Stochastic Process Fundamentals in R

2019DMB02
Madras School of Economics

Preface

The concepts of Stochastic processes are tightly integrated with the fields of Finance and Economics. For example, Stock returns are typically modeled as a sequence of random variables over time and hence it is important for us to grasp the concepts of probability distributions, Statistical Analysis and convergence in order to study their behaviour. Starting off with an introduction to random variables and various kinds of distribution, we will move on to slightly more involved concepts relating to the Central Limit Theorem and the Law of Large Numbers.

Random Variables

A random variable is a function over the outcomes of events in a sample space. This function maps outcomes to real numbers. Distribution of a random variable is a representation of all the possible values it might take, alongside the respective probabilities with which those values occur.

$$rv : \Omega \rightarrow R$$

A popular random variable is the **Bernoulli random variable** which is a variable taking on essentially two values. For example if we toss a coin, getting Heads of Tails are the possible outcomes and hence the result of one coin toss is a Bernoulli variable.

```
sample(c('H','T'), 1)
```

```
## [1] "T"
```

If we conduct these Bernoulli trials multiple times and define our random variable as the **number of successes (heads)** then what we have is essentially a **binomial** variable. The equation below

denotes : n = number of trials, k = number of successes.

$$f(k) = \binom{n}{k} p^k (1-p)^{n-k}$$

We can now compute the probability of getting 5 successes :

$$P(k = 5 | n = 10, p = 0.5)$$

```
# dbinom calulcates the pdf values

dbinom(x=5, size=10, prob=0.5)
```

```
## [1] 0.2460938
```

Therefore the probability of getting 5 Heads from 10 coin tosses is 0.24. We can also compute **cumulative density function** values with ease. For example we might want to calculate the given cumulative probabilities :

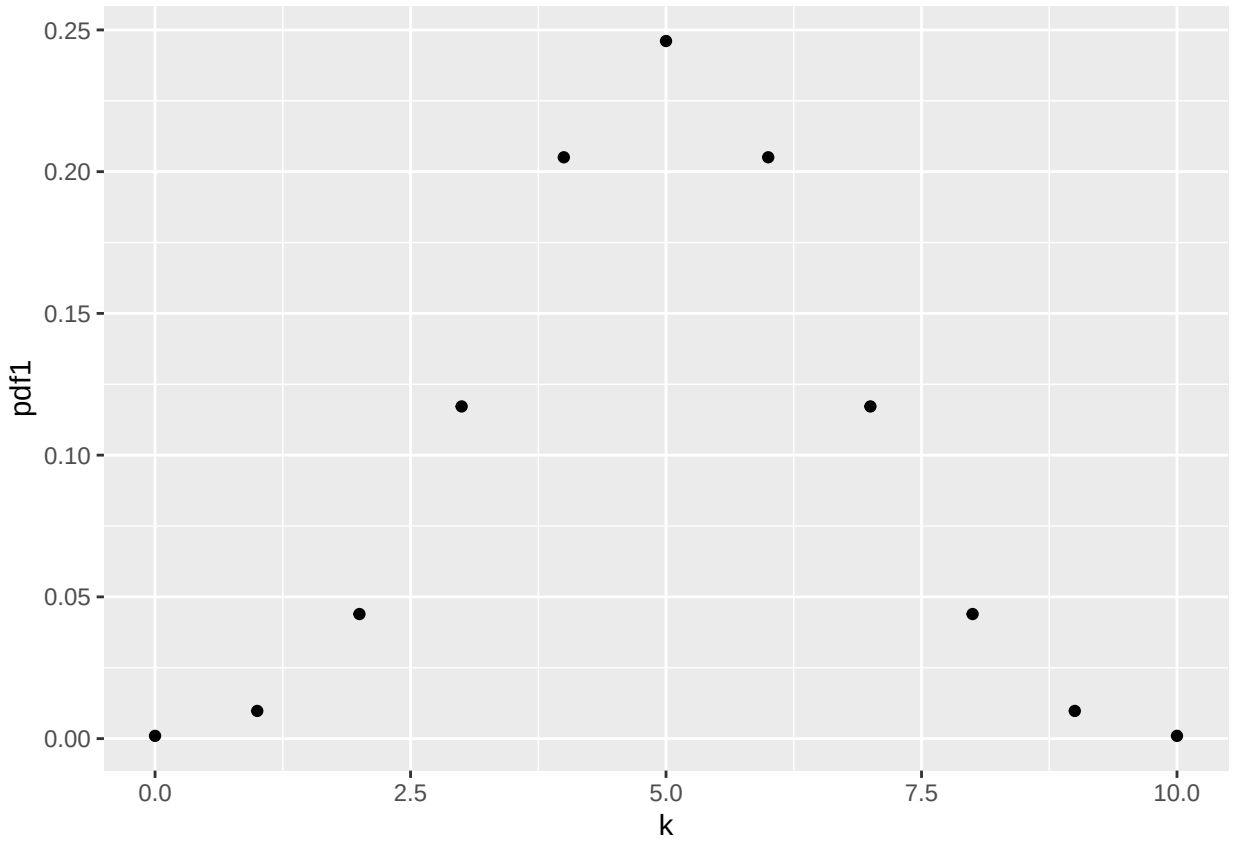
$$P(4 \leq k \leq 8) = P(k \leq 8) - P(k \leq 4)$$

```
# pbinom calculates the cdf values

pbinom(size=10, prob=0.5, q=8) - pbinom(size=10, prob=0.5, q=4)
```

```
## [1] 0.6123047
```

