a DRAM approach where the top level proposal is the AM algorithm and the subsequent level has a scaled version of the AM covariance matrix.
4. Justify and check your implementation by showing its performance on the simple banana shaped distribution. In this problem the posterior is two dimensional $\pi(x_1, x_2)$. so that

$$\left(x_1, x_2 + (x_1^2 + 1)\right) \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix} \begin{bmatrix} 1 & 0.9 \\ 0.9 & 1 \end{bmatrix}\right) \tag{1}$$

5. Compare the standard Metropolis Hastings, AM, DR, and DRAM, algorithms on the above example and describe which algorithm works best. Justify your answer by presenting the following results (similar to what we did in class)
   (a) Plots of 1D and 2D marginals
   (b) Autocorrelation plots
   (c) Integrated autocorrelation values
   (d) Acceptance ratios
   (e) Visual inspection of mixing.
6. **Do not forget burn-in!**

2 Setting up the truth model for a deterministic system

We will first setup a problem and generate some data. We will consider an SIR model, a common benchmarking problem for MCMC algorithms. This model describes the evolution of a disease in a population by evolving the (S)usceptible, (I)nfected and (R)ecovered portions of the population. In this model it is assumed that humans infect each other directly rather than through a disease vector such as a bird or mosquito. You will setup two versions of this problem. One in which all the parameters are identifiable, and the second in which they are not.

2.1 Identifiable Version

This model has three parameters $\theta = (\beta, r, \delta)$ and the evolution is given by the following nonlinear ODE

$$\frac{dS}{dt} = \delta N - \delta S - \beta I S \tag{2}$$
$$\frac{dI}{dt} = \beta I S - (r + \delta) I \tag{3}$$
$$\frac{dR}{dt} = rI - \delta R, \tag{4}$$

where N is the total population, in our case we fill fix this to $N = 1000$. We will use the initial conditions $S(0) = 900$, $I(0) = 100$, and $R(0) = 0$.
Generate a reference simulation

1. The "True" parameter settings are $\theta = (0.02, 0.6, 0.15)$.
2. Solve this system with a built-in timestepping scheme, i.e., ODE45 in Matlab or `scipy.integrate.solve_ivp` in python using the "RK45" method.

3. Take 61 data points using linearly spaced points between $t \in [0, 6]$. Your data consists of measurements of I corrupted by zero mean noise with standard deviation 50.

4. Plot the trajectories of each state on a separate plot. Label the axes.

5. Plot the data on the appropriate plot.

You will use these 61 noisy measurements of I for the inference problem.

2.2 Non-Identifiable Version

This model has four parameters $\theta = (\gamma, \kappa, r, \delta)$ and the evolution is given by the following nonlinear ODE

$$\frac{dS}{dt} = \delta N - \delta S - \gamma \kappa I S \tag{5}$$

$$\frac{dI}{dt} = \gamma \kappa I S - (r + \delta) I \tag{6}$$

$$\frac{dR}{dt} = rI - \delta R, \tag{7}$$

where N is the total population, in our case we fill fix this to $N = 1000$. We will use the initial conditions $S(0) = 900$, $I(0) = 100$, and $R(0) = 0$.

Generate a reference simulation

1. The "True" parameter settings are $\theta = (0.2, 0.1, 0.6, 0.15)$.

2. Solve this system with a built-in timestepping scheme, i.e., ODE45 in Matlab or `scipy.integrate.solve_ivp` in python using the "RK45" method.

3. Take 61 data points using linearly spaced points between $t \in [0, 6]$. Your data consists of measurements of I corrupted by zero mean noise with standard deviation 50.

4. Plot the trajectories of each state on a separate plot. Label the axes.

5. Plot the data on the appropriate plot.

You will use these 61 noisy measurements of I for the inference problem.

3 Apply DRAM for Bayesian Inference

For this problem you will use the DRAM algorithm to learn the SIR model parameters. Note that because the dynamics are deterministic, we do not need to infer the state at every time period. Use a standard Gaussian prior for each parameter $\theta_i \sim \mathcal{N}(0, 1)$. **Answer the following questions for each of the data sets you have created above.**

1. What is the likelihood model? Please describe how you determine this.
2. What is the form of the posterior?
3. How did you tune your proposal? Think carefully about what a good initial point and initial covariance could be?
4. Analyze your results using the same deliverables as you used in Section 1.
5. Plot the true parameters on your plots of the marginals for reference.

What is the difference between these two models? How does non-identifiability affect the Bayesian approach? In many models it may not be clear by inspection that certain parameters are non-identifiable. How can you use the Bayesian approach to probe whether this might be the case?

4 Extra Credit

Find COVID-19 or any other disease data, and try to fit the SIR model for some time period.

1. Document where you got the data, and how you "cleaned it". Describe the dataset.
2. After doing inference and some period of time, provide a posterior predictive showing its prediction over that period of time, and slightly in the future.
3. Hint: Many papers have done this, so perhaps try to reproduce a result that you found.