The exponential distribution

The exponential distribution models the distance between events in a Poisson point process. Parameterised by the rate, θ, the distribution function is

$$F(x) = 1 - e^{-\theta x},$$

while the probability density function is

$$f(x|\theta) = \theta e^{-\theta x}.$$

This distribution has mean $\frac{1}{\theta}$ and variance $\frac{1}{\theta^2}$.

Programs

The exponential pdf is implemented as `exponential_pdf`, the distribution function as `exponential_distribution`, and the inverse of the distribution function as `exponential_distribution_inv`.

We can sample from the exponential distribution with the following code:

```
theta <- 1.2
n <- 200
distribution <- Curry(exponential_distribution_inv, rate = theta)

exp_samples <- distribution_sampler(distribution, n)
head(exp_samples)
```

```
## [1] 0.7533789 0.2234259 0.0336711 0.4024187 4.5403538 0.1778181
```

R also has a built-in exponential distribution with pdf `dexp`, which can be sampled with `rexp`. We can test that this matches our implementation using a one-sample Kolmogorov-Smirnov test:

```
ks_results <- ks.test(exp_samples, "dexp")
ks_results[["p.value"]]
```

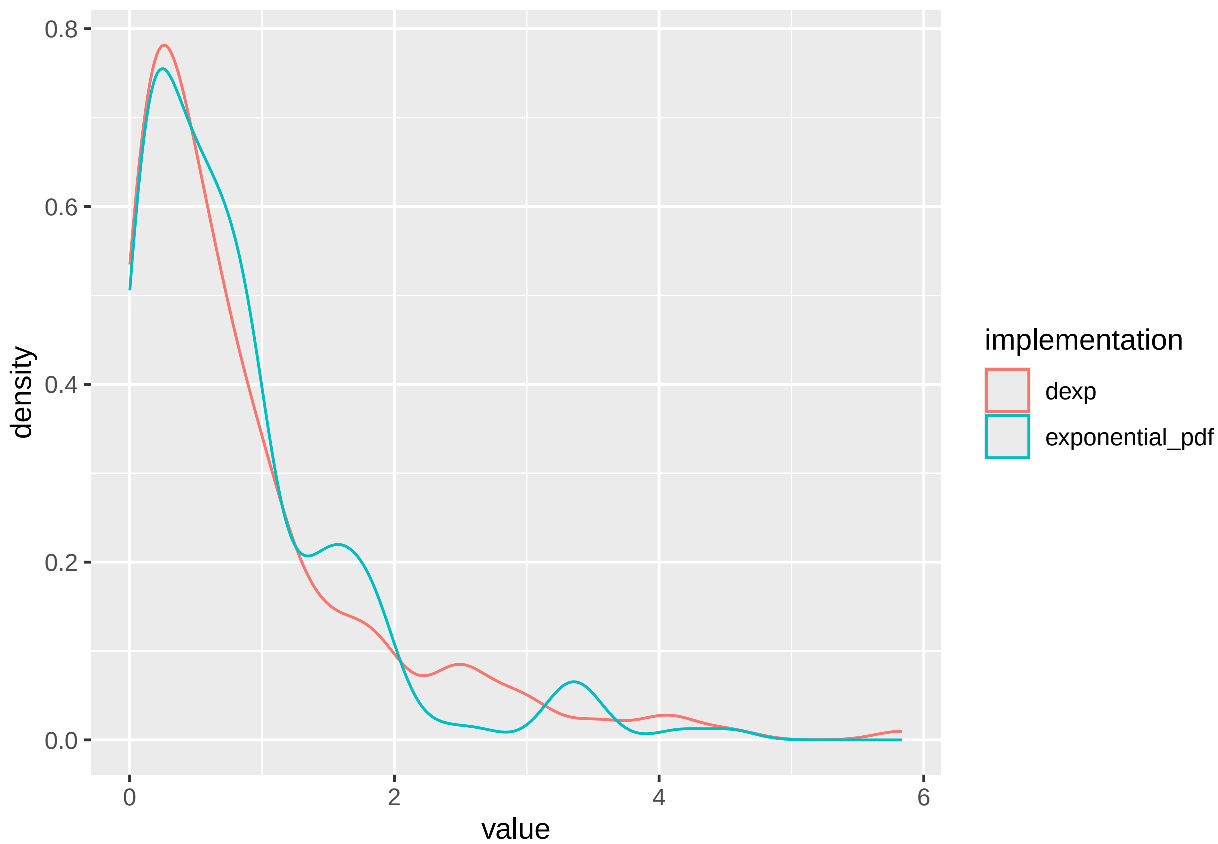
```
## [1] 1.864105e-170
```

In this instance, since $p < 0.05$, we conclude that the distributions match.

We can also plot the distributions against each other:

```
dat <- data.frame(implementation =
                    factor(rep(c("exponential_pdf", "dexp"), each = n)),
                  value = c(exp_samples, rexp(n, rate = theta)))

ggplot(dat, aes(x = value, colour = implementation)) + geom_density()
```



Here, we have some noise from our relatively low number of samples, but the distributions clearly have the same shape.

