Lecture 4 - Key

Posterior Inference

Example. Consider a binomial model where we have a posterior distribution for the probability term, θ. Suppose the research goal is to make inferences on the log-odds $\gamma = \log \frac{\theta}{1-\theta}$.

How would you do this in a classical framework?

From a Bayesian perspective consider the following procedure:

Example. Consider making comparisons between two properties of a distribution. For example a simple contrast, $\gamma = \theta_1 - \theta_2$, or a more complicated function of θ_1, and θ_2 such as $\gamma = \log(\frac{\theta_1}{\theta_2})$.

Posterior Predictive Distribution

Recall the posterior predictive distribution $p(y^*|y_1, ..., y_n)$ is the predictive distribution for an upcoming data point given the observed data.

How does the parameter θ (mathematically) factor into this equation?

Often the predictive distribution is hard to sample from, so a two-step procedure is completed instead.

Similar ideas extend to posterior predictive model checking, which we will return to after studying Bayesian regression.

These procedures are both examples of Monte Carlo procedures to approximate integrals.

Monte Carlo Procedures

Monte Carlo procedures use random sampling to estimate mathematical or statistical quantities. These computational algorithms are defined by running for a fixed time (number of samples/iterations) and result in a random estimate.

There are three main uses for Monte Carlo procedures:

Monte Carlo methods were introduced by John von Neumann and Stanislaw Ulam at Los Alamos. The name Monte Carlo was a code name referring the Monte Carlo casino in Monaco. Monte Carlo methods were central to the Manhattan project and continued development of physics research related to the hydrogen bomb.

An essential part of many scientific problems is the computation of the integral, $I = \int_{\mathcal{D}} g(x)dx$ where $\mathcal{D}$ is often a region in a high-dimensional space and $g(x)$ is the target function of interest. If we can draw independent and identically distributed random samples $x_1, x_2, \ldots, x_n$ uniformly from $\mathcal{D}$ (by a computer) an approximation to I can be obtained as

The law of large numbers states that the average of many independent random variables with common mean and finite variances tends to stabilize at their common mean; that is

$$\lim_{n \leftarrow \infty} \hat{I}_n = I, \text{ with probability 1.}$$

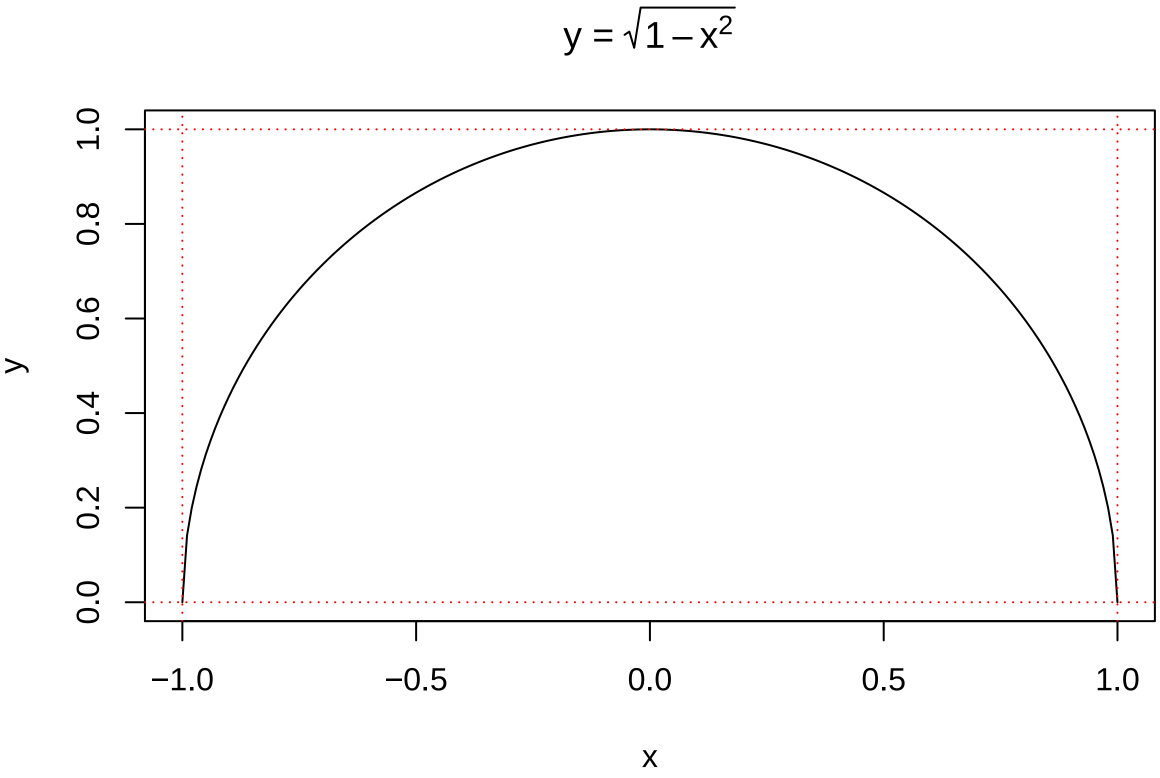
A related procedure that you may be familiar with is the *Riemann approximation.* Consider a case where $\mathcal{D} = [0, 1]$ and $I = \int_0^1 g(x)dx$ then

