first-rmarkdown

Shozen Dan

2024-09-27

Contents

1 Objective

The objective of this tutorial is to introduce you to R Studio and R Markdown along with some basic LaTeX and R. As a simple example, I will attempt to implement a random sampler for the general Gaussian distribution from scratch and document the process.

2 Background

2.1 Probability density function

The probability density function (p.d.f.) of the univariate Gaussian distribution is given as

$$f(x;\mu,\sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2}\frac{(x-\mu)^2}{\sigma^2}\right) \tag{1}$$

where μ and σ denote the mean and standard deviation, respectively.

2.2 Cumulative distribution function

The cumulative distribution function (c.d.f.) associated with (1) is given as

$$\begin{aligned} F(x;\mu,\sigma) &= \int_{-\infty}^{x} f(t;\mu,\sigma)dt \\ &= \frac{1}{2}\left[1 + \text{erf}\left(\frac{x-\mu}{\sigma\sqrt{2}}\right)\right] \end{aligned} \tag{2}$$

where erf is the error function:

$$\mathrm{erf}(z) = \frac{2}{\sqrt{\pi}} \int_0^z e^{-t^2} dt.$$

3 Random sampler for the Gaussian

In this section, we implement a random sampler for the Gaussian distribution from scratch. To simplify the problem, we shall take the following two step approach:

1. Implement a random sampler for the standard Gaussian distribution
2. Use the sampler from the previous step to implement a sampler for the general Gaussian distribution.

3.1 Box-Muller transform

A basic method for pseudo-random number sampling is the inverse transform sampling. However, this requires us to derive the inverse of the c.d.f. (2) which does not have a closed form expression. Instead, we will use a trick called the Box-Muller transform (Box and Muller 1958).

Let U_1 and U_2 denote random variables which follow a uniform distribution on the unit interval $(0, 1)$. Define X_1 and X_2 to be

$$X_1 = \sqrt{-2 \log U_1} \cos(2\pi U_2)$$
$$X_2 = \sqrt{-2 \log U_1} \sin(2\pi U_2),$$

It can be shown that (X_1, X_2) is a pair of random variables from the standard Gaussian.

```r
#' Draw samples from a standard Gaussian
#'
#' @param n number of samples
#'
#' @return a vector of n independent samples from a standard Gaussian
sample_standard_gauss <- function(n) {
  if (n %% 2 == 0) { # If n is even
    n1 <- n/2; n2 <- n/2
  } else {
    n1 <- round(n/2); n2 <- n1 - 1
  }
  U1 <- runif(n1, min = 0, max = 1)
  U2 <- runif(n2, min = 0, max = 1)

  X1 <- sqrt(-2*log(U1))*cos(2*pi*U2)
  X2 <- sqrt(-2*log(U1))*sin(2*pi*U2)

  return(c(X1, X2))
}
```

From Figure 1, we verify that our sampler is indeed sampling from a standard Gaussian distribution.

