# Lab 8 – Exponential distributions, more distributions of sample means

Name: Jihyo Kim

Student number: A01017545 Set: 3M

In this lab, we are going to be investigating the exponential distribution. We will also continue investigating the distributions of meansof samples of populations of various different distributions.

**To submit: answers to all numbered questions. When the question asks you to write code or create graphs, submit the code and/or graphs in the Word document as part of your answer. Also submit a single .R file that contains all of your code.**

## Continuous Distributions, continued

Last week, we imagined a person waiting for a bus that comes very reliably every 20 minutes. However, anyone who has ever waited for buses knows that no bus comes exactly every 20 minutes (or exactly every 5 minutes, or exactly every hour). More realistically, a bus may come *on average* every 20 minutes. The exponential distribution with mean 20 gives a fairly accurate model of waiting times **between** buses.

# Experiment: Waiting for a more realistic bus (exponential distribution)

We use the **qexp()** function to model exponential waiting times. For an exponential distribution with rate **r,**  the command

qexp(p, rate = r, lower.tail = TRUE, log.p = FALSE)

returns the value such that there is a probability p that the time between occurrences will be **less** than that value.

> qexp(.1, rate = 1, lower.tail = TRUE, log.p = FALSE)

[1] 0.10536

> qexp(.5, rate = 1, lower.tail = TRUE, log.p = FALSE)

[1] 0.69315

> qexp(.9, rate = 1, lower.tail = TRUE, log.p = FALSE)

[1] 2.3026

For instance, at a mean rate of 1 occurrence per minute, there is a 50% chance the waiting time between occurrences will be less than 0.69315 minutes.

1. For the bus example, what is **r**? (Hint: the units of **r** are “buses per minute”.)

1 bus per 20 minutes => 1/20 bus per minute; therefore, 0.05.

1. Find qexp(.5, rate = r, lower.tail = TRUE, log.p = FALSE) for that value of **r**. This should be the median of an exponentially-distributed random variable with mean 20. How does it compare to the mean? Based on the distribution of the exponential random variable, does this makes sense? (Hint: the word “skew” should appear in your answer.)

The median is calculated to 13.86, which is smaller than the mean. Therefore, the distribution is positively skewed (or right skewed), which agrees to exponential probability distribution.

In order to model **n** waiting times, we need to get waiting times corresponding to **n** probabilities. Those **n** probabilities will be distributed uniformly. That way, approximately 10% of waiting times will be in the bottom 10% of the exponential distribution.