where $b_i = i/n$. Essentially this method is a grid based evaluation. This works will for a smooth function in low dimension, but quickly runs into the "curse of dimensionality".

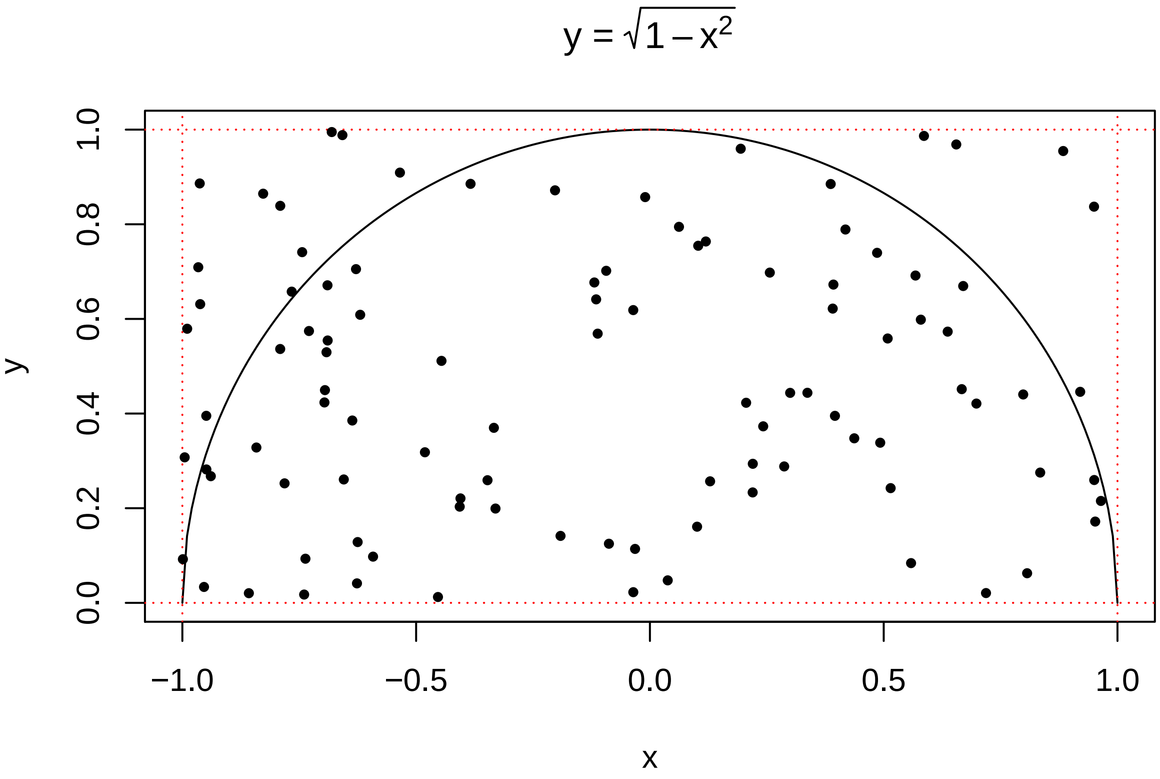
Exercise 1. Consider the function $g(x) = \sqrt{1 - x^2}$, where $x \in [-1, 1]$. The goal is to estimate $I = \int g(x)dx$.

