mylab1

chenpeng

2024-07-01

目录

Lab1

Part 1

1&2

```
rates <- c(0.1, 0.5, 1, 5, 10)
exp.draws <- vector("list", length(rates))
means <- vector("numeric", length(rates))
sds <- vector("numeric", length(rates))

for (i in seq_along(rates)) {
  exp.draws[[i]] <- rexp(200, rate = rates[i])
  means[i] <- mean(exp.draws[[i]])
  sds[i] <- sd(exp.draws[[i]])

  cat("exp.draws.",rates[i],": mean = ",means[i],", ",sep="")
  cat("standard deviation = ",sds[i],"\n",sep="")

}
```

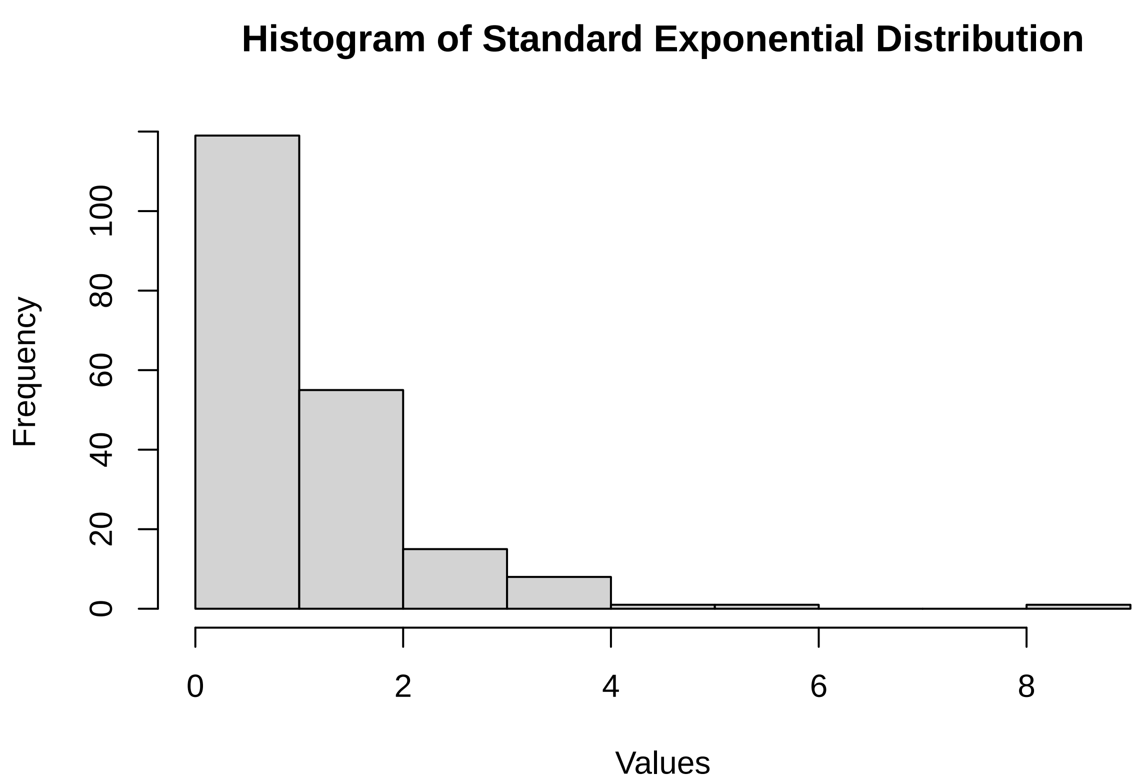
```
## exp.draws.0.1: mean = 10.42904, standard deviation = 9.967474
## exp.draws.0.5: mean = 1.796304, standard deviation = 1.707906
## exp.draws.1: mean = 1.039251, standard deviation = 1.040004
## exp.draws.5: mean = 0.1924539, standard deviation = 0.1735181
## exp.draws.10: mean = 0.1061536, standard deviation = 0.1113563
```

```r
exp.draws.0.1 <- exp.draws[[1]]
exp.draws.0.5 <- exp.draws[[2]]
exp.draws.1 <- exp.draws[[3]]
exp.draws.5 <- exp.draws[[4]]
exp.draws.10 <- exp.draws[[5]]
```