The command

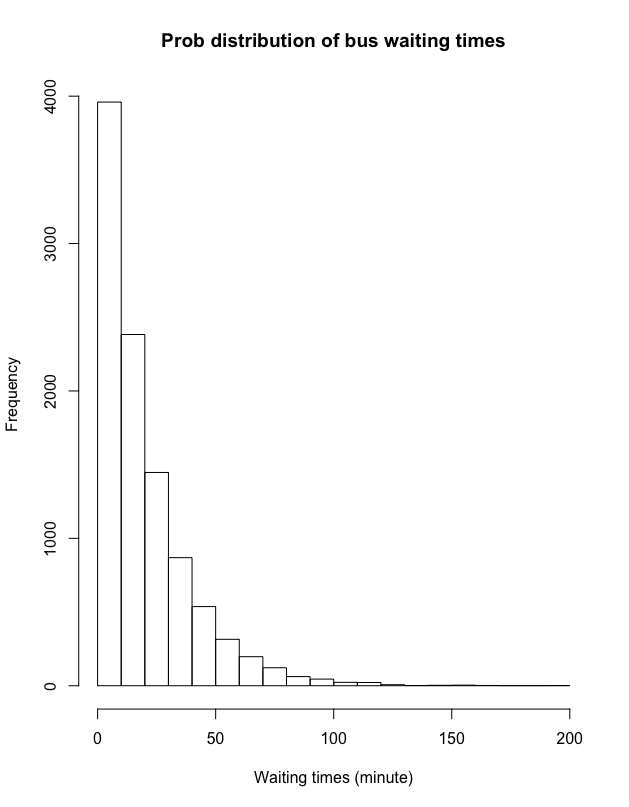
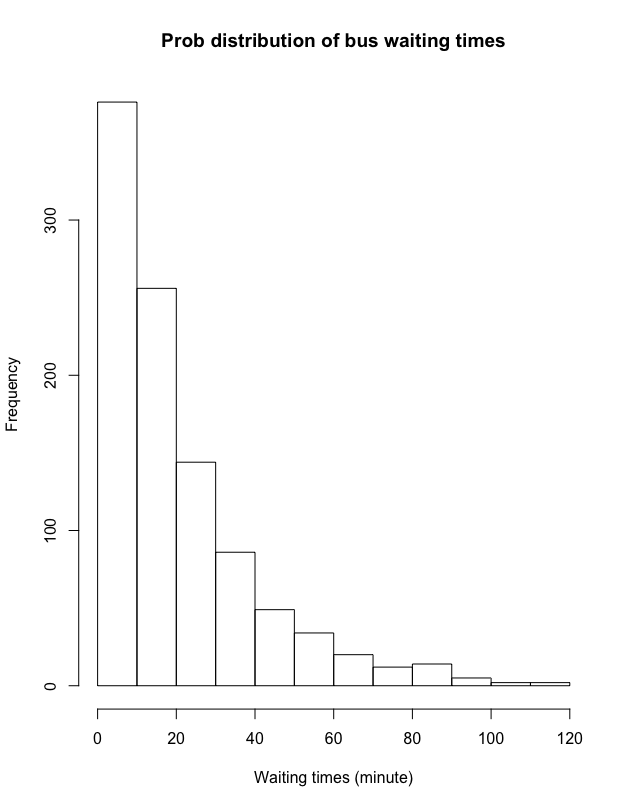
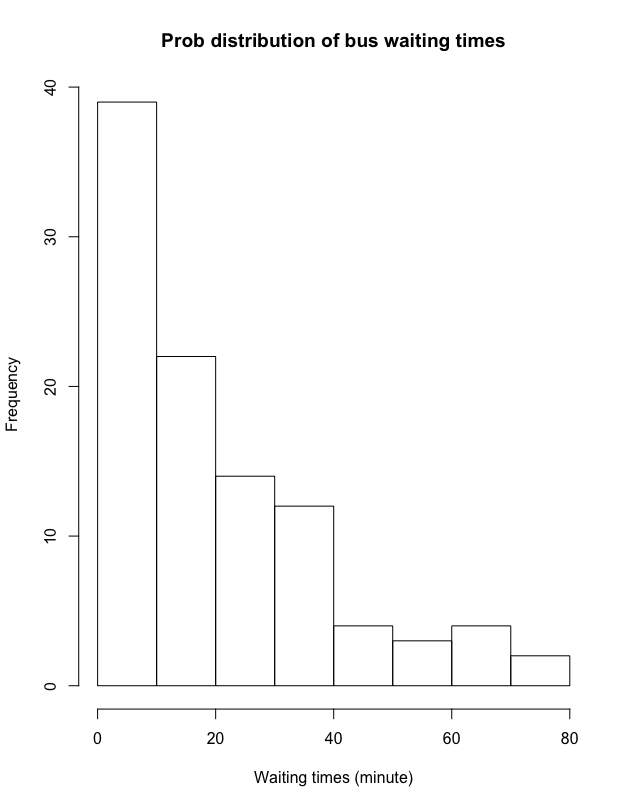
> qexp(runif(n), rate = r, lower.tail = TRUE, log.p = FALSE))

gives **n** waiting times distributed exponentially with a mean rate of **r** per unit time.

Equivalently, we can use the **rexp** command to generate random values that follow an exponential distribution. For instance, the command

> rexp(n, r)

gives the same result as the earlier **qexp()** command.