One Monte Carlo approach to solve this problem is to simulate points from a uniform distribution with a known area.

```
x <- seq(-1,1, by=.01)
y <- sqrt(1-x^2)
plot(x,y,type='l', main= expression(paste("y = ", sqrt(1-x^2))))
abline(h=0,lty=3, col = 'red')
abline(h=1,lty=3, col = 'red')
abline(v=1,lty=3, col = 'red')
abline(v=-1,lty=3, col = 'red')
```



```
num_sims <- 100000
x_samples <- runif(num_sims,-1,1)
y_samples <- runif(num_sims)
plot(x,y,type='l', main= expression(paste("y = ", sqrt(1-x^2))))
abline(h=0,lty=3, col = 'red')
abline(h=1,lty=3, col = 'red')
abline(v=1,lty=3, col = 'red')
abline(v=-1,lty=3, col = 'red')
points(x_samples[1:100], y_samples[1:100], pch = 16, cex=.7)
```



```
accept_samples <- mean(y_samples < sqrt(1-x_samples^2))
area <- 2 * accept_samples
```

The resultant value is 1.57398, which is *close* to the analytical solution of $\pi/2 =$ ‘r pi/2.. Is this close enough?

What will the answer be if we do this again?

What happens if we use more samples?

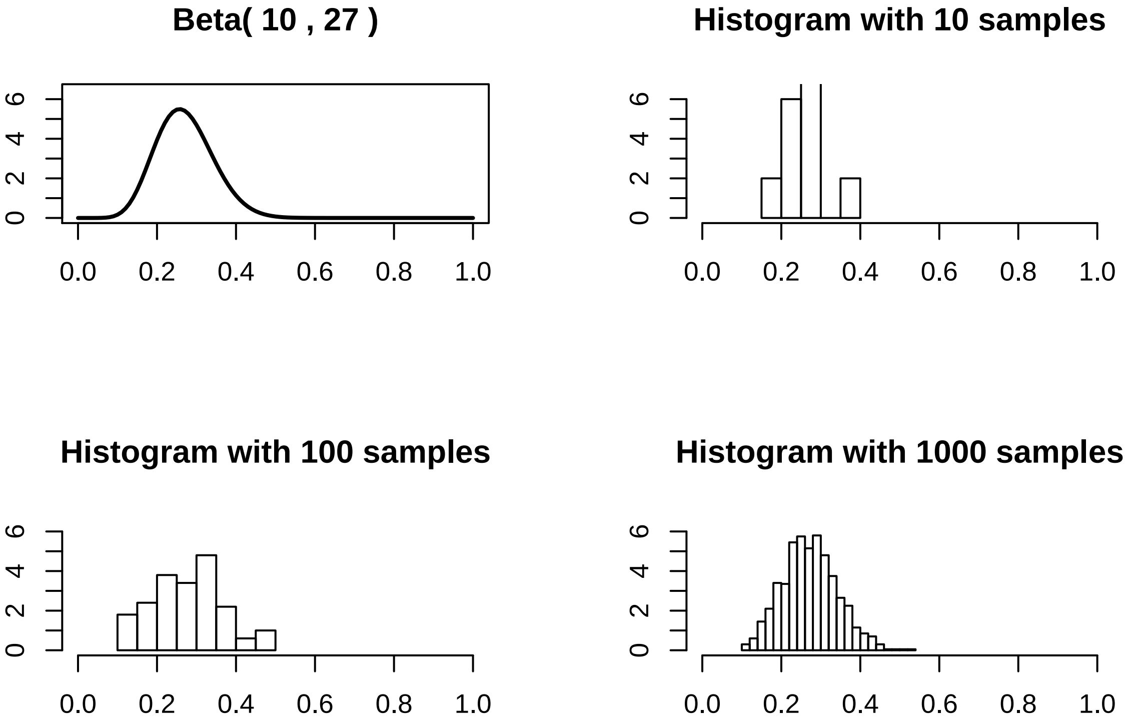
Markov Chain Monte Carlo

In the previous section, we saw that a beta distribution was a conjugate prior for a sampling model, meaning that the posterior was also a beta distribution. This prior specification allows easy posterior computations because

In many situations, this type of prior is not available and we need to use other means to understand the posterior distribution $p(\theta|y)$.

Approximating a Distribution with a Large Sample

Consider taking sample to visualize an entire distribution.



As the sample size gets larger, the histrogram representation begins to look more like the true distribution. Additionally the moments of the sampled distribution approach that of the true distribution.