3

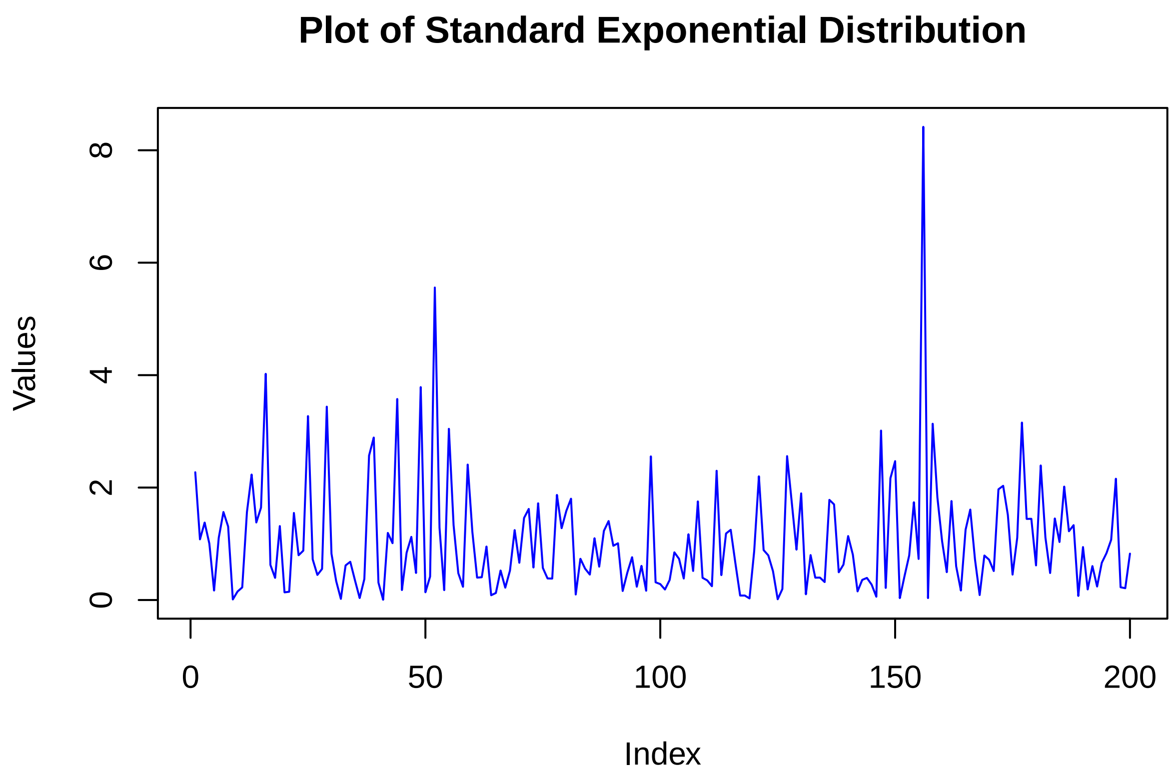
a.

```r
# Assuming exp.draws.1 is already defined as the standard exponential distribution
hist(exp.draws.1, main = "Histogram of Standard Exponential Distribution", xlab = "Values")
```



b.

