Lesson 3: Monte Carlo estimation

Lesson 3.1

Simulation and CLT

Before we learn how to simulate from complicated posterior distributions, let's review some of the basics of Monte Carlo estimation. Monte Carlo estimation refers to simulating hypothetical draws from a probability distribution in order to calculate important quantities. These quantities might include the mean, the variance, the probability of some event, or quantiles of the distribution. All of these calculations involve integration, which except for the simplest distributions, can be very difficult or impossible.

Suppose we have a random variable θ that follows a $\text{Gamma}(a, b$ distribution. Let's say $a = 2$ and $b = 1/3$, where b is the rate parameter. To calculate the mean of this distribution, we would need to compute the following integral

$$\text{E}(\theta) = \int_0^\infty \theta f(\theta) d\theta = \int_0^\infty \theta \frac{b^a}{\Gamma(a)} \theta^{a-1} e^{-b\theta} d\theta \, .$$

It is possible to compute this integral, and the answer is a/b (6 in this case). However, we could verify this answer through Monte Carlo estimation. To do so, we would simulate a large number of draws (call them θ_i^* for $i = 1, \ldots, m$) from this gamma distribution and calculate their sample mean.

Why can we do this? Recall from the previous course that if we have a random sample from a distribution, the average of those samples converges in probability to the true mean of that distribution by the Law of Large Numbers. Furthermore, by the Central Limit Theorem (CLT), this sample mean $\bar{\theta}^* = \frac{1}{m} \sum_{i=1}^m \theta_i^*$ approximately follows a normal distribution with mean $\text{E}(\theta)$ and variance $\text{Var}(\theta)/m$. The theoretical variance of θ is the following integral:

$$\text{Var}(\theta) = \int_0^\infty (\theta - E(\theta))^2 f(\theta) d\theta \, .$$

Just like we did with the mean, we could approximate this variance with the sample variance $\frac{1}{m} \sum_{i=1}^m (\theta_i^* - \bar{\theta}^*)^2$.

Calculating probabilities

This method can be used to calculate many different integrals. Say $h(\theta)$ is any function and we want to calculate $\int h(\theta) p(\theta) d\theta$. This integral is precisely what is meant by $\text{E}(h(\theta))$, so we can conveniently approximate it by taking the sample mean of $h(\theta_i^*)$. That is, we apply the function h to each simulated sample from the distribution, and take the average of all the results.

One extremely useful example of an h function is is the indicator $I_A(\theta)$ where A is some logical condition about the value of θ. To demonstrate, suppose $h(\theta) = I_{[\theta<5]}(\theta)$, which will give a 1 if $\theta < 5$ and return a 0 otherwise. What is $E(h(\theta))$? This is the integral

$$\int_0^\infty I_{[\theta<5]}(\theta) p(\theta) d\theta = \int_0^5 1 \cdot p(\theta) d\theta + \int_5^\infty 0 \cdot p(\theta) d\theta = P(\theta < 5) \, .$$

So what does this mean? It means we can approximate the probability that $\theta < 5$ by drawing many samples θ_i^*, and approximating this integral with $\frac{1}{m}\sum_{i=1}^{m} I_{\theta^*<5}(\theta_i^*)$. This expression is simply counting how many of those samples come out to be less than 5, and dividing by the total number of simulated samples. That's convenient!

Likewise, we can approximate quantiles of a distribution. If we are looking for the value z such that $P(\theta < z) = 0.9$, we simply arrange the samples θ_i^* in ascending order and find the smallest drawn value that is greater than 90% of the others.

Lesson 3.2

Monte Carlo error

How good is an approximation by Monte Carlo sampling? Again we can turn to the CLT, which tells us that the variance of our estimate is controlled in part by m. For a better estimate, we want larger m.

For example, if we seek $\mathrm{E}(\theta)$, then the sample mean $\bar{\theta}^*$ approximately follows a normal distribution with mean $\mathrm{E}(\theta)$ and variance $\mathrm{Var}(\theta)/m$. The variance tells us how far our estimate might be from the true value. One way to approximate $\mathrm{Var}(\theta)$ is to replace it with the sample variance. The standard deviation of our Monte Carlo estimate is the square root of that, or the sample standard deviation divided by $\sqrt{m}$. If m is large, it is reasonable to assume that the true value will likely be within about two standard deviations of your Monte Carlo estimate.