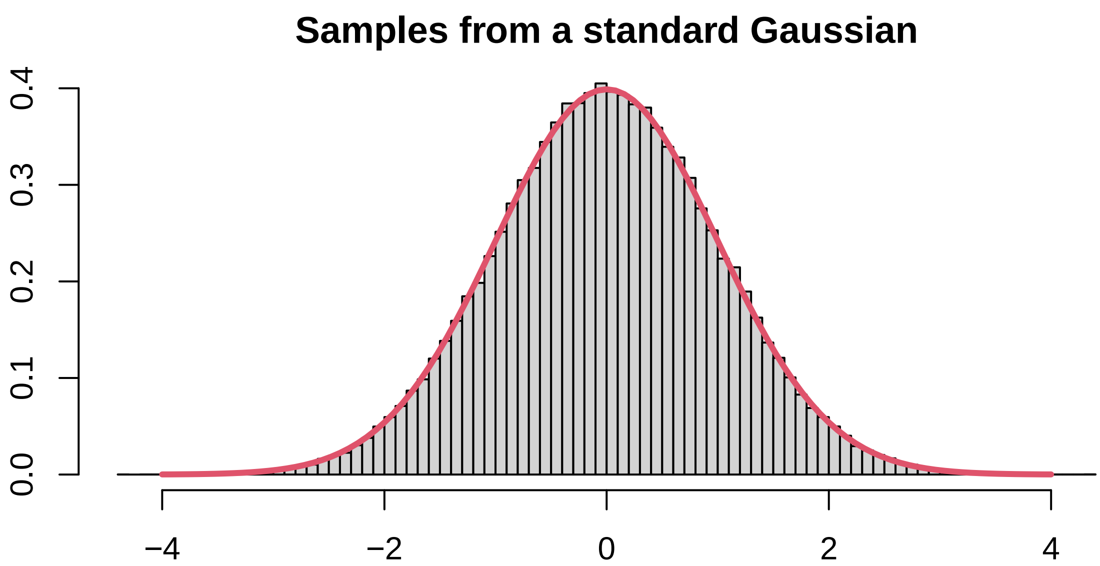

Figure 1: A histogram of samples from the standard Gaussian distribution overlayed with the true density funtion

3.2 Sampling from a General Normal Distribution

Let, Z be a random variable distributed according to a standard Gaussian distribution. Define Z to be,

$$X = \mu + \sigma Z.$$

Then Z will follow a Gaussian distribution with mean μ and variance σ^2.

```r
#' Draw samples from a general normal distribution
#'
#' @param n number of samples
#' @param mean the mean of the distribution
#' @param scale the standard deviation
#'
#' @return a vector containing independent samples from a normal distribution
sample_gauss <- function(n, mean, scale) {
  Z <- sample_standard_gauss(n)
  return(mean + scale*Z)
}
```

Figure 2 shows a histogram of 10000 samples generated by our sampler overlayed with the true density function.

References

Box, G. E. P., and Mervin E. Muller. 1958. "A Note on the Generation of Random Normal Deviates." *The Annals of Mathematical Statistics* 29 (2): 610–11. https://doi.org/10.1214/aoms/1177706645.

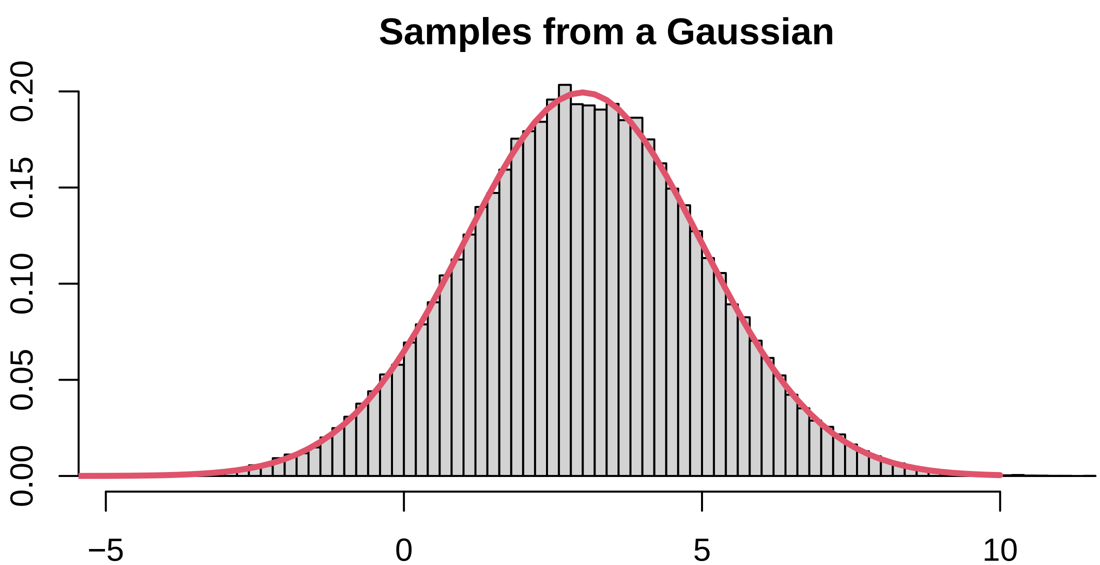

Figure 2: A histogram of samples from the Gaussian distribution with mean 3 and variance 4, overlayed with the true density funtion