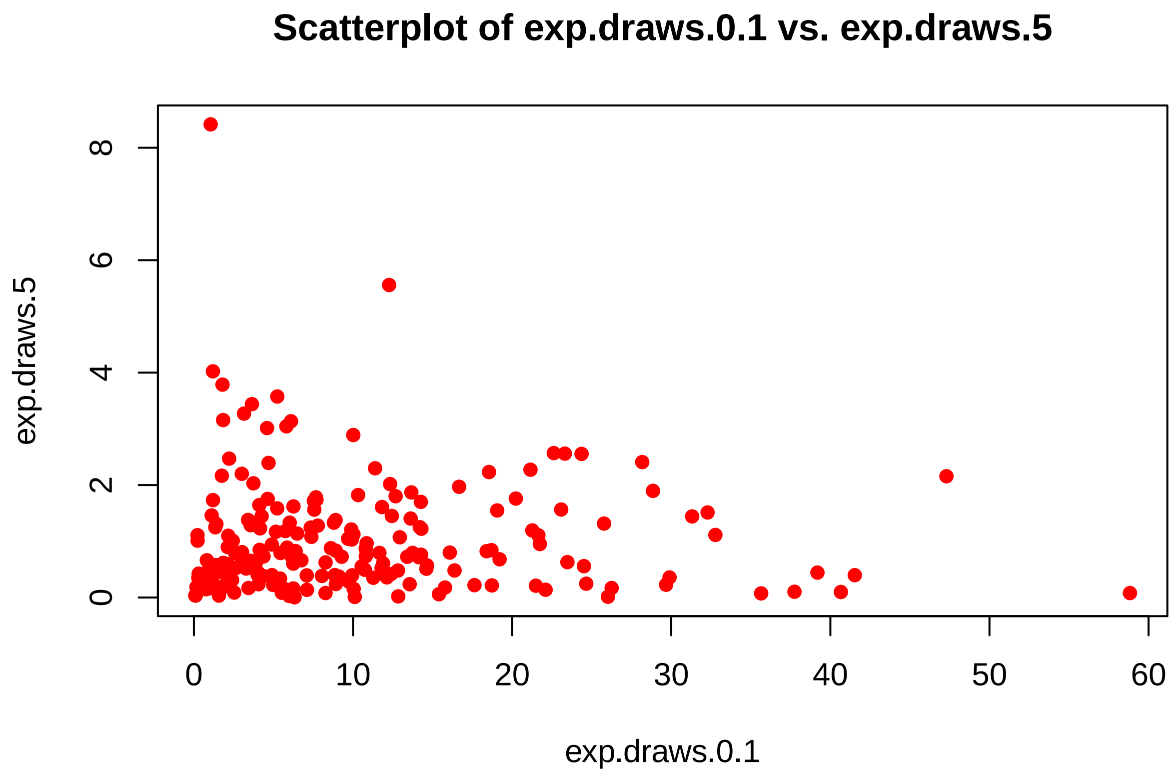
```r
# Plotting random values from standard exponential distribution
plot(exp.draws.1, type = "l", col = "blue", main = "Plot of Standard Exponential Distribution", xl
```



c.

Let's say you want to plot exp.draws.0.1 against exp.draws.5:

```r
# Plotting scatterplot of two random value vectors
plot(exp.draws[[1]], exp.draws[[3]], main = "Scatterplot of exp.draws.0.1 vs. exp.draws.5", xlab =
```