1. Generate appropriately-labelled histograms that give the frequency of waits between buses that have exponentially-distributed waiting times with mean 20 minutes for n=100, 1000, and 10000. Use the same number of classes for each. Do the distributions look exponential?

## Distribution of Sample Means, Continued

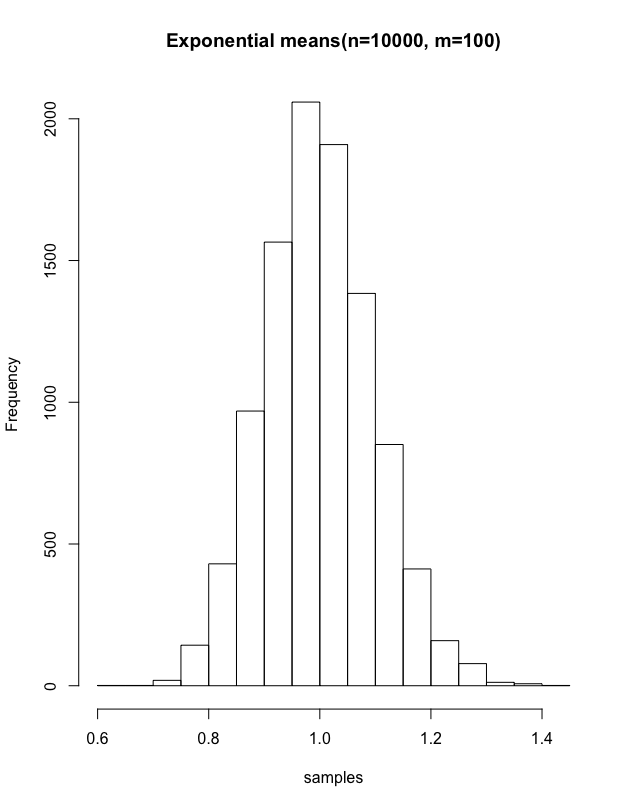
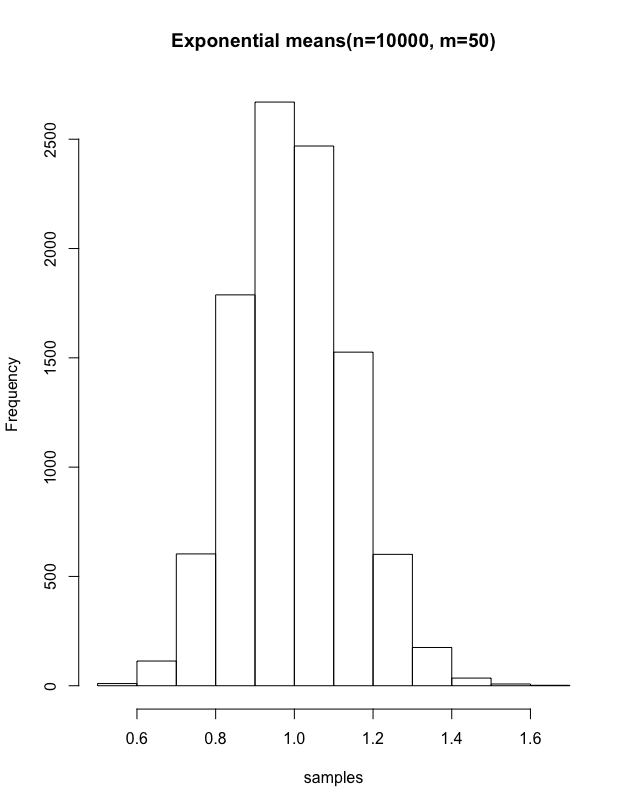
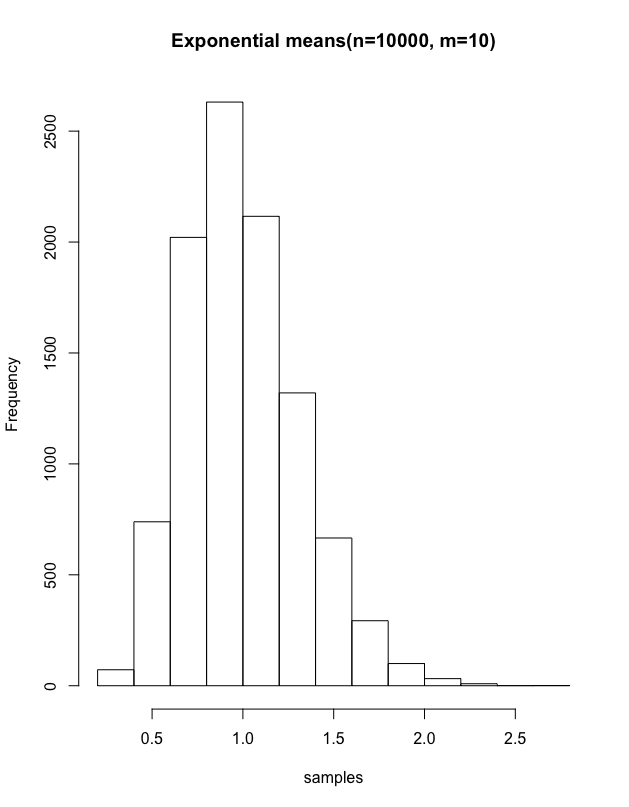
Last week, we simulated sampling from a uniformly-distributed population – die rolls – and investigated the distribution of the sample means. Now we will investigate the distributions of sample means of populations that have normal and exponential kinds of distributions.

