Exercise 1

TMA4300 Computer Intensive Statistical Models

Mads Adrian Simonsen, William Scott Grundeland Olsen

07 februar, 2021

Problem A: Stochastic simulation by the probability integral transform and bivariate techniques

1.

Let $X \sim \text{Exp}(\lambda)$, with the cdf

$$F_X(x) = 1 - e^{-\lambda x}, \quad x \geq 0.$$

Then the random variable $Y := F_X(X)$ has a $\text{Uniform}(0, 1)$ distribution. The probability integral transform becomes

$$Y = 1 - e^{-\lambda X} \Leftrightarrow X = -\frac{1}{\lambda}\log(1 - Y). \tag{1}$$

Thus, we sample Y from `runif()` and transform it using (1), to sample from the exponential distribution. Figure 1 shows one million samples drawn from the `generate_from_exp()` function defined in the code chunk below.

```
set.seed(123)

generate_from_exp <- function(n, rate = 1) {
  Y <- runif(n)
  X <- -(1 / rate) * log(1 - Y)
  X
}

# sample
n <- 1000000  # One million samples
lambda <- 4.32
x <- generate_from_exp(n, rate = lambda)

# plot
hist(x,
  breaks      = 80,
  probability = TRUE,
  xlim        = c(0, 2)
)
curve(dexp(x, rate = lambda),
  add = TRUE,
  lwd = 2,
```

```
  col = "red"
)
```

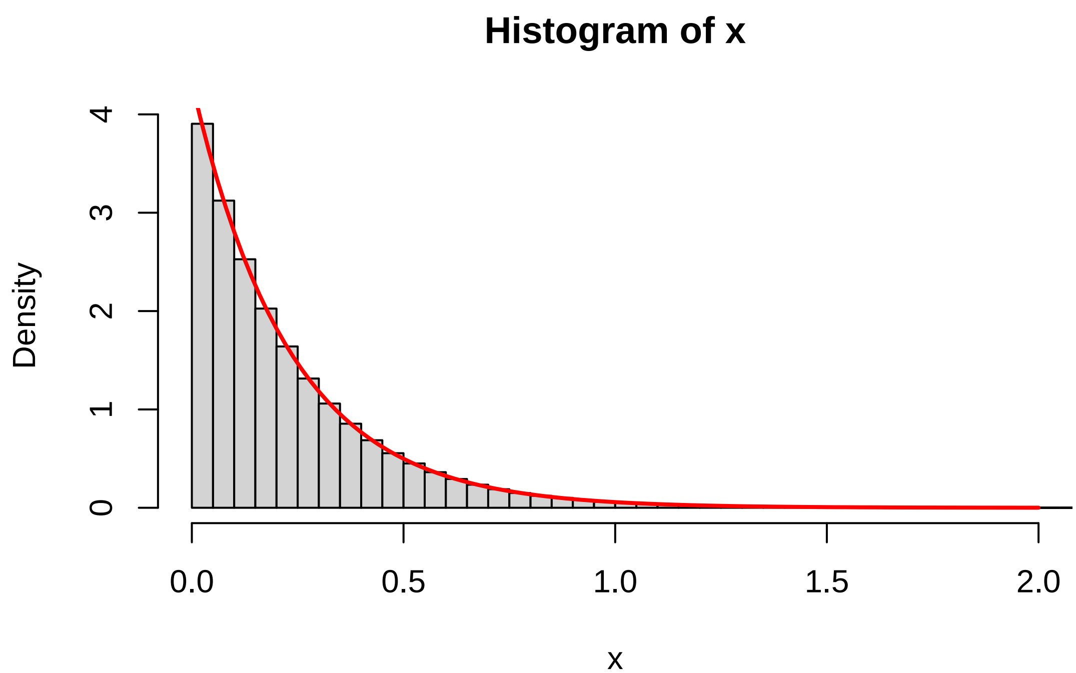

Figure 1: Normalized histogram of one million samples drawn from the exponential distribution, together with the theoretical pdf, with $\lambda = 4.32$.

2.

(a)

(b)

3.

(a)

(b)

(c)

4.

5

Problem B: The gamma distribution

1.