Tip: pch determines the type of the point

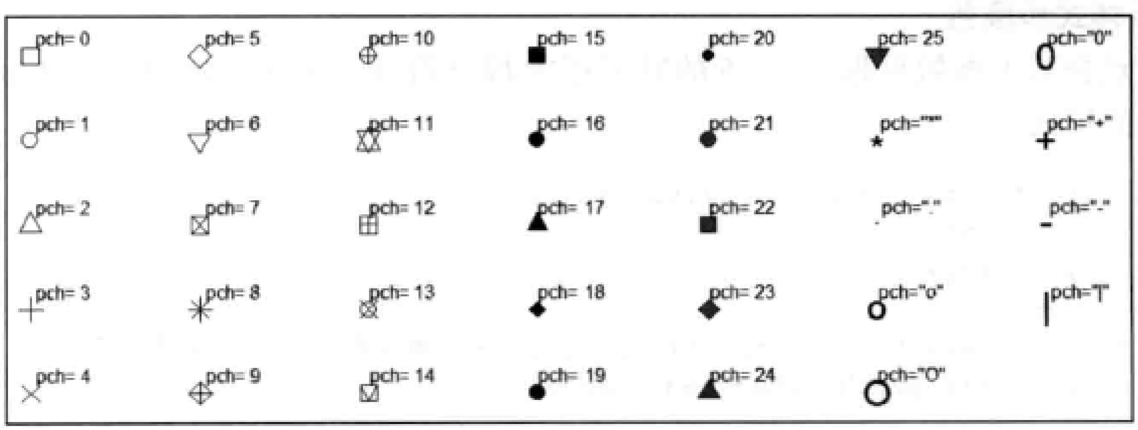
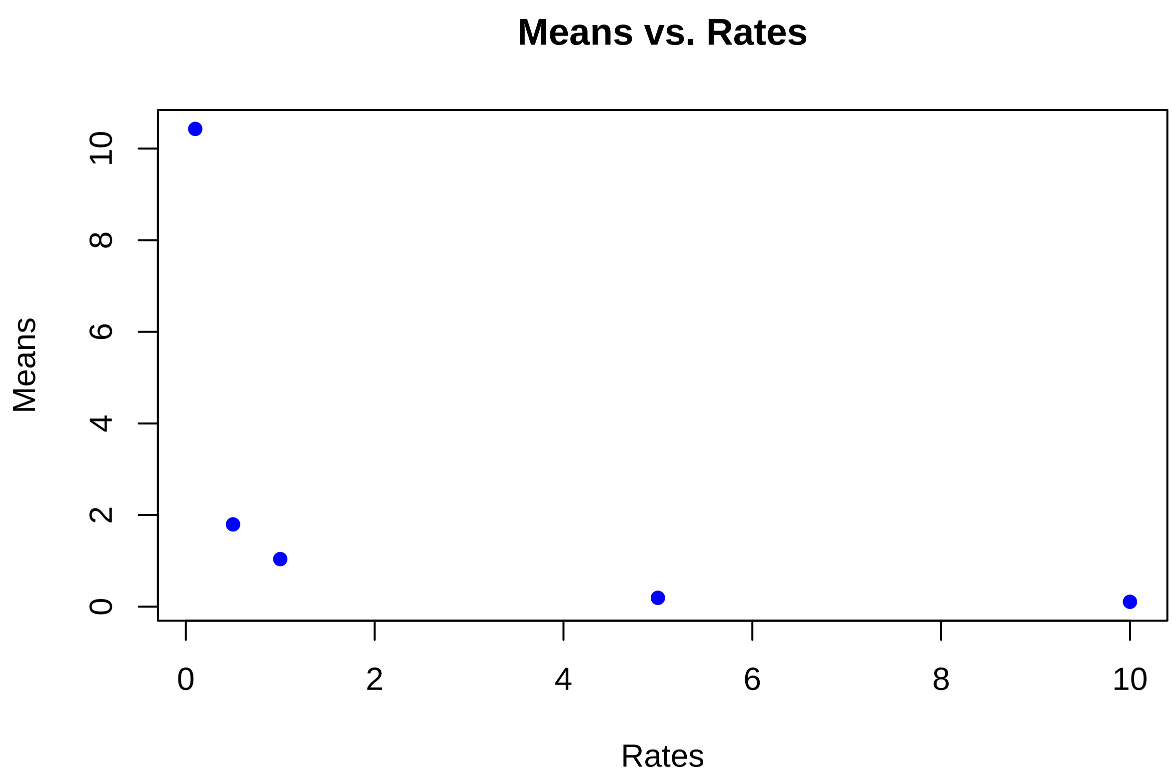