Problems

Problem 1

Suppose that instead of indexing the probability distribution function by its rate θ, we decide to index it by its median m given by

$$\int_0^m f(x|\theta)\ \mathrm{d}x = \frac{1}{2}.$$

Find θ as a function of m and hence find $g(x|m) = f(x|\theta(m))$.

Solution We know that $\int f(x|\theta)\ \mathrm{d}x = F(x)$, so we need to solve $F(m) = \frac{1}{2}$. We find that $m = \frac{\ln(2)}{\theta}$, or $\theta(m) = \frac{\ln(2)}{m}$, so that

$$g(x|m) = f(x|\theta(m)) \ = \frac{\ln(2)}{m} e^{-\frac{\ln(2)}{m}x} \ = \frac{\ln(2)}{m} 2^{-\frac{x}{m}}.$$

Problem 2

Take $(u_1, ..., u_n)$, sampled from Unif$[0, 1]$, and hence compute the x_i defined by $u_i = 1 - e^{-\theta x_i}$, giving $(x_1, ..., x_n)$ sampled from $f(x|\theta)$. Try this for $n = 6$, $\theta = 1.2$. Plot the resulting log likelihood function $\ell(m)$

against m where

$$\ell(m) = \ln \prod_{i=1}^{n} g(x_i|m).$$

Derive analytically $\hat{m}$, the value of m which maximises $\ell(m)$, and compare this with m_0, the true value of the median.

Solution We sample using the following code (the sampled u_i are computed in the `distribution_sampler`).

```
theta <- 1.2
n <- 6
distribution <- Curry(exponential_distribution_inv, rate = theta)

exp_samples <- distribution_sampler(distribution, n)
```

To compute the log likelihood, we can observe that

$$\ell(m) = \ln \prod_{i=1}^{n} g(x_i|m) \;\; = \ln \prod_{i=1}^{n} \frac{\ln(2)}{m} 2^{-\frac{x_i}{m}} \;\; = \ln(\frac{\ln(2)^n}{m^n}) + \ln(2^{-\frac{\sum x_i}{m}}) \;\; = n \ln(\frac{\ln(2)}{m}) - \frac{\sum x_i}{m} \ln(2) \;\; = n \times (\ln \ln(2) - \ln(m) - \frac{\bar{x}}{m} \ln($$

Where $\bar{x}$ is the mean of the x_i.