Marginalization

We can also obtain Monte Carlo samples from hierarchical models. As a simple example, let's consider a binomial random variable where $y \mid \phi \sim \mathrm{Bin}(10, \phi)$, and further suppose ϕ is random (as if it had a prior) and is distributed beta $\phi \sim \mathrm{Beta}(2, 2)$. Given any hierarchical model, we can always write the joint distribution of y and ϕ as $p(y, \phi) = p(y \mid \phi)p(\phi)$ using the chain rule of probability. To simulate from this joint distribution, repeat these steps for a large number m:

1. Simulate ϕ_i^* from its $\mathrm{Beta}(2, 2)$ distribution.
2. Given the drawn ϕ_i^*, simulate y_i^* from $\mathrm{Bin}(10, \phi_i^*)$.

This will produce m independent pairs of $(y^*, \phi^*)_i$ drawn from their joint distribution. One major advantage of Monte Carlo simulation is that marginalizing is easy. Calculating the marginal distribution of y, $p(y) = \int_0^1 p(y, \phi)d\phi$, might be challenging. But if we have draws from the joint distribution, we can just discard the ϕ_i^* draws and use the y_i^* as samples from their marginal distribution. This is also called the prior predictive distribution introduced in the previous course.

In the next segment, we will demonstrate some of these principles. Remember, we do not yet know how to sample from the complicated posterior distributions introduced in the previous lesson. But once we learn that, we will be able to use the principles from this lesson to make approximate inferences from those posterior distributions.

Lesson 3.3

Example from Lesson 3.1

Monte Carlo simulation from the most common distributions is very straightforward in `R`.

Let's start with the example from the previous segment, where $\theta \sim \text{Gamma}(a, b)$ with $a = 2, b = 1/3$. This could represent the posterior distribution of θ if our data came from a Poisson distribution with mean θ and we had used a conjugate gamma prior. Let's start with $m = 100$.

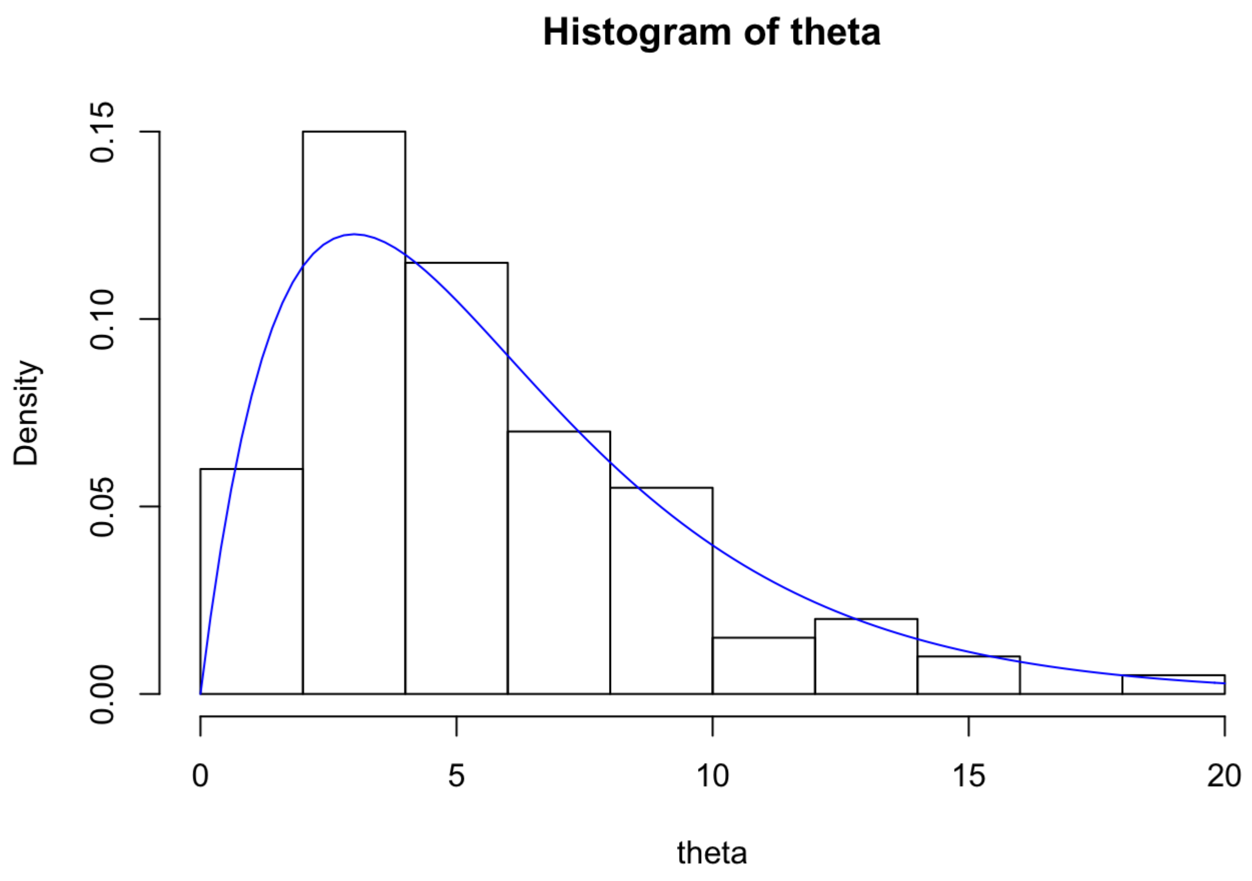
```
set.seed(32) # Initializes the random number generator so we can replicate these results. To get different random numbers, change the seed.
m = 100
a = 2.0
b = 1.0 / 3.0
```