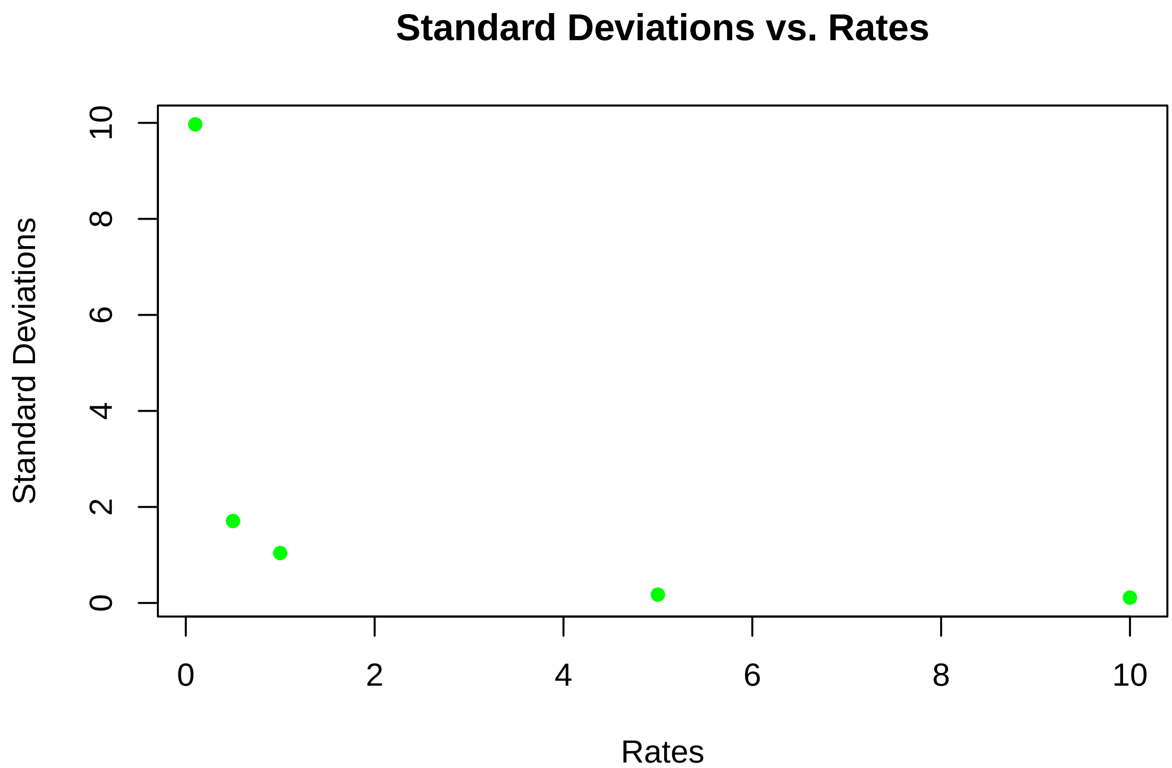
图 1: pch 对应图

d.

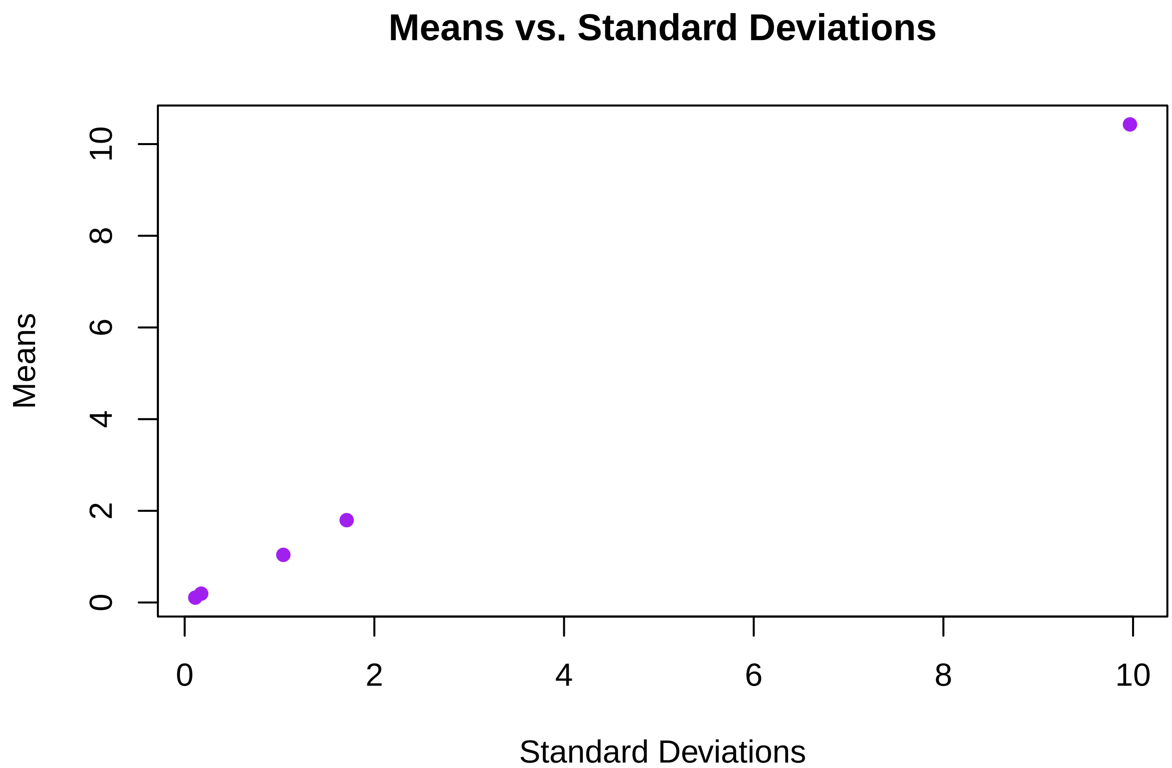
```r
# Scatterplots
# a. Means versus Rates
plot(rates, means, main = "Means vs. Rates", xlab = "Rates", ylab = "Means", col = "blue", pch = 1
```




```r
# b. Standard Deviations versus Rates
plot(rates, sds, main = "Standard Deviations vs. Rates", xlab = "Rates", ylab = "Standard Deviatio
```




```r
# c. Means versus Standard Deviations
plot(sds, means, main = "Means vs. Standard Deviations", xlab = "Standard Deviations", ylab = "Mea
```