The true mean is $\frac{a}{a+b} = 0.27$ and quantiles can be computed using `qbeta(.025,a,b)` = 0.142 and `qbeta(.975,a,b)` = 0.422

- with 10 samples: the mean is 0.258 and the quantiles are (0.171, 0.356)

- with 100 samples: the mean is 0.276 and the quantiles are (0.127, 0.458)

- with 1000 samples: the mean is 0.272 and the quantiles are (0.145, 0.418)

The Metropolis Algorithm

In many cases we cannot sample directly from a posterior distribution using for example `rbeta()`, so we need to use Markov Chain Monte Carlo (MCMC).

Politican stumbles across the Metropolis algorithm

DBDA provides a nice intuitive overview of a special kind of an MCMC algorithm called the Metropolis algorithm.

The elected politician lives on a chain of islands and wants to stay in the public eye by traveling from island-to-island.

At the end of day the politician needs to decide whether to:

1. stay on the current island,

2. move to the adjacent island to the west, or

3. move to the adjacent island to the east.

The politician's goal is to visit all islands proportional to their population, so most time is spent on the most populated islands. Unfortunately his office doesn't know the total population of the island chain. When visiting (or proposing to vist) an island, the politician can ask the local mayor for the population of the island.

The politician has a simple heuristic for deciding whether to travel to a proposed island, by flipping a coin to determine whether to consider traveling east or west.

Q: after flipping the coin, what criteria should the politician use to determine whether to visit a neighboring island or stay at the current island? the relative population of the islands is important.

Q: now consider two cases and determine what decision the politician should make:

1. the neighboring island has twice as many residents than the island the politician is currently on.

2. the neighboring island has half as many residents than the island the politician is currently on

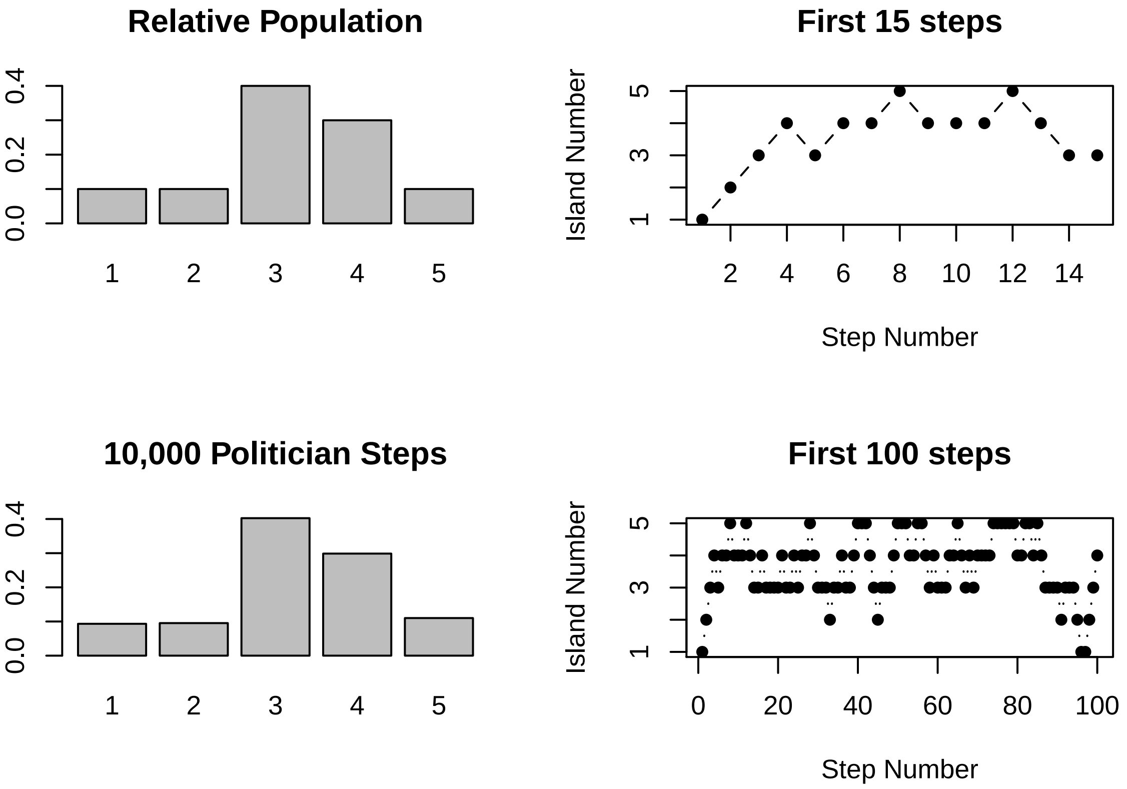
```r
par(mfcol=c(2,2))
# set up islands and relative population
num.islands <- 5
relative.population <- c(.1,.1,.4,.3,.1)
barplot(relative.population, names.arg = as.character(1:5), main='Relative Population')

# initialize politician
num.steps <- 10000
island.location <- rep(1,num.steps) # start at first island

# algorithm
for (i in 2:num.steps){
  direction <- sample(c('right','left'),1)
  if (direction == 'right'){
    proposed.island <- island.location[i-1] + 1
    if (proposed.island == 6) {
      island.location[i] <- island.location[i-1] #no island 6 exists, stay at island 5
    } else {
      prob.move <- relative.population[proposed.island] / relative.population[island.location[i-1]]
      if (runif(1) < prob.move){
        # move
        island.location[i] <- proposed.island
      } else{
        #stay
        island.location[i] <- island.location[i-1]
      }
    }
  }
  if (direction == 'left'){
    proposed.island <- island.location[i-1] - 1
    if (proposed.island == 0) {
      island.location[i] <- island.location[i-1] #no island 0 exists, stay at island 1
    } else {
      prob.move <- relative.population[proposed.island] / relative.population[island.location[i-1]]
      if (runif(1) < prob.move){
        # move
        island.location[i] <- proposed.island
      } else{
        #stay
        island.location[i] <- island.location[i-1]
      }
    }
  }
}
barplot(table(island.location) / num.steps, names.arg = as.character(1:5),
        main='10,000 Politician Steps')

plot(island.location[1:15], ylim=c(1,5),ylab='Island Number',xlab='Step Number',
     pch=16,type='b', main='First 15 steps')

plot(island.location[1:100], ylim=c(1,5),ylab='Island Number',xlab='Step Number',
     pch=16,type='b', main='First 100 steps')
```