(a)

Let $f(x)$ be the target distribution we wish to sample from, and let $g(x)$ be the proposal distribution. For the rejection sampling algorithm, we require that

$$f(x) \leq c \cdot g(x), \quad \forall x \in \mathbb{R}, \tag{2}$$

for some constant $c > 0$. Let X and U be independent samples where $X \sim g(x)$ and $U \sim \text{Uniform}(0, 1)$. Then the acceptance probability is

$$\begin{aligned}
\Pr\left(U \leq \frac{f(X)}{c \cdot g(X)}\right) &= \int_{-\infty}^{\infty} \int_{0}^{f(x)/(c\ g(x))} f_{X,U}(x, u)\, du\, dx \\
&= \int_{-\infty}^{\infty} \int_{0}^{f(x)/(c\ g(x))} g(x) \cdot 1\, du\, dx \\
&= \int_{-\infty}^{\infty} \frac{f(x)}{c\ g(x)} g(x)\, dx \\
&= \frac{1}{c} \int_{-\infty}^{\infty} f(x)\, dx \\
&= \frac{1}{c}
\end{aligned}$$

We wish to sample from Gamma$(\alpha, \beta = 1)$, using the proposal distribution $g(x)$ given in `eqref{??????}`. We want to choose c such that the acceptance probability is maximized while (2) is satisfied. We must check three cases. The trivial case when $x \leq 0$, we have $f(x) = g(x) = 0$ so (2) is satisfied for all c. When $0 < x < 1$ we have

$$\begin{aligned}
f(x) &\leq c\, g(x) \\
\frac{1}{\Gamma(\alpha)} x^{\alpha-1} e^{-x} &\leq c\, \frac{1}{\alpha^{-1} + e^{-1}} x^{\alpha-1} \\
c &\geq \left(\frac{1}{\alpha} + \frac{1}{e}\right) \frac{1}{\Gamma(\alpha)} e^{-x} \\
c &\geq \left(\frac{1}{\alpha} + \frac{1}{e}\right) \frac{1}{\Gamma(\alpha)}.
\end{aligned}$$

The last case, when $x \geq 1$, we have

$$\begin{aligned} f(x) &\leq c\, g(x) \\ \frac{1}{\Gamma(\alpha)} x^{\alpha-1} e^{-x} &\leq c\, \frac{1}{\alpha^{-1} + e^{-1}} e^{-x} \\ c &\geq \left(\frac{1}{\alpha} + \frac{1}{e}\right) \frac{1}{\Gamma(\alpha)} x^{\alpha-1} \\ c &\geq \left(\frac{1}{\alpha} + \frac{1}{e}\right) \frac{1}{\Gamma(\alpha)}. \end{aligned}$$

That is, we choose $c := (\alpha^{-1} + e^{-1})/\Gamma(\alpha)$, such that the acceptance probability becomes

$$\Pr\left(U \leq \frac{f(X)}{c \cdot g(X)}\right) = \frac{1}{c} = \frac{\Gamma(\alpha)}{\alpha^{-1} + e^{-1}}, \quad \alpha \in (0, 1).$$

(b)

```r
set.seed(137)

sample_from_gamma_rej <- function(n, shape = 0.5) {
  c <- (1 / shape + 1 / exp(1)) / gamma(shape)    # constant that minimizes the envelope
  x <- vector(mode = "numeric", length = n)
  for (i in 1:n) {
    repeat {
      x[i] <- generate_from_gx(1, alpha = shape)  # draw from proposal
      u <- runif(1)                               # draw from U(0, 1)
      f <- dgamma(x[i], shape = shape)            # target value
      g <- theo_gx(x[i], alpha = shape)           # proposal value
      alpha <- (1 / c) * (f / g)
      if (u <= alpha) {
        break
      }
    }
  }
  return(x)
}

# n <- 1000000
# alpha <- 0.9
# x <- sample_from_gamma_rej(n, shape = alpha)
# hist(x,
#   breaks      = 80,
#   probability = TRUE,
#   xlim        = c(0, 6)
# )
# curve(dgamma(x, shape = alpha),
#   add = TRUE,
#   lwd = 2,
#   col = "red"
# )
```

2.

(a)

We will now use the ratio-of-uniforms method to simulate from Gamma$(\alpha, \beta = 1)$. Additionally we have $\alpha > 1$ this time. Let us define

$$C_f = \left\{(x_1, x_2) : 0 \leq x_1 \leq \sqrt{f^*\left(\frac{x_2}{x_1}\right)}\right\}, \quad \text{where} \quad f^*(x) = \begin{cases} x^{\alpha-1}e^{-x}, & x > 0, \\ 0, & \text{otherwise,} \end{cases} \tag{3}$$

and

$$a = \sqrt{\sup_x f^*(x)}, \quad b_+ = \sqrt{\sup_{x\geq 0}(x^2 f^*(x))} \quad \text{and} \quad b_- = -\sqrt{\sup_{x\leq 0}(x^2 f^*(x))}, \tag{4}$$

such that $C_f \subset [0,1] \times [b_-, b_+]$.