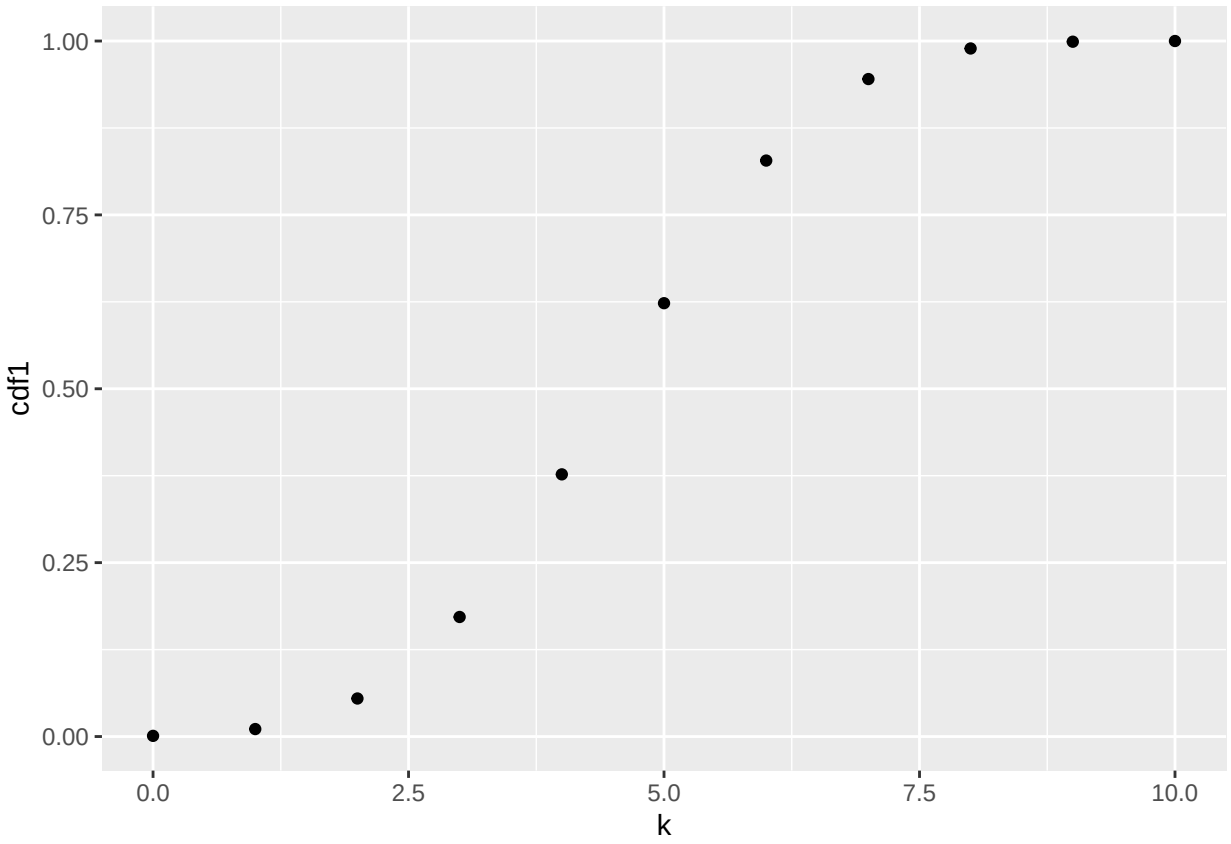
Therefore the probability that we get between 4 and 8 heads out of 10 coin tosses is about 0.61. We can visualize the **probability mass function** by essentially plotting the various probability values against corresponding realisations of the random variables (in this case the number of heads in 10 coin tosses).

```
k <- 0:10
pdf1 <- dbinom(x=k, size=10, p=0.5)

# ggplot plotting requires data frame objects,
# so converted the pdf array to a df

ggplot(as.data.frame(pdf1), aes(x=k, y=pdf1))+
  geom_point()
```