More details about the Markov chain

This procedure that we have described is a Markov chain. As such we can consider a few probabilities, let $l(i)$ be the politician's location at time i and assume the politician begins at island 5.:

- $Pr[l(1) = 5] = 1$
- $Pr[l(2) = 1] = 0$
- $Pr[l(2) = 2] = 0$
- $Pr[l(2) = 3] = 0$
- $Pr[l(2) = 4] = \frac{1}{2}$
- $Pr[l(2) = 5] = \frac{1}{2}$

We can also think about transition probabilities from state i to state j

Table 1: transition probabilities for 5 island traveling politician example

current.state	to.1	to.2	to.3	to.4	to.5
at.1	0.5	0.500	0.0	0.000	0.000
at.2	0.5	0.000	0.5	0.000	0.000
at.3	0.0	0.125	0.5	0.375	0.000
at.4	0.0	0.000	0.5	0.333	0.167
at.5	0.0	0.000	0.0	0.500	0.500

More details about Metropolis

The process to determine the next location to propose is known as the *proposal distribution*.

The key elements of this process are:

1. Generate a random value from the proposal distribution to create $\theta_{proposed}$.

2. Evaluate the the target distribution to compute $\frac{\theta_{proposed}}{\theta_{current}}$.

3. Generate a random variable from uniform distribution to accept or reject proposal according to $p_{move} = \min\left(\frac{\theta_{proposed}}{\theta_{current}}, 1\right)$.

The ability to complete these three steps allows indirect sampling from the target distribution, even if it cannot be done directly (viz. `rnorm()`).

Generally our target distribution will be the posterior distribution, $p(\theta|\mathcal{D})$.

Furthermore, this process does not require a normalized distribution, which will mean we don't have to compute $\mathcal{D}$ in the denominator of Bayes rule as it will be the same for any θ and θ'. Hence evaluating the target distribution will amount to evaluating $p(\mathcal{D}|\theta) \times p(\theta)$.

The traveling politician example has:

- discrete positions (islands),
- one dimension (east-west),
- and a proposal distribution of one step east or west.

This procedure also works more generally for:

- continuous values (consider the probability parameter in occupancy model),
- any number of dimensions,
- and more general proposal distributions.