To simulate m independent samples, use the `rgamma` function.

```
theta = rgamma(n=m, shape=a, rate=b)
```

We can plot a histogram of the generated data, and compare that to the true density.

```
hist(theta, freq=FALSE)
curve(dgamma(x=x, shape=a, rate=b), col="blue", add=TRUE)
```



To find our Monte Carlo approximation to $\text{E}(\theta)$, let's take the average of our sample (and compare it with the truth).

```
sum(theta) / m # sample mean
```

```
## [1] 5.514068
```

```
mean(theta) # sample mean
```

```
## [1] 5.514068
```

```
a / b # true expected value
```

```
## [1] 6
```

Not bad, but we can do better if we increase m to say, 10,000.

```
m = 1e4
theta = rgamma(n=m, shape=a, rate=b)
mean(theta)
```

```
## [1] 6.023273
```

How about the variance of θ?

```
var(theta) # sample variance
```

```
## [1] 18.04318
```

```
a / b^2 # true variance of Gamma(a,b)
```

```
## [1] 18
```

We can also approximate the probability that $\theta < 5$.

```
ind = theta < 5.0 # set of indicators, TRUE if theta_i < 5
mean(ind) # automatically converts FALSE/TRUE to 0/1
```

```
## [1] 0.497
```

```
pgamma(q=5.0, shape=a, rate=b) # true value of Pr( theta < 5 )
```

```
## [1] 0.4963317
```

What is the 0.9 quantile (90th percentile) of θ? We can use the `quantile` function which will order the samples for us and find the appropriate sample quantile.

```
quantile(x=theta, probs=0.9)
```

```
##      90% 
## 11.74338
```

```
qgamma(p=0.9, shape=a, rate=b) # true value of 0.9 quantile
```

```
## [1] 11.66916
```

Lesson 3.4

Monte Carlo error

We can use the CLT to approximate how accurate our Monte Carlo estimates are. For example, if we seek $\mathrm{E}(\theta)$, then the sample mean $\bar{\theta}^*$ approximately follows a normal distribution with mean $\mathrm{E}(\theta)$ and variance $\mathrm{Var}(\theta)/m$. We will use the sample standard deviation divided by the square root of m to approximate the Monte Carlo standard deviation.

```
se = sd(theta) / sqrt(m)
2.0 * se # we are reasonably confident that the Monte Carlo estimate is no more than this far fr
om the truth
```

```
## [1] 0.08495454
```

These numbers give us a reasonable range for the quantity we are estimating with Monte Carlo. The same applies for other Monte Carlo estimates, like the probability that $\theta < 5$.

```
ind = theta < 5.0
se = sd(ind) / sqrt(m)
2.0 * se # we are reasonably confident that the Monte Carlo estimate is no more than this far fr
om the truth
```

```
## [1] 0.01000032
```

Marginalization

Let's also do the second example of simulating a hierarchical model. In our example from the previous segment, we had a binomial random variable where $y \mid \phi \overset{\text{iid}}{\sim} \mathrm{Bin}(10, \phi)$, and $\phi \sim \mathrm{Beta}(2, 2)$. To simulate from this joint distribution, repeat these steps for a large number m:

1. Simulate ϕ_i from its $\mathrm{Beta}(2, 2)$ distribution.
2. Given the drawn ϕ_i, simulate y_i from $\mathrm{Bin}(10, \phi_i)$.