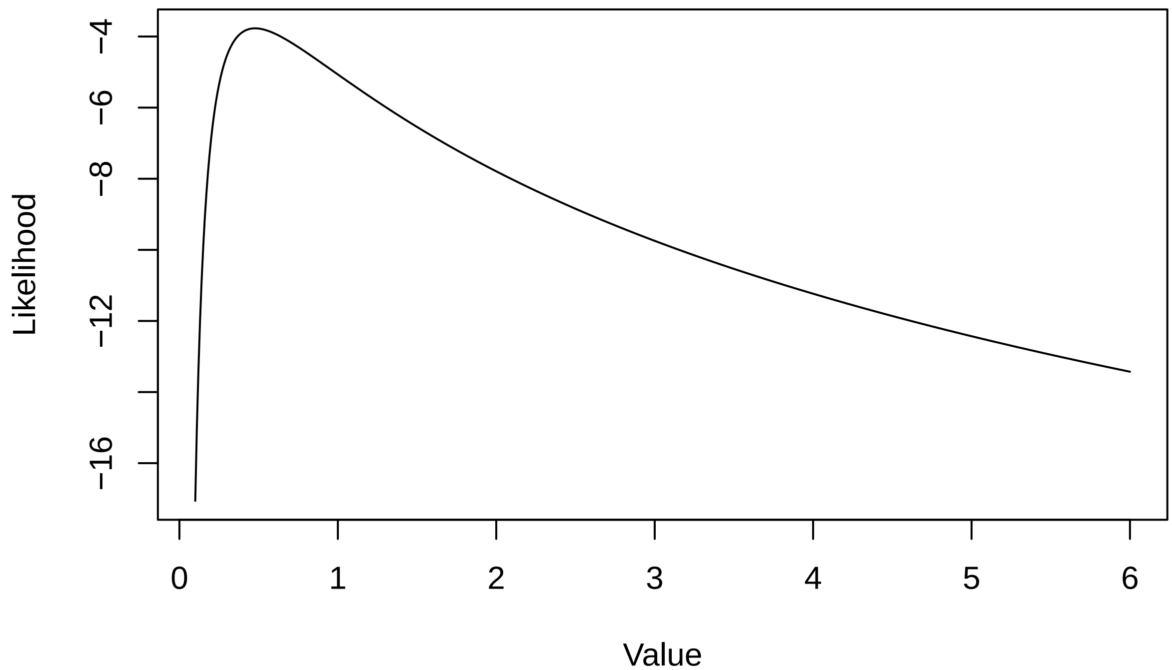
The following code plots ℓ:

```
log_likelihood <- function(n, samples, m) {
  n * log(log(2) / m) + (-sum(samples) / m) * log(2)
}

ell <- Curry(log_likelihood, n = n, samples = exp_samples)

x <- seq(0.1, 6, 0.01)
y <- sapply(x, ell)

plot(x, y, type = "l", xlab = "Value", ylab = "Likelihood")
```



Now, we can compute $\hat{m}$ analytically by considering $\frac{\mathrm{d}}{\mathrm{dm}}\ell(m) = n \times (\frac{\bar{x}}{m^2}\ln(2) - \frac{1}{m})$.

$\hat{m}$ will be the value for which $\ell'(\hat{m}) = 0$, i.e. for which $\hat{m} = \bar{x} \times \ln(2)$. Since $\bar{x}$ is an efficient and unbiased estimator of the sample mean, $\frac{1}{\theta}$, we expect $\hat{m} \approx \frac{\ln(2)}{\theta} = m_0$. We can compute $\hat{m} = 0.478...$, while $m_0 = 0.578...$, which are approximately equal.

Problem 3

Repeat all of Problem 2 for $n = 25, 50, 100$, and comment on the qualitative changes you observe (if any) in the shape of $\ell(m)$.