Similarly the CDF can be plotted as well.

```
cdf1 <- pbinom(size=10, prob=0.5, q=k)

ggplot(as.data.frame(cdf1), aes(x=k, y=cdf1))+
  geom_point()
```



Continuous random variables and moments

Continuous random variables are associated with **probability density functions** and take on a continuous set of values. The basic probability principle governing continuous random variables is given by:

$$P(a \leq X \leq b) = \int_a^b f_Y(y).dy$$

In addition, the expected value and variance of a continuous random variable is given by:

$$E(Y) = \int y f_Y(y).dy$$

$$VAR(Y) = \int (y - \mu_y)^2 f_Y(y).dy$$

Just to demonstrate these computations using code, we will take an example. Consider a PDF defined as:

$$f_X(x) = \frac{3}{x^4}, x > 1$$

Therefore we have the following conditions with respect to the PDF, expectation and the second moment:

$$\int_1^\infty \frac{3}{x^4}.dx = 1$$

$$E(X) = \int_1^{\infty} x.\frac{3}{x^4}.dx = \frac{3}{2}$$

$$E(X^2) = \int_1^{\infty} x^2\frac{3}{x^4}.dx = 3$$

We can compute the above formulations using some simple R functions.

```
# defining the functions

f <- function(x){ 3/x^4 }
g <- function(x){ x*f(x) }
h <- function(x){ x^2*f(x) }

# integrating

area_curve <- integrate(f, lower=1, upper=Inf)$value
area_curve
```

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

```
EX <- integrate(g, lower=1, upper=Inf)$value
EX
```

```
## [1] 1.5
```

```
VAR <- integrate(h, lower=1, upper=Inf)$value - EX^2
VAR
```

```
## [1] 0.75
```

Normal Distribution

The normal distribution is widely used throughout statistics and probability and is applied across disciples. Its PDF is given by:

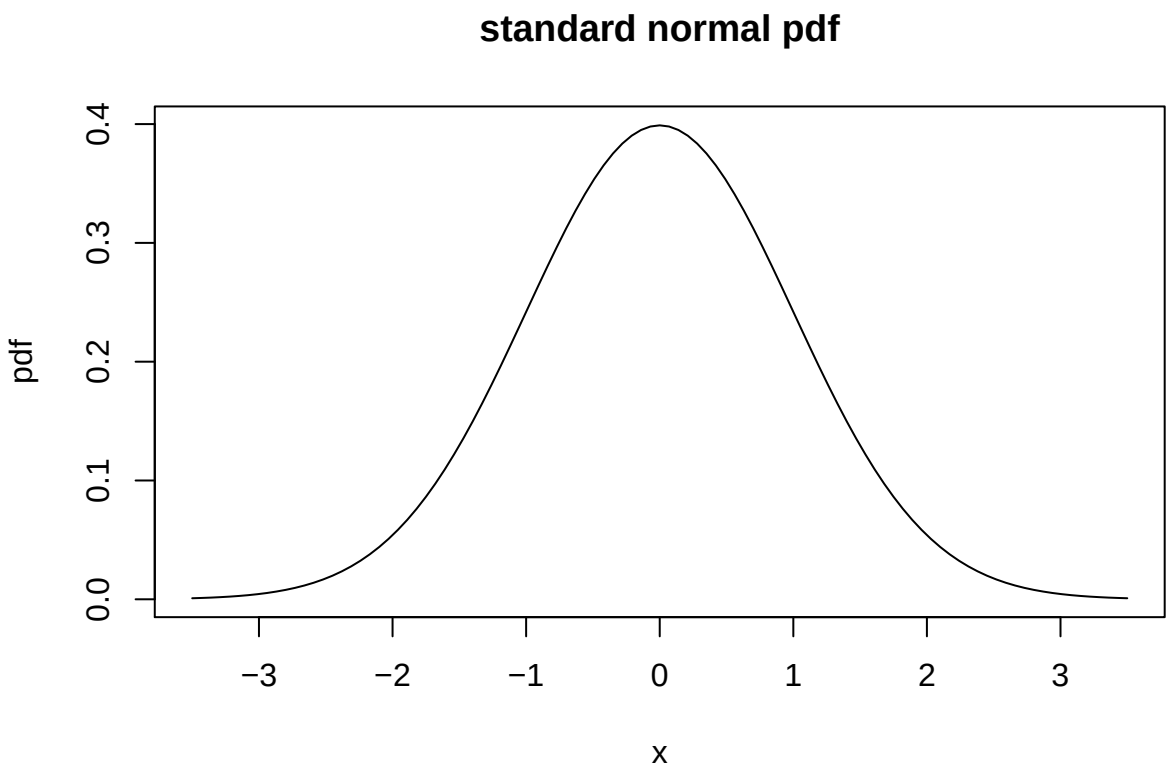
$$N(\mu, \sigma^2) = f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\{\frac{-(x-\mu)^2}{2\sigma^2}\}$$

For a standard normal random variable, the mean is 0 and variance is 1 and its corresponding PDF is represented as:

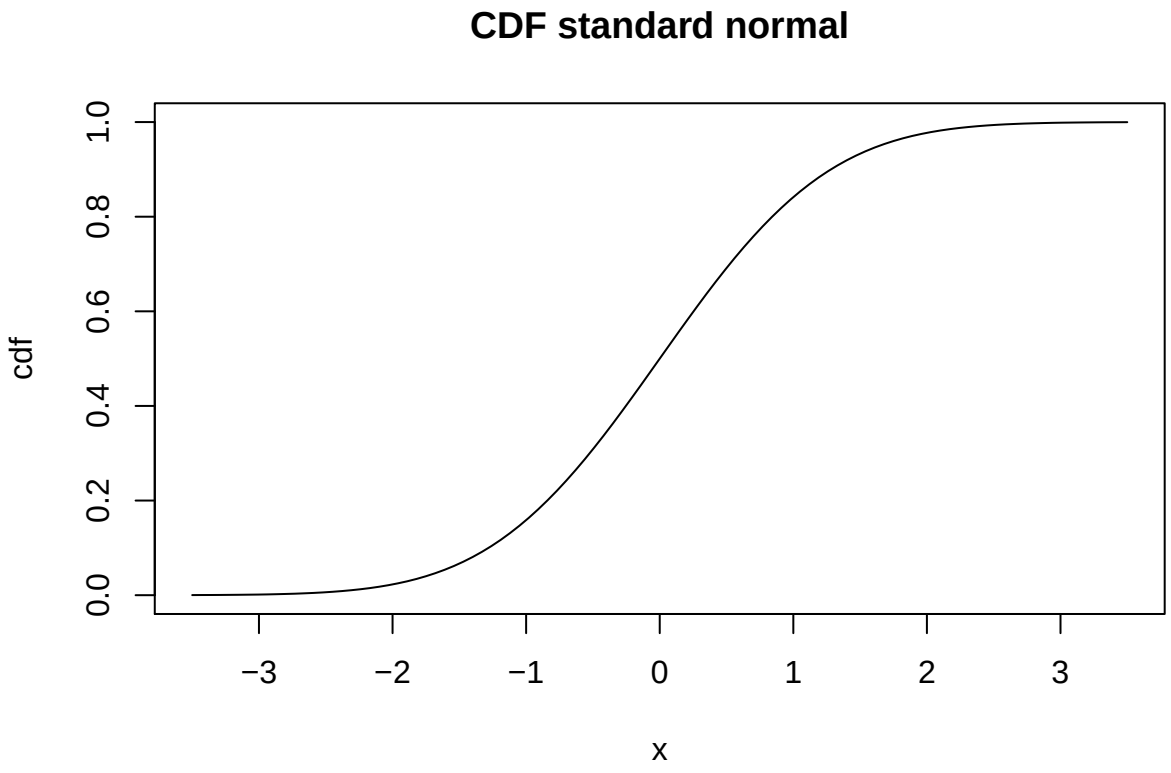
$$\Phi(x) = P(Z \leq x)$$

Below we can see how the PDF and CDF of a normal random variable are plotted:

```
curve(dnorm(x), xlim=c(-3.5,3.5),
      ylab='pdf', main="standard normal pdf")
```




```
curve(pnorm(x), xlim=c(-3.5,3.5), ylab='cdf', main='CDF standard normal')
```



Sampling distributions

In modeling, we are often dealing with samples of random variables. What is then generated is essentially a **sampling distribution**. Now before we move on to the various properties of these sampling distributions we must note that the sample mean is given as:

$$\bar{Y} = \frac{1}{n}\sum_{i=1}^{n} Y_i$$

Now considering that Y(1 till n) are a set of identically and independently distributed random variables having finite mean and variance, we could then formulate the mean and variance of the sample mean of these iid random variables as:

$$E(\bar{Y}) = \mu_Y$$

$$VAR(\bar{Y}) = \frac{\sigma_Y^2}{n}$$

So we can make the following statements:

$$if\, Y_i \sim N(\mu_Y, \sigma_Y^2)$$

$$then\, \bar{Y} \sim N(\mu_Y, \sigma_Y^2/n)$$

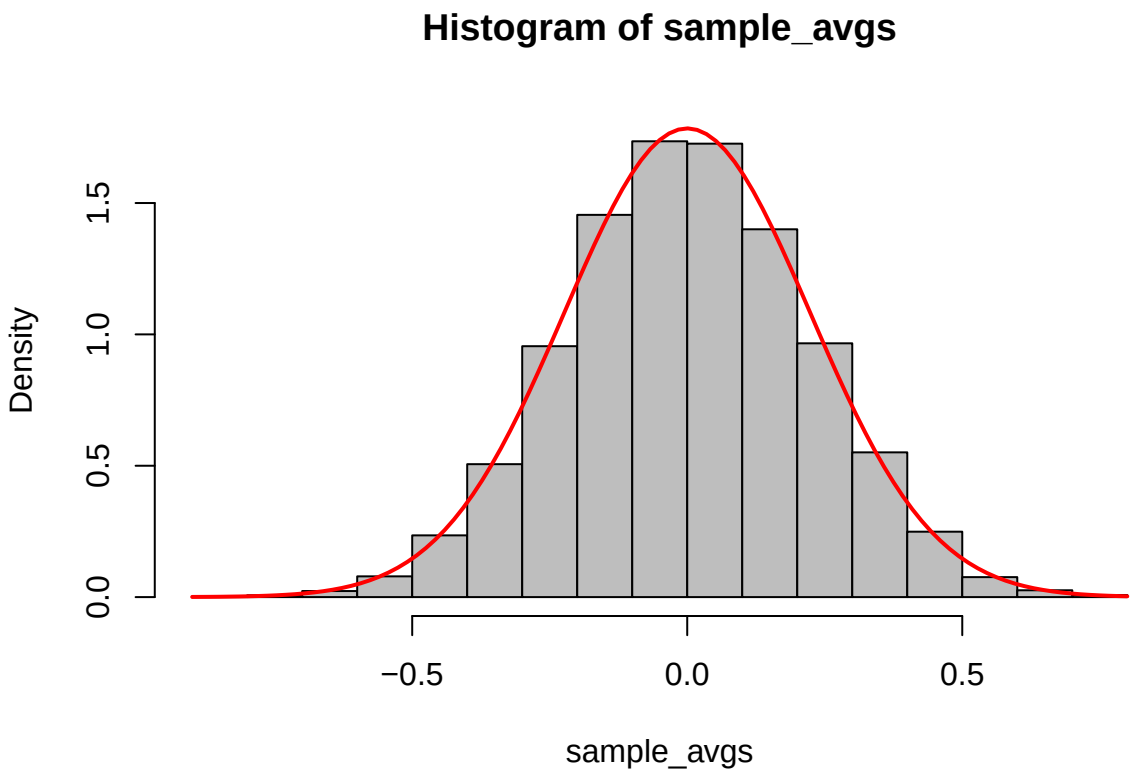
Now we will essentially choose a sample size (n) and draw that sample size repeated number of times, essentially simulating our experiment many times, so as to obtain many sample averages. We are computing many sample averages since we are interested in finding the distribution behaviour of this sample average.

```
# sample size and number of repititions

n <- 20
reps <- 10000

# sample 20 random variables, 10000 times
# rnorm pulls out random numbers from standard normal dist

samples <- replicate(reps, rnorm(n))

# get the vector of 10000 sample means

sample_avgs <- colMeans(samples)
head(sample_avgs)
```

```
## [1]  0.22800546  0.23212483  0.17731303  0.03199272  0.10987015 -0.13202014
```

Now we will plot the distribution of these samples in terms of a binned histogram, which will tell us the frequency of multiple ranges of sample mean values obtained from the 10000 repititions of our experiment. Then we will overlay the theoretical distribution to justify our concepts.

```r
hist(sample_avgs, ylim=c(0, 1.8), col='gray',
     freq=F, breaks=20)
curve(dnorm(x, sd=1/sqrt(n)), col='red', lwd='2', add=T)
```



Our theoretical formulation is indeed justified since we can clearly see the similarities in the two distributions.

Weak Law of Large Numbers

This law basically is a condition concerning convergence in probability. From the above examples, can state that the sample average converges in probability to the population mean as the sample size becomes infinitely large. The notations to describe this statement is as follows:

$$\bar{Y} \xrightarrow{P} \mu_Y$$

$$\lim_{n\to\infty} P(|\bar{Y} - \mu_Y| \geq \epsilon) = 0$$

We will now see with an example as to how our sample average would turn out to be quite an accurate representation of the true population mean. Consider sampling from a Bernoulli random variable wherein we are tossing a coin. Now we will essentially show that the fraction of times we get a heads among our samples, will be approximately equal to the true probability of getting a heads, as our sample size increases. Following are the definitions of our experiment.