## Distribution of sample means in a normally distributed population

1. We are now going to investigate the answer to the question: What is the probability that the mean of a sample of size **m** taken from a standard normal distribution is between **x** and **–x**? We will do this using a simulation. Copy and paste your **NormalMeans(n,m)** function from last week in the your Lab7.m script file and rename it **NormalMeansProb(n,m,x)**. Adjust your new function so that it returns the proportion of sample means that are between -**x** and **+x**. This proportion approximates the probability that a sample of size **m** taken from a normal distribution is between **-x** and **x** standard deviations of the **target** population mean.  
     
   Run your function for **n=10000** and values of **m** and **z** as given in the table below. Do your results agree with the theory from class?

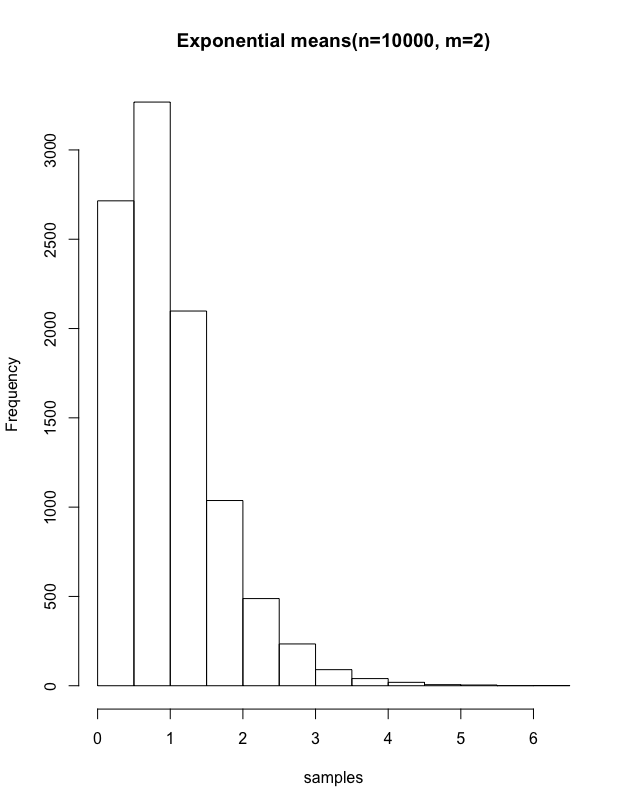
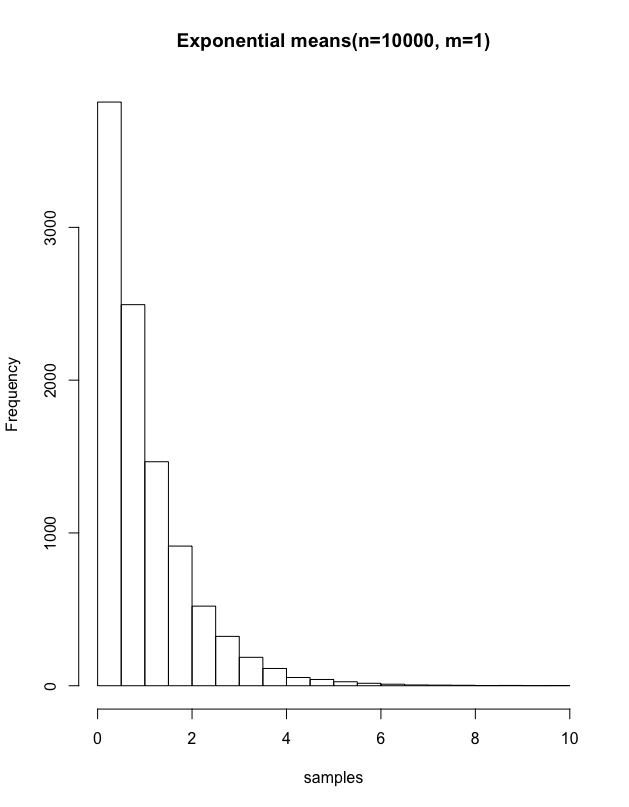
|  |  |  |  |
| --- | --- | --- | --- |
| **m\x** | 1 | 0.5 | 0.2 |
| 2 |  |  |  |
| 5 |  |  |  |
| 10 |  |  |  |

## Distribution of sample means in an exponentially-distributed population

1. Create a function called **ExponentialMeans(n,m)** that simulates taking **n** samples of size **m** from exponentially-distributed populations with rate 1. As before, this new function will compute (and keep track of) the sample mean for each of those **n** samples.Your function should return the following:

* The mean of the **n** sample means
* The standard deviation of the **n** sample means
* An appropriately-named **histogram** of the **n** sample means (use the default bins)

Run your function for **n=10000** and for the following values of **m**: 1, 2, 10, 50, 100.