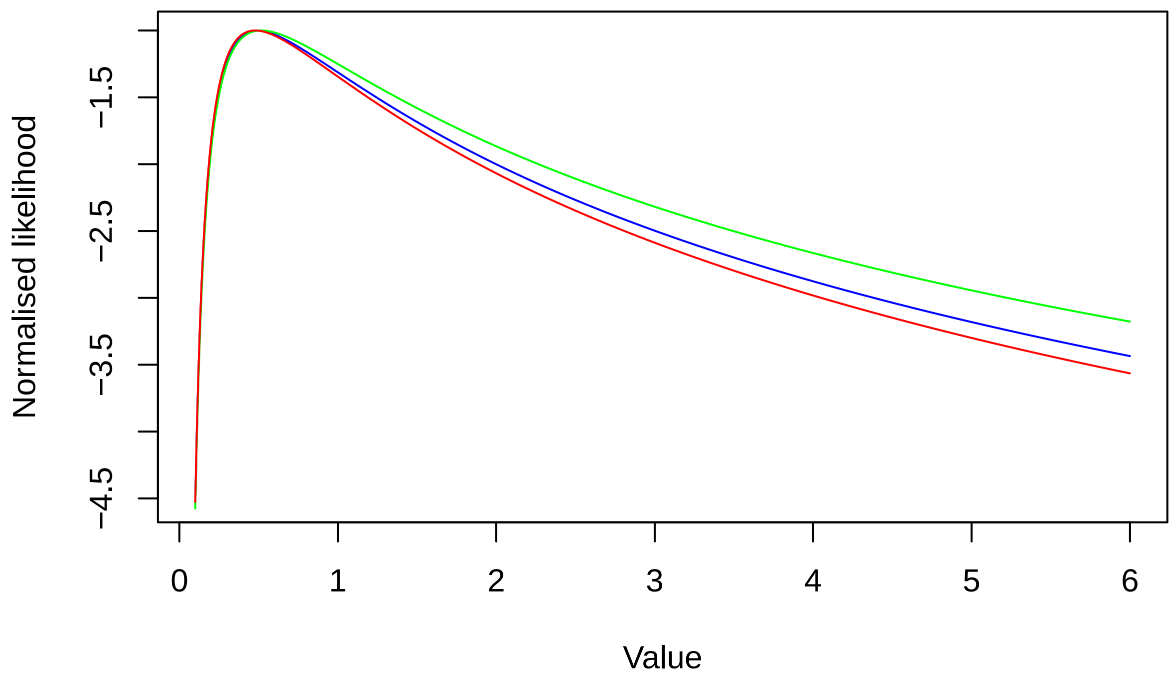
Solution We plot ℓ with the following code:

```
exp_samples_1 <- distribution_sampler(distribution, 25)
exp_samples_2 <- distribution_sampler(distribution, 50)
exp_samples_3 <- distribution_sampler(distribution, 100)

el_1 <- Curry(log_likelihood, n = 25, samples = exp_samples_1)
el_2 <- Curry(log_likelihood, n = 50, samples = exp_samples_2)
el_3 <- Curry(log_likelihood, n = 100, samples = exp_samples_3)

m_1 <- sum(exp_samples_1) / 25 * log(2)
m_2 <- sum(exp_samples_2) / 50 * log(2)
m_3 <- sum(exp_samples_3) / 100 * log(2)
```

```r
x <- seq(0.1, 6, 0.01)
plot(x, sapply(x, el_1) / -el_1(m_1), type = "l", col = "blue",
     xlab = "Value", ylab = "Normalised likelihood")
lines(x, sapply(x, el_2) / -el_2(m_2), col = "green")
lines(x, sapply(x, el_3) / -el_3(m_3), col = "red")
```



We observe that all three plots are similar, but that smaller sample sizes result in poorer estimations of the median, and different gradients for ℓ.

Problem 4

Suppose that X, Y are independent random variables, each with a probability distribution function corresponding to an exponential with mean $\frac{1}{\theta}$. Calculate the moment generating function $M_X(\lambda) = E(e^{\lambda X})$ of X. Show that $X + Y \sim \Gamma(2, \theta)$.

Solution We can compute

$$M_X(\lambda) = \int_0^\infty e^{\lambda x} f(x|\theta)\mathrm{dx} \;= \int_0^\infty \theta e^{(\lambda-\theta)x}\mathrm{dx} \;= \frac{\theta}{\theta - \lambda}.$$

Then we know that, since X and Y are independent, $M_{X+Y}(\lambda) = M_X(\lambda) \times M_Y(\lambda) = \frac{\theta^2}{(\theta-\lambda)^2}$. But this is the moment generating function for a $\Gamma(2, \theta)$-distributed variable. Since the moment generating function determines the distribution, we are done.