Part 2

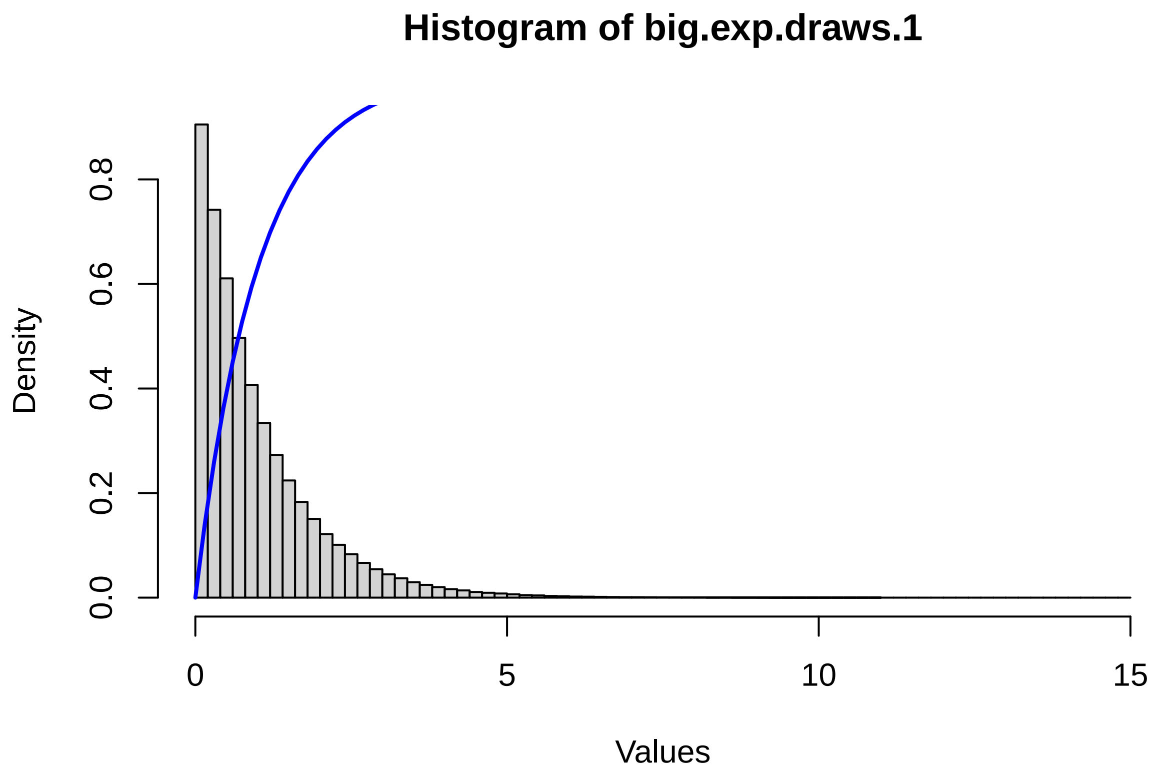
a.

```
big.exp.draws.1 <- rexp(1.1e6, rate = 1)
mean_big <- mean(big.exp.draws.1)
sd_big <- sd(big.exp.draws.1)
cat("big.exp.draws.1: mean = ", mean_big, ", standard deviation = ", sd_big,"\n",sep="")
```

```
## big.exp.draws.1: mean = 0.9985369, standard deviation = 0.9976614
```

b.

```
# Plot histogram
hist(big.exp.draws.1, breaks = 100, freq = FALSE, main = "Histogram of big.exp.draws.1", xlab = "V

# Plot function 1 - exp(-x)
curve(1 - exp(-x), add = TRUE, col = "blue", lwd = 2)
```



Explanation for 5b):

The histogram of big.exp.draws.1 resembles the function (1 - e^{-x}), which is the cumulative distribution function (CDF) of the exponential distribution. This indicates that the data matches the expected distribution.