First we find $\sup_x f^*(x)$. This must be when $x > 0$. We differentiate $f^*(x)$ and setting the expression equal to zero to find the stationary point.

$$\begin{aligned} 0 &= \frac{d}{dx} f^*(x) \\ &= \frac{d}{dx} x^{\alpha-1} e^{-x} \\ &= e^{-x} x^{\alpha-2}\left((\alpha-1) - x\right) \\ \Rightarrow \quad x &= \alpha - 1, \quad \text{where} \quad \alpha > 1. \end{aligned}$$

Since we have only one stationary point, $f^*(x)$ is continuous, $f^*(x) > 0 \ \forall x > 0$ and $\lim_{x\to 0+} f^*(x) = \lim_{x\to\infty} f^*(x) = 0$, then $x = \alpha - 1$ must be the global maximum point. That is

$$a = \sqrt{f^*(\alpha-1)} = \sqrt{(\alpha-1)^{\alpha-1}e^{-(\alpha-1)}} = \left(\frac{\alpha-1}{e}\right)^{(\alpha-1)/2}. \tag{5}$$

We now wish to find b_+.

$$\begin{aligned} 0 &= \frac{d}{dx} x^2 f^*(x) \\ &= \frac{d}{dx} x^{\alpha+1} e^{-x} \\ &= e^{-x} x^{\alpha}\left((\alpha+1) - x\right) \\ \Rightarrow \quad x &= \alpha + 1, \quad \text{where} \quad \alpha > 1. \end{aligned}$$

Using the same reasoning as for a, we have that $x = \alpha + 1$ is a global maximum point for $x^2 f^*(x)$. Then

$$b_+ = \sqrt{(\alpha+1)^2 f^*(\alpha+1)} = \sqrt{(\alpha+1)^{\alpha+1}e^{-(\alpha+1)}} = \left(\frac{\alpha+1}{e}\right)^{(\alpha+1)/2}. \tag{6}$$

Finally, we have that

$$b_- = -\sqrt{\sup_{x\leq 0}(x^2 \cdot 0)} = 0. \tag{7}$$

(b)

To avoid producing `NaNs`, we will implement the ratio-of-uniform method on a log scale. We get the following log-transformations.

$$
\begin{aligned}
X_1 \sim \text{Uniform}(0, a) &\Rightarrow \log X_1 = \log a + \log U_1, \quad U_1 \sim \text{Uniform}(0, 1); \\
X_2 \sim \text{Uniform}(b_- = 0, b_+ = b) &\Rightarrow \log X_2 = \log b + \log U_2, \quad U_2 \sim \text{Uniform}(0, 1); \\
y = \frac{x_2}{x_1} &\Rightarrow y = \exp\{(\log x_2) - (\log x_1)\}; \\
0 \le x_1 \le \sqrt{f^*(y)} &\Rightarrow \log x_1 \le \frac{1}{2} \log f^*(y); \\
f^*(y) = \begin{cases} y^{\alpha-1} e^{-y}, & y > 0, \\ 0, & \text{otherwise}, \end{cases} &\Rightarrow \log f^*(y) = \begin{cases} (\alpha - 1)\log y - y, & y > 0, \\ -\infty, & \text{otherwise}. \end{cases}
\end{aligned}
$$

```r
set.seed(434)

lgamma_core <- function(x, alpha = 2) {
  ifelse(
    test = x <= 0,
    yes  = -Inf,
    no   = (alpha - 1)*log(x) - x
  )
}

sample_from_gamma_rou <- function(n, shape = 2, include_trials = FALSE) {
  log_a <- ((shape - 1) / 2) * (log(shape - 1) - 1)
  log_b <- ((shape + 1) / 2) * (log(shape + 1) - 1)
  trials <- 0
  y <- vector(mode = "numeric", length = n)
  for (i in 1:n) {
    repeat {
      log_x1 <- log_a + log(runif(1))
      log_x2 <- log_b + log(runif(1))
      y[i] <- exp(log_x2 - log_x1)
      log_f <- lgamma_core(y[i], alpha = shape)
      if (log_x1 <= 0.5 * log_f) {
        break
      } else {
        trials <- trials + 1
      }
    }
  }
  if (include_trials) {
    return(list(x = y, trials = trials))
  }
  return(y)
}

# generate 1000 samples for each alpha and record number of trials
n <- 1000
m <- 50
alpha <- seq(2, 2000, length.out = m)
trials <- vector(mode = "integer", length = m)
```

```r
for (i in 1:m) {
  trials[i] <- sample_from_gamma_rou(n, alpha[i], include_trials = TRUE)$trials
}

# plot trials wrt. alpha
ggplot(mapping = aes(x = alpha, y = trials)) +
  geom_point()
```

