Bayesian Statistics

Exercises for week 03

1 Bayesian inference with a grid

4 pt(s)

There are many different distributions. For this exercise, we'll consider the *Poisson distribution*. It describes how often some rare event occurred within a specified time frame (e.g. the number of days of rain in the Sahara dessert, in the span of a year). The Poisson distribution assigns a probability to an observation k given parameter λ according to the following probability distribution:

$$p(k|\lambda) = \text{Poisson}(k|\lambda) = \frac{e^{-\lambda}\lambda^k}{k!} \quad , \tag{1}$$

with k the number of events and λ the expected number of these occurrences. k is observed, λ is a parameter that we wish to learn, using that data.

1. If we want to learn about λ, the unknown parameter, should we use the Poisson distribution as a likelihood or a prior?
2. Implement the Poisson probability distribution in R and use it to make three plots: the first shows the probability distribution for $\lambda = 4$, the second for $\lambda = 7$ and the third for $\lambda = 10$. Plot these distributions for a range of possible observed values, e.g. $k = 0, \ldots, 20$. (Note that we are only considering the case where we have one data point k.)

Of course, there exists a probability distribution that is *conjugate* to the Poisson distribution. This one is known as the *Gamma* distribution. It has this form:

$$p(\lambda|\alpha, \beta) = \text{Gamma}(\lambda|\alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)}\lambda^{\alpha-1}e^{-\beta\lambda} \quad , \tag{2}$$

with two *hyperparameters*, α and β.

3 Implement the Gamma distribution[1] and plot it for $\lambda = 0, \ldots, 10$, and try a number of different values for α and β. Make sure you indicate which values you used. What do these hyperparameters mean? You may use the conjugate prior list on Wikipedia or other online sources for help!

4 Let's fix $\alpha = 7.5$, $\beta = 1$ and say we observe $k = 7$. Compute the posterior distribution 'manually'; for each value $\lambda = 0, \ldots, 20$, compute likelihood $p(k|\lambda)$ times prior $p(\lambda|\alpha, \beta)$ and normalize each of the outcomes by the evidence $\sum_i p(x|\lambda_i)p(\lambda_i|\alpha, \beta)$. Your result should be a plot with λ on the x-axis and the *posterior* probability of λ on the y-axis.

5 Now, compute the posterior using the conjugacy property. That means that you only need to compute the Gamma distribution, but you have to figure out which parameter values you should give it. Tip: look in the Wikipedia conjugate prior list again, and recall that we have one ($n = 1$) observation, $k = 7$.

Show the code you used to do your computations. **Do not use `R`'s prebuild distributions for these functions.**

Note: Typically, we would use the Poisson distribution with larger values for k and λ. However, then we need to ensure that we do not run into problems with the digital representation of very small and very large numbers (due to the factorial and the gamma function). To avoid this for now, this exercise uses only small values.

[1] Note that $\Gamma(n) = (n-1)!$.

2 Multimodal distributions

3 pt(s)

Markov chain Monte Carlo sampling can be a great tool, but it should be used with some caution. Here we explore how *multimodal* distributions can make life difficult. A multimodal distribution is a probability distribution that has multiple, distinct peaks (i.e. multiple values are likely). If the random walk is exploring one peak ('mode'), it can be difficult to reach the next peak since that requires moving through a region of low probability.

First, we construct a (prior, in this case) distribution that has multiple areas of high probability. We define:

$$p(\theta) = \frac{2(cos(4\pi\theta) + 1)^2}{3} \quad . \tag{3}$$

1. Make a plot of this prior distribution as a function of θ on the interval $[0, 1]$.
2. A Metropolis MCMC sampler for a Bernoulli likelihood is available in `BernMetrop.R`. Change the specification of the prior to (3).
3. Initially we will just explore the prior, so set `myData=c()`, so there is no contribution from the likelihood. Furthermore, change the initial value of the sampler to a *uniform random* sample, and comment out the line `set.seed(47405)`. This way, each time you run the script, you run a different sampling chain. Finally, set the standard deviation of the proposal density (`proposalSD`) to 0.2 and run the script a few times. Show the resulting estimated distributions. Are they similar to the previous plot of the prior?
4. Now change `proposalSD` to 0.02 and run the script for different initial conditions $\theta^{(1)}$, i.e. try e.g. a low value, a high value and an intermediate one. How does this affect the resulting approximated distribution?
5. Set the data term back to `myData=c(rep(0,6), rep(1,14)` and repeat the same experiment. Does the problem go away when we have added data? Why or why not?

3 Making a Gibbs sampler

3 pt(s)

Gibbs sampling can be a useful tool if you have more than one parameter of interest, in which case we have a *multivariate* distribution. A particularly common multivariate distribution is the multivariate Gaussian distribution. In this exercise, you'll construct a Gibbs sampler for a special case of this distribution: the bivariate Gaussian (see Fig. 1). For this exercise, we will not be using any observations.