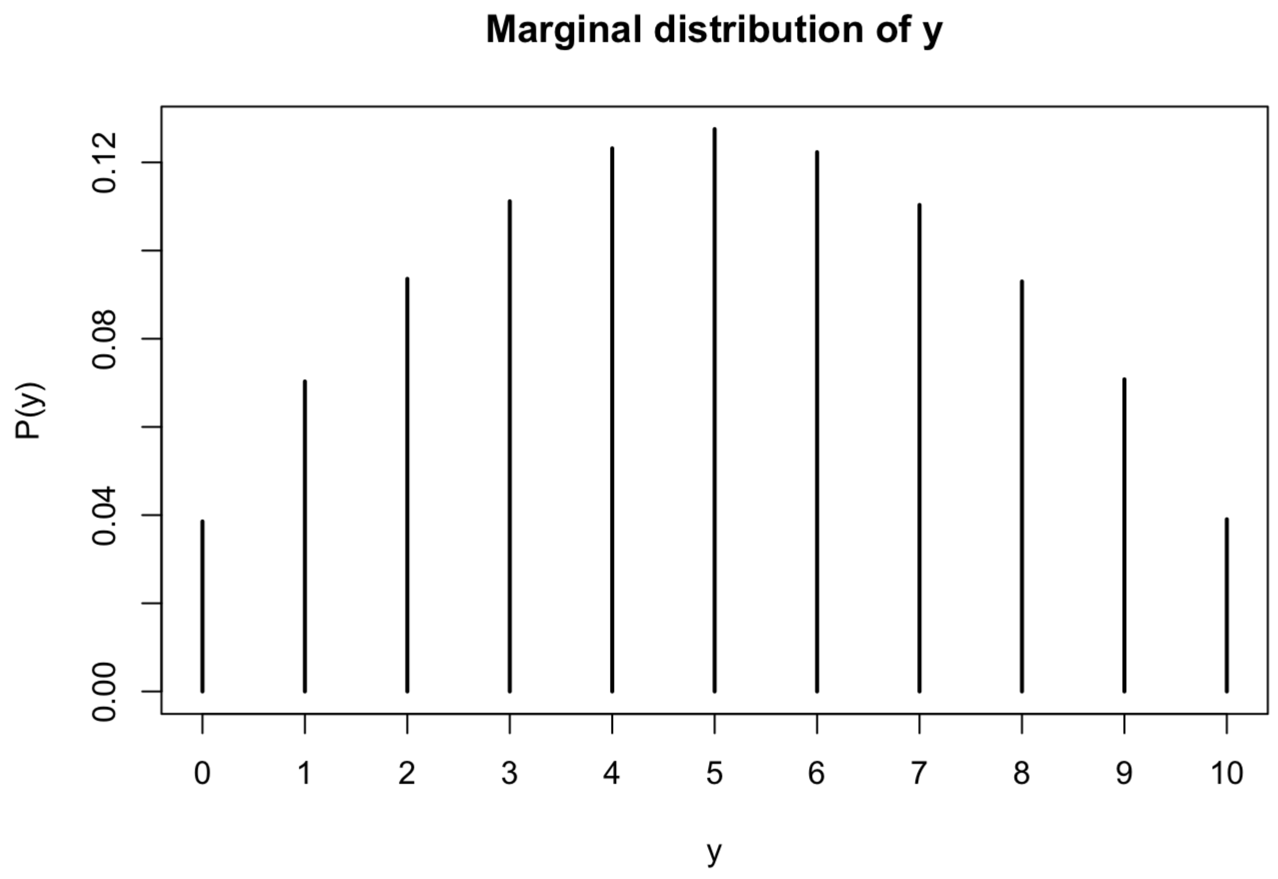
```
m = 10e4

y = numeric(m) # create the vectors we will fill in with simulations
phi = numeric(m)

for (i in 1:m) {
  phi[i] = rbeta(n=1, shape1=2.0, shape2=2.0)
  y[i] = rbinom(n=1, size=10, prob=phi[i])
}
# which is equivalent to the following 'vectorized' code
phi = rbeta(n=m, shape1=2.0, shape2=2.0)
y = rbinom(n=m, size=10, prob=phi)
```

If we are interested only in the marginal distribution of y, we can just ignore the draws for ϕ and treat the draws of y as a sample from its marginal distribution.

```
mean(y)
```

```
## [1] 5.00008
```

```
plot(prop.table(table(y)), ylab="P(y)", main="Marginal distribution of y")
```


Marginal distribution of y
P(y)
y