R Demonstration: Sampling Distribution and Central Limit Theorem

Abstract

After learning of **probability models**, especially 6 particular **random variables**, we need to learn how to sample from these **distributions** in `R`. This note contains related R codes for sampling from known distributions. *You should download all the files to one folder in your local drive and then set the working directary there in* `RStudio`.

1 How to sample from a given probability model?

Example: Sample many RVs from distribution with PMF

x	0	1	2	3
$f(x)$	0.1	0.2	0.4	0.3

We will use the following self constructed `R` function named as `sampling` to sample `n` observations from the given distribution `dist`. Here `dist` is a list with two components, values and probabilities respectively.

```
sampling = function(n, dist) {
    x = sample(dist$value, size = n, replace = TRUE, prob = dist$prob)
    return(x)
}
```

Here is the way to construct the list to store the distribution given in the example.

```
dist = list(value = c(0, 1, 2, 3), prob = c(0.1, 0.2, 0.4, 0.3))
```

Here is the way to use the `sampling` function to sample 10 observations from the above distribution.

```
sampling(10, dist)
##  [1] 3 2 2 1 2 3 2 2 3 2
```

Binomial Distribution Here is the way to sample 10 observations from the Binomial distribution $X \sim \text{Bin(n=25,p=0.2)}$.

```
rbinom(10, 25, 0.2)
##  [1] 6 5 2 4 8 1 6 7 4 5
```

Hypergeometric Distribution Here is the way to sample 10 observations from the Hypergeometric distribution $X \sim \text{H(N=25, n=8,k=12)}$.

```
rhyper(10, 12, 13, 8)
##  [1] 4 3 5 4 5 4 3 5 4 4
```

Poisson Distribution Here is the way to sample 10 observations from the Poisson distribution $X \sim \text{Poi}(\lambda = 4)$.

```
rpois(10, 4)
##  [1] 4 4 7 3 6 3 2 5 7 1
```

Uniform Distribution Here is the way to sample 10 observations from the Uniform distribution $X \sim \text{Unif}(a = 1, b = 3)$.

```
runif(10, 1, 3)
##  [1] 2.048749 1.526007 1.718589 2.915373 2.506263 2.827846 1.549536
##  [8] 2.596789 1.691025 1.533210
```

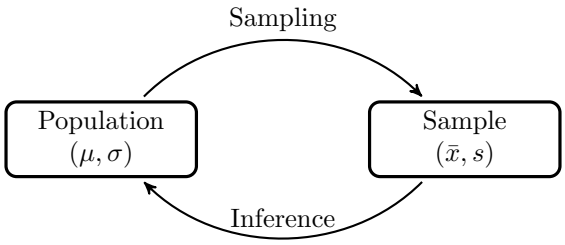
Exponential Distribution Here is the way to sample 10 observations from the Exponential distribution $X \sim \text{Exp}(\lambda = 4)$.

```
rexp(10, 4)
##  [1] 0.620971036 0.648196118 0.762712371 1.881625886 0.248192989
##  [6] 0.854821065 0.127458490 0.330754215 0.005406757 0.604029653
```

Normal Distribution Here is the way to sample 10 observations from the Normal distribution $X \sim \text{N}(\mu = 4, \sigma^2 = 2)$.

```
rnorm(10, 4, sqrt(2))
##  [1] 4.582033 4.663247 4.026501 4.559529 4.904616 4.072657 4.844096
##  [8] 3.561812 3.538142 5.298130
```

2 Law of Large Numbers (LLN)



LLN: Suppose a sequence of IID RVs $X_1, X_2, \ldots$ are observed, with common mean μ and common variance σ^2. Let $\bar{X}_n = \frac{1}{n}\sum_{i=1}^{n} X_i$ be the mean of the first n observations $X_1, \ldots, X_n$. Then $\bar{X}_n$ will be arbitrary close to μ as long as n large enough.