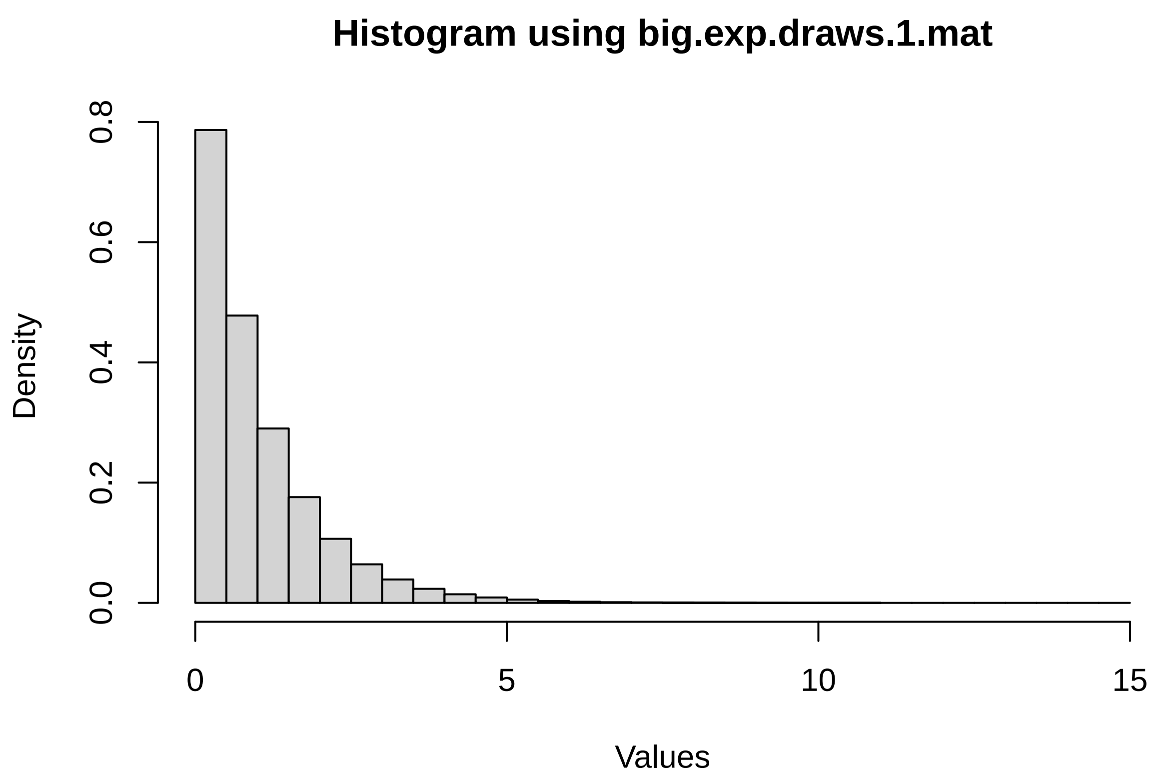
c.

```
# Create vector of entries greater than 1
greater_than_1 <- big.exp.draws.1[big.exp.draws.1 > 1]

# Calculate mean
mean_greater_than_1 <- mean(greater_than_1)
mean_greater_than_1
```

```
## [1] 1.99677
```

d.

```
# Create matrix with 1100 rows and 1000 columns
big.exp.draws.1.mat <- matrix(big.exp.draws.1, nrow = 1100, ncol = 1000)

# Histogram using hist() function
hist_result <- hist(big.exp.draws.1.mat, freq = FALSE, main = "Histogram using big.exp.draws.1.mat
```



Explanation for 5d):

The hist() function treats each column of the matrix independently and creates histograms for each column.