Submit five outputs (five histograms, each with the means and standard deviations listed below). How do the means and standard deviations compare as **m** increases? How do the shapes of the graphs compare to one another, and to the graphs you obtained earlier?

m=1) mean: 1.016, standard deviation: 1.014

m=2) mean: 0.9889254, standard deviation: 0.7007895

m=10) mean: 0.9998865, standard deviation: 0.3146353

m=50) mean: 0.9990515, standard deviation: 0.1420149

m=100) mean: 0.9990655, standard deviation: 0.09825373

As ‘m’ increases, the graph becomes more like the normal distribution graph.

1. We are now going to investigate the answer to the question: What is the probability that the mean of a sample of size **m** taken from an exponential distribution with mean 1 is between **1-x** and **1+x**? Again we will do this using a simulation. Copy and paste your **ExponentialMeans(n,m)** function in the same script file and rename it **ExponentialMeansProb(n,m,x)**. Adjust your new function so that it returns the proportion of sample means that are between **1-x** and **1+x**. This proportion approximates the probability that the mean of a sample of size **m** taken from an exponential distribution is between **-x** and **x** standard deviations of the **target** population mean.  
     
   Run your function for **n=10000** and values of **m** and **x** as given in the table below. Do your results agree with the theory from class? How do your results compare to the ones in Question 6?

|  |  |  |  |
| --- | --- | --- | --- |
| **m\x** | 1 | 0.5 | 0.2 |
| 2 |  |  |  |
| 5 |  |  |  |
| 10 |  |  |  |

it becomes more like normal distribution