Example (cont.): Here we first calculate the mean and standard deviation of the distribution:

$$\mu = \sum_i x_i p_i, \sigma = \sqrt{\sum_i (x_i - \mu)^2 p_i}.$$

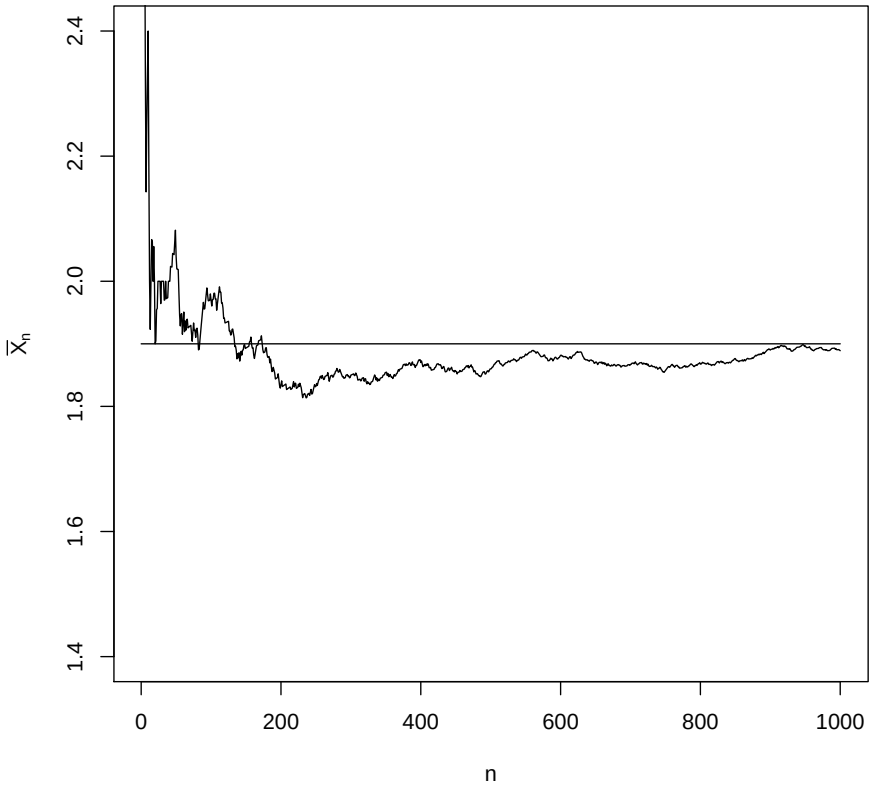
```
mu = sum(dist$value * dist$prob)
mu
## [1] 1.9
sdv = sqrt(sum((dist$value - mu)^2 * dist$prob))
sdv
## [1] 0.9433981
```

Now we want to track the sample averages with a sequence of sample size, say from 1 to 1000. We can use the following self constructed function.

```
mean_tracking = function(n, dist) {
    x = sample(dist$value, size = n, replace = TRUE, prob = dist$prob)
    s = cumsum(x)
    r = s/(1:n)
    return(r)
}
```

Now to see the relationship between the sample average and the sample size, we can draw the following plot:

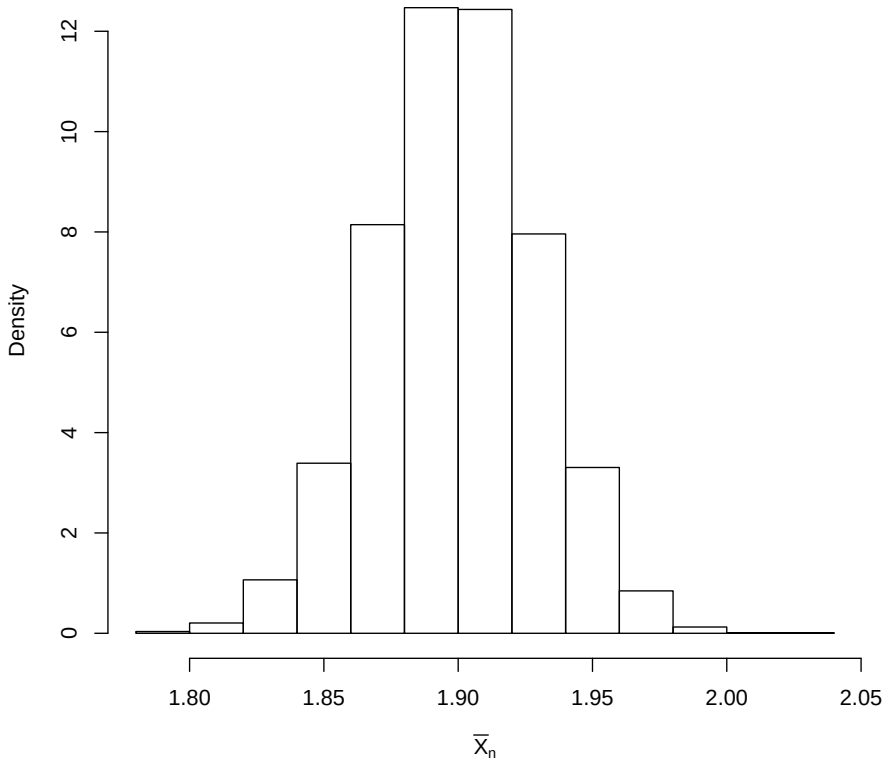
```
n = 1000
result = mean_tracking(n, dist)
plot(result, type = "l", ylim = c(mu - 0.5, mu + 0.5), ylab = expression(bar(X)[n]),
    xlab = "n")
lines(c(0, n), c(mu, mu))
```



3 Central Limit Theorem (CLT)

Since $\overline{X}_n$ is a random variable, are you curious about the sampling distribution of this random variable?

```
N = 10000
n = 1000
result = replicate(N, sampling(n, dist))
hist(colSums(result)/n, xlab = expression(bar(X)[n]), main = "", prob = TRUE)
```




```
mean(colSums(result)/n)
## [1] 1.89982
sd(colSums(result)/n)
## [1] 0.02990794
sdv/sqrt(n)
## [1] 0.02983287
```

Now I make a random sample from the population distribution and compute the sample mean .

> xbarstar=mean(sampling(n,dist))
> xbarstar
[1] 1.904
> mean((colSums(result)/n) < xbarstar) #(This is to approximate the probability of $\overline{X}_n$ being less than xbarstar)

[1] 0.5431

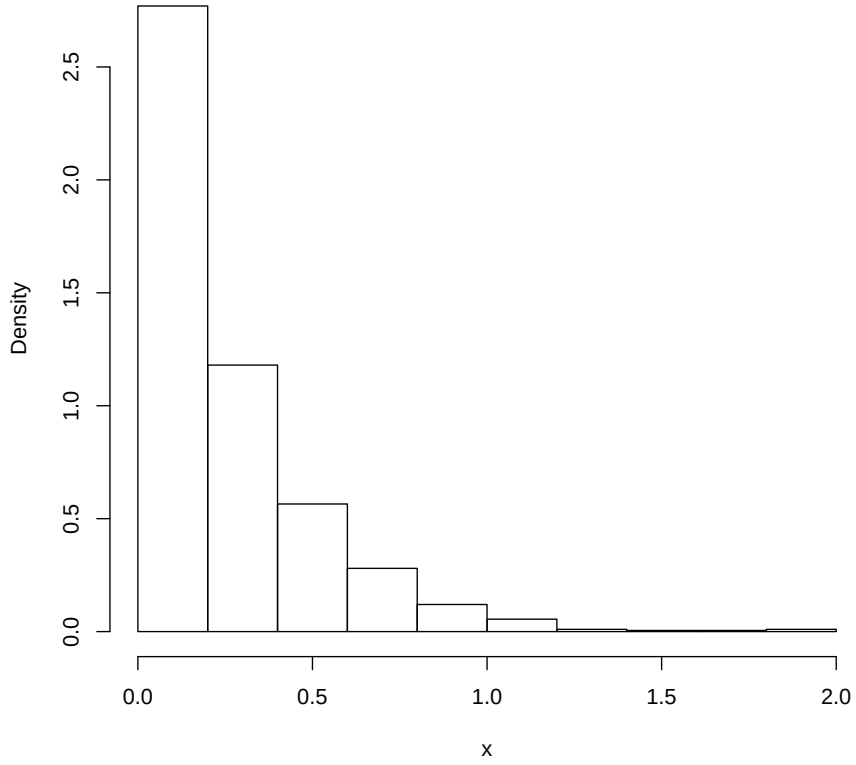
> mean((colSums(result)/n) < 1.8) #(This is to approximate the probability of $\overline{X}_n$ being less than 1.8)

