Grid approximation and posterior sampling

Math 315, Adam Loy

Example (Exercise 1.11 from BSM): In a study the uses Bayesian methods to forecast the number of species that will be discovered in future years, Edie, Smits & Jablonski (2017), report that the number of marine bivalve species discovered each year from 2010-2015 was 64, 13, 33, 18, 30, and 20. Denoting Y_t as the number of species in year t and assuming $Y_t|\lambda \overset{\text{iid}}{\sim} \text{Poisson}(\lambda)$ and $\lambda \sim \text{Uniform}(0, 100)$ (a continuous uniform), plot the posterior distribution of λ.

First, define a vector for the data

```r
y <- c(64, 13, 33, 18, 30, 20)
```

Grid approximation

Follow these steps to approximate the posterior via grid approximation:

(1) Create a grid of possible values for λ.

```r
lambda <- seq(0, 100, by = .01)
```

(2) Evaluate the prior distribution at each value of `lambda`.

```r
prior <- dunif(lambda, min = 0, max = 100)
```

Note: The distribution functions in R are vectorized. In other words, when you run the above code chunk, the uniform density will be evaluated at each value in `lambda` with a single call.

(3) Evaluate the likelihood at each value of `lambda`. As seen in the below code chunk, we often work with the log likelihood first for computational stability and then exponentiate to obtain the likelihood.

```r
# initialize the log_likelihood object
log_likelihood <- 0

# calculate the log likelihood at each value of lambda
for(t in seq_along(y)) {
  log_likelihood <- log_likelihood + dpois(y[t], lambda = lambda, log = TRUE)
}

# exponentiate to obtain the likelihood
likelihood <- exp(log_likelihood)
```

(4) Calculate the posterior probabilities in the usual way.

```r
posterior <- prior * likelihood / sum(prior * likelihood)
```

Plotting the posterior

Now that you have a grid-approximate posterior, you can plot the posterior probabilities vs. the grid for λ to visualize the distribution:

```r
plot(x = lambda, y = posterior, type = "l", xlab = expression(lambda))
```

MAP estimator

If you have a grid-approximate posterior, then to calculate the MAP estimate you simply find the value of λ corresponding to the maximum value of the posterior density.

```
lambda[which.max(posterior)]
```

Sampling from the posterior

To draw a Monte Carlo sample from your grid-approximate posterior, use the `sample()` command. The below code chunk draws a sample from `lambda` of `size` 10000, with replacement, and each lambda has its `posterior` probability of being drawn.

```
lambda_draws <- sample(lambda, size = 10000, replace = TRUE, prob = posterior)
```

Note: You use the `r*()` functions to draw random samples from a parametric distribution in R. For example, we would use the `rbeta()` function to draw a random sample from a Beta distribution. This will be useful if you derive a posterior distribution by hand and recognize the distributional form.

Once you have a sample from the posterior distribution, you can obtain estimates of λ:

- Posterior mean

  ```
  mean(lambda_draws)
  ```

- Posterior median

  ```
  median(lambda_draws)
  ```

- MAP estimate

  If you have a Monte Carlo sample from your posterior you can calculate the MAP estimate by first approximating the density, and then finding which value of λ maximizes that density.

  ```
  approx_density <- density(lambda_draws)
  approx_density$x[which.max(approx_density$y)]
  ```

You can also easily calculate a 95% credible interval

```
quantile(lambda_draws, probs = c(0.025, 0.975))
```

or 95% HDPI

```
library(HDInterval)
hdi(lambda_draws, credMass = 0.95)
```

To calculate posterior probabilities, use logical operators and the `mean()` function. For example

$P(\lambda < 34)$

```
mean(lambda_draws < 34)
```

$P(\lambda > 34)$

```
mean(lambda_draws > 34)
```

$P(25 < \lambda < 34)$

```
mean(lambda_draws > 25 & lambda_draws < 34)
```