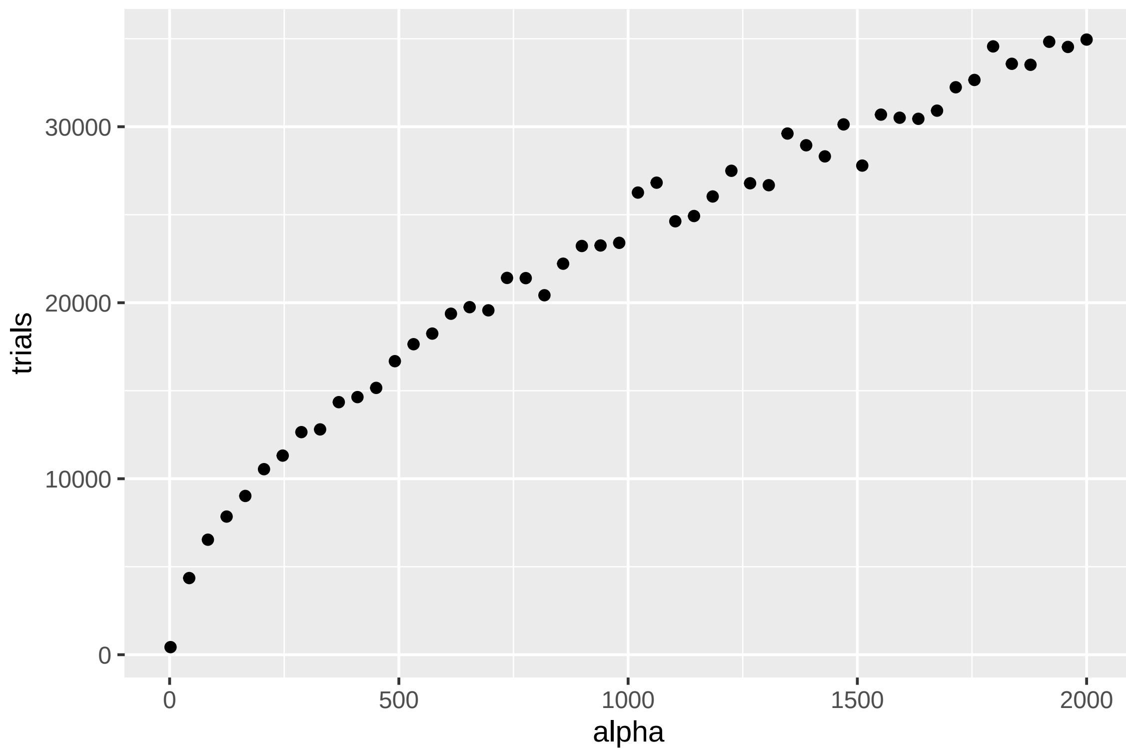

Figure 2: Number of trials before accepting $N = 1000$ simulations for various shape parameters (α) using the ratio-of-uniform method.

Figure 2 strongly suggests that the acceptance probability decreases with increasing α. That is, the ratio of the area of the square $a \cdot (b_+ - b_-) = ab$ and the region C_f is increasing when α increases.

3.

(a)

Let X_1 and X_2 be independent random variables where $X_1 \sim \text{Gamma}(\alpha_1, \beta = 1)$ and $X_2 \sim \text{Gamma}(\alpha_2, \beta = 1)$. Then $X_1 + X_2$ has the following mgf:

$$\begin{aligned} M_{X_1+X_2}(t) &= M_{X_1}(t) \cdot M_{X_2}(t) \\ &= (1-t)^{-\alpha_1} \cdot (1-t)^{-\alpha_2} \\ &= (1-t)^{-(\alpha_1+\alpha_2)}, \quad t < 1. \end{aligned}$$

That is, $X_1 + X_2 \sim \text{Gamma}(\alpha_1 + \alpha_2, \beta = 1)$.