Our target distribution is has this form:

$$p(\mathbf{x}) = \text{Normal}\left(\mathbf{x} \mid \boldsymbol{\mu}, \boldsymbol{\Sigma}\right) \quad . \tag{4}$$

You'll notice that some symbols are in **boldface**. These are vectors (small characters) and matrices (capital letters). To be precise, $\mathbf{x} = [x_1, x_2]$ is a vector containing the two variables we are interested in[2], x_1 and x_2. The vector $\boldsymbol{\mu} = [\mu_1, \mu_2] = [0, 0]$ is the vector of *means* of the distribution. To make life easier on ourselves, we set them both to zero for this exercise. The interaction between x_1 and x_2 is captured in the variance-covariance matrix $\boldsymbol{\Sigma}$, which has the following form:

$$\boldsymbol{\Sigma} = \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix} \quad , \tag{5}$$

and the following interpretation: x_1 and x_2 each have a *variance* of 1, and *covary* with strength ρ. You can see examples of this distribution for different values of ρ in Fig. 1. Note that if $\rho = 0$, the x_1 and x_2 do not covary (they are independent).

[2]To be exact, we're interested in the joint distribution $p(x_1, x_2)$.

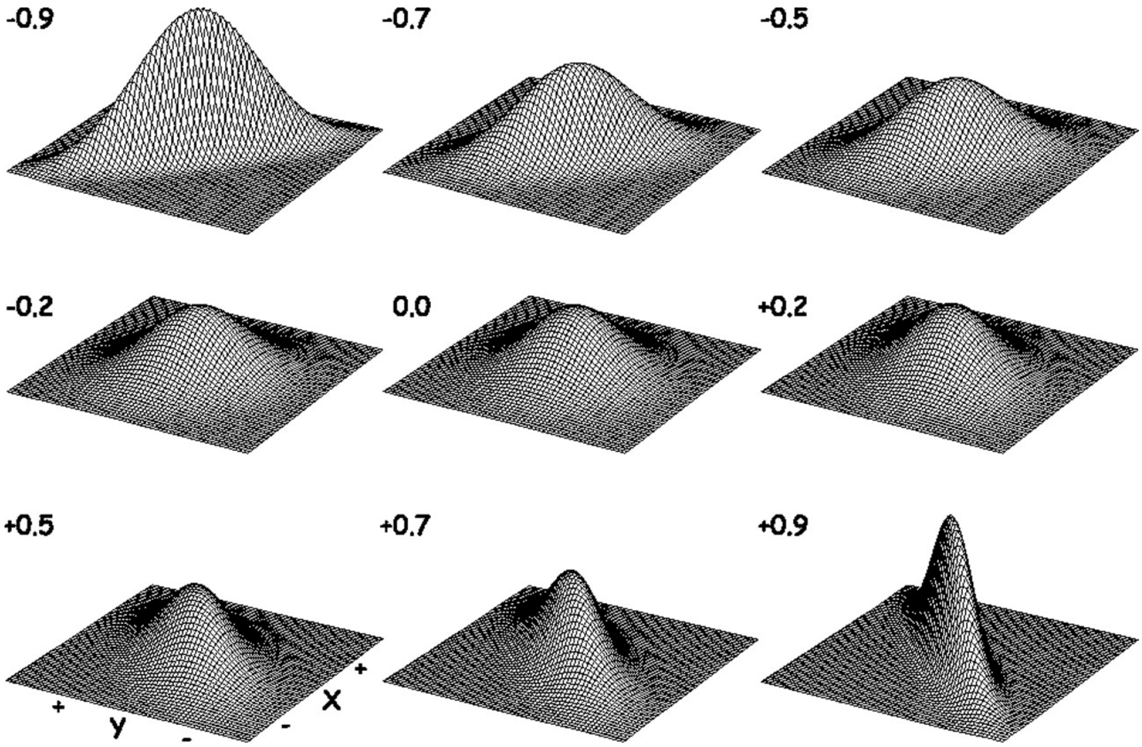

Figure 1: The bivariate Gaussian distribution, with different amounts of correlation between the two variables.

Given these parameters, we know exactly what shape the distribution should have. But for now we'll pretend this is a complex distribution of which we do not know the shape. We'll recover this shape using Gibbs sampling. Gibbs sampling requires the conditional distributions $p(x_1|x_2)$ and $p(x_2|x_1)$. Because I am in a good mood, I'll give you these for free:

$$p(x_1|x_2) = \text{Normal}(x_1|\rho x_2, \sqrt{1-\rho^2}) \qquad \text{and} \qquad p(x_2|x_1) = \text{Normal}(x_2|\rho x_1, \sqrt{1-\rho^2}) \ . \tag{6}$$

Note that these are both univariate distributions, and can be easily sampled from using any statistics software (e.g. using the `rnorm` function in `R`)[3].

With these conditional distributions, you can now complete the main loop in the script `exercise03_gibbs.R`. Read the comments in the script for additional hints.

Show the resulting scatter plot for $\rho = \{0.2, 0.9\}$ using 500 and 5000 iterations.

Congratulations, you have now constructed your first sampler!

[3]And note that the exact formula of the univariate Normal distribution is $p(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$.