[1] 3e-04

Example: Exponential Distribution We try the CLT again for this known continuous distribution $X \sim \text{Exp}(\lambda = 4)$.

First let's see the distribution of the $X \sim \text{Exp}(\lambda = 4)$ by plotting the histogram of the 1000 drawn sample points from $X \sim \text{Exp}(\lambda = 4)$.

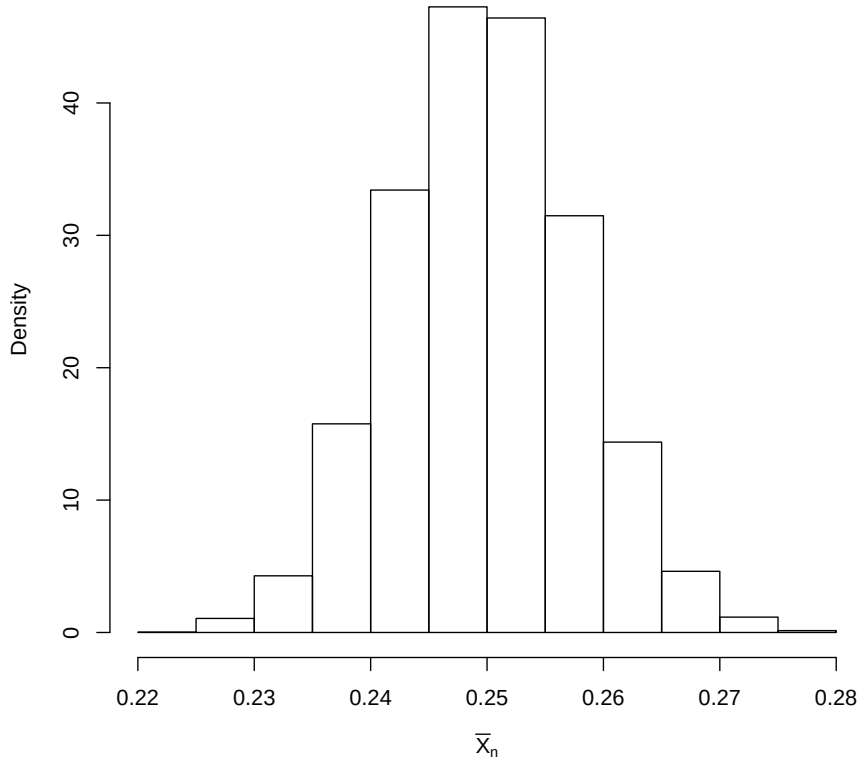
```
n = 1000
result = rexp(n, 4)
hist(result, xlab = "x", main = "", prob = TRUE)
```




```
mean(result)
## [1] 0.2532914
sd(result)
## [1] 0.2512896
```

Next let's draw $N = 10000$ samples with same sample size $n = 1000$, and plot the histogram of these $N = 10000$ sample means $\{\bar{x}_n^k, k = 1, 2, \cdots, N = 10000\}$ to see the distribution of $\bar{X}_n$.

```
N = 10000
result = replicate(N, rexp(n, 4))
hist(colSums(result)/n, xlab = expression(bar(X)[n]), main = "", prob = TRUE)
```




```
mean(colSums(result)/n)
## [1] 0.2499025
sd(colSums(result)/n)
## [1] 0.007930024
(1/4)/sqrt(n)
## [1] 0.007905694
```