(b)

A Exp(1) distribution is a special case of Gamma(α, β) with parameters $\alpha = \beta = 1$. Using the result obtained in **(a)**, we then have that the sum of a random sample $X_1, \ldots, X_k$ drawn from Exp(1) has a Gamma$(k, 1)$ distribution. We can use this fact to improve our algorithm. Let $k \in \mathbb{N}_0$ and let $r \in [0, 1)$ and assume $k + r > 0$. Then we can decompose any $\alpha > 0$ as

$$\alpha = k + r.$$

Let $X_1, \ldots, X_k \sim \text{Exp}(1)$ be a random sample and $W \sim \text{Gamma}(r, 1)$, where W and X_i, $i = 1, \ldots, k$ are mutual independent. Then

$$Y = W + \sum_{i=1}^{k} X_i \sim \text{Gamma}(\alpha = k + r, \beta = 1). \tag{8}$$

That is, for any $\alpha > 0$, we will only use the rejection sampling method for $r = \alpha \bmod 1$ to sample $W \sim \text{Gamma}(r, 1)$, and for the remaining $k = \alpha - r$ (possibly zero), we sample $X_1, \ldots, X_k \sim \text{Exp}(1)$, and use (8) to sample from Gamma$(\alpha, 1)$.

```
sample_from_gamma_improved <- function(n, shape = 1) {
  r <- shape %% 1
  if (r > 0) {
    w <- sample_from_gamma_rej(n, shape = r)
  } else {
    w <- 0
  }
  k <- shape - r
  if (k >= 1) {
    xk <- matrix(generate_from_exp(n * k, rate = 1),
                 nrow = n,
                 ncol = k
    )
    x <- rowSums(xk)
  } else {
    x <- 0
  }
  return(x + w)
}
```

4.

Since β is an inverse scale parameter, we can simply draw samples from Gamma$(\alpha, 1)$ and divide every sample by β. This can be shown by looking at the mgf of X/β where $X \sim \text{Gamma}(\alpha, 1)$.

$$M_{X/\beta}(t) = M_X\left(\frac{t}{\beta}\right) = \left(1 - \frac{t}{\beta}\right)^{-\alpha} \sim \text{Gamma}(\alpha, \beta).$$

```
sample_from_gamma_final <- function(n, shape = 1, rate = 1) {
  (1 / rate) * sample_from_gamma_improved(n, shape = shape)
}
```

5.

(a)

Let X and Y be independent random variables where $X \sim \text{Gamma}(\alpha, 1)$ and $Y \sim \text{Gamma}(\beta, 1)$. Let

$$z = g_1(x, y) = \frac{x}{x+y} \quad \text{and} \quad w = g_2(x, y) = x + y$$

Then $Z = g_1(X, Y) \in (0, 1)$ and $W = g_2(X, Y) > 0$. This gives us

$$\begin{aligned}
&x = g_1^{-1}(z, w) = zw \quad \text{and} \quad y = g_2^{-1}(z, w) = w(1-z), \\
&|\det(J)| = \left|\det \begin{pmatrix} \partial_z g_1^{-1}(z, w) & \partial_w g_1^{-1}(z, w) \\ \partial_z g_2^{-1}(z, w) & \partial_w g_2^{-1}(z, w) \end{pmatrix}\right| = \left|\det \begin{pmatrix} w & z \\ -w & 1-z \end{pmatrix}\right| = |w| = w.
\end{aligned}$$

The marginal distribution $f_Z(z)$ is then found as follows.

$$\begin{aligned}
f_Z(z) &= \int_0^\infty f_{Z,W}(z, w)\, dw \\
&= \int_0^\infty f_{X,Y}\left(g_1^{-1}(z, w), g_2^{-1}(z, w)\right) |\det(J)|\, dw \\
&= \int_0^\infty f_X(zw) \cdot f_Y(w(1-z)) \cdot w\, dw \\
&= \int_0^\infty \frac{1}{\Gamma(\alpha)} (zw)^{\alpha-1} e^{-zw} \cdot \frac{1}{\Gamma(\beta)} (w(1-z))^{\beta-1} e^{-w(1-z)} \cdot w\, dw \\
&= \frac{1}{\Gamma(\alpha)\Gamma(\beta)} z^{\alpha-1}(1-z)^{\beta-1} \Gamma(\alpha+\beta) \int_0^\infty \frac{1}{\Gamma(\alpha+\beta)} w^{(\alpha+\beta)-1} e^{-w}\, dw \\
&= \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} z^{\alpha-1}(1-z)^{\beta-1}, \quad z \in (0, 1).
\end{aligned}$$

That is, $f_Z(z) \sim \text{Beta}(\alpha, \beta)$.

(b)

```
sample_from_beta <- function(n, alpha, beta) {
  x <- sample_from_gamma_final(n, shape = alpha)
  y <- sample_from_gamma_final(n, shape = beta)
  return(x / (x + y))
}
```

Problem C: Monte Carlo integration and variance reduction

1.

2.

3.

(a)

(b)

Problem D: Rejection sampling and importance sampling

1.

2.

3.

4.