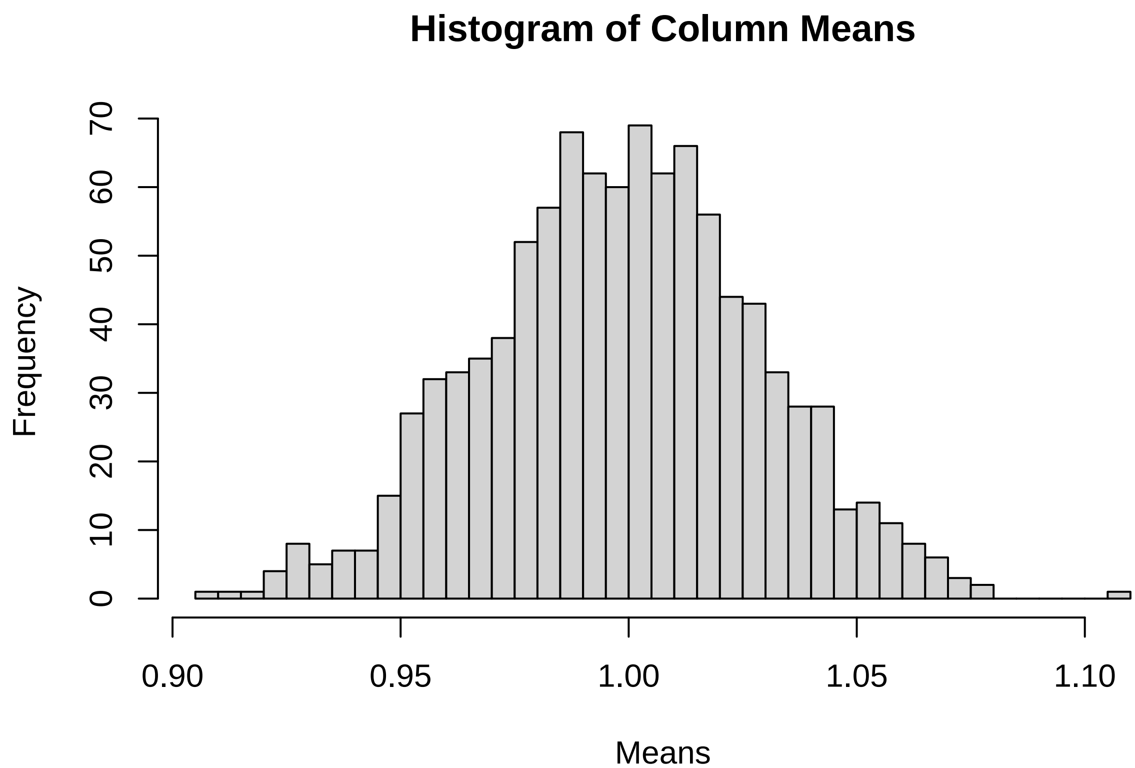
e.

```
# Calculate mean of the 371st column
mean_371st <- mean(big.exp.draws.1.mat[, 371])
mean_371st
```

```
## [1] 1.009559
```

f.

```r
# Calculate means of all columns
col_means <- colMeans(big.exp.draws.1.mat)

# Plot histogram of column means
hist(col_means, main = "Histogram of Column Means", xlab = "Means", breaks = 30)
```



Explanation for 5f):

The histogram of column means does not match the shape of the original histogram (problem 5b) because the distribution of means of many samples (columns) tends to approximate a normal distribution (by the Central Limit Theorem), whereas the original data (big.exp.draws.1) follows an exponential distribution.

g.

```r
# Square each number in big.exp.draws.1
squared_values <- big.exp.draws.1^2

# Calculate mean of squared values
```

```r
mean_squared <- mean(squared_values)
mean_squared
```

```
## [1] 1.992403
```

Explanation for 5g):

Taking the square of each number in big.exp.draws.1 affects both the mean and the standard deviation. Specifically, if (X) is exponentially distributed with mean (μ) and standard deviation (σ), then (X^2) has a mean of ($\mu^2 + \sigma^2$). This results in an increase in the mean due to the squaring effect, while the standard deviation changes as per the mathematical properties of variance and standard deviation.