$$P(Y_i) = \begin{cases} 1,\ p \\ 0,\ 1-p \end{cases}$$

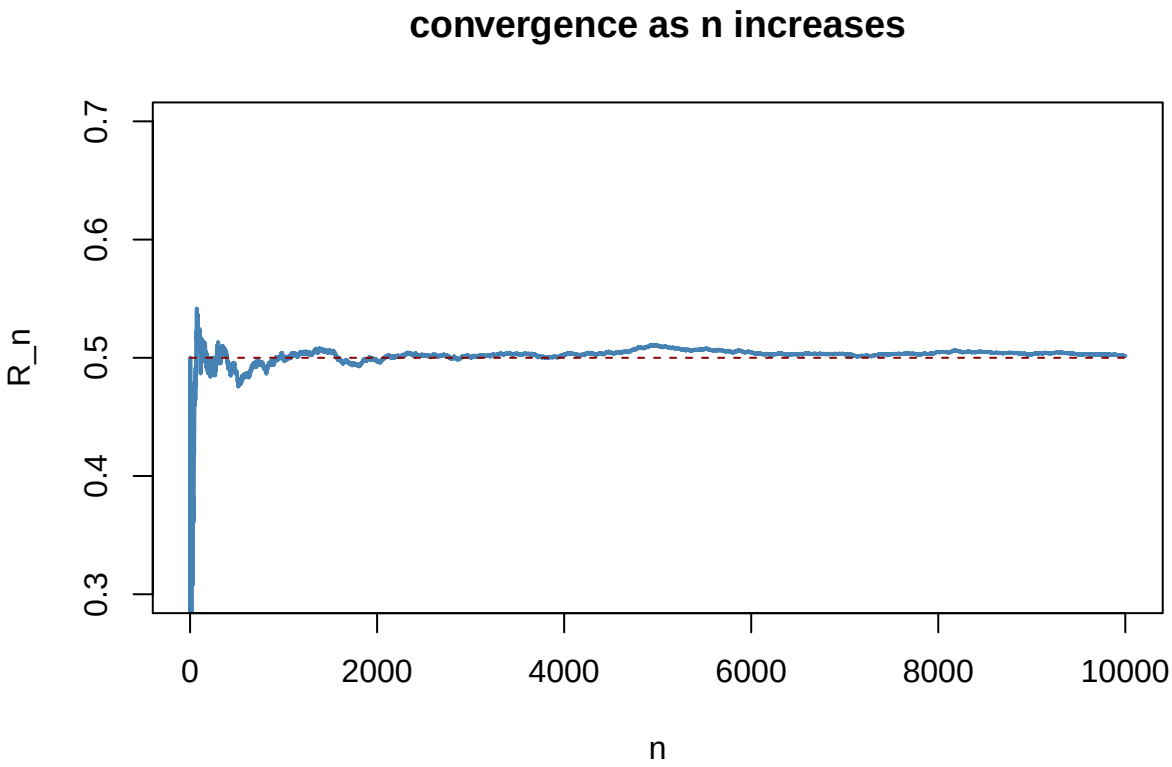
We will assume that we are tossing a fair coin and hence p is 0.5. Note that the proportion of heads in our experiment would be given by:

$$R_n = \frac{1}{n}\sum_{i=1}^{n} Y_i$$

We will prove the weak law of large numbers now, which states that the proportion of heads converges in probability to the population mean which is 0.5, which in turn is the true probability of getting heads in a single toss.

$$R_n \xrightarrow{P} \mu_Y = 0.5,\ as,\ n \to \infty$$

```
set.seed(1)

# initialize number of coin tosses and sample

N = 10000
Y = sample(0:1, N, replace=T)

# computing sample mean

S <- cumsum(Y)
R_n <- S/(1:N)

# now plot the fraction vs number of tosses

plot(R_n, ylim=c(0.3,0.7), type='l',
     col='steelblue', lwd=2, xlab='n',
     ylab='R_n', main='convergence as n increases')

# adding reference line for true prob

lines(c(0,N), c(0.5,0.5), col='darkred', lty=2, lwd=1)
```



We can clearly see that as the sample sizes is increasing, the proportion of heads, or our sample average is tending to the true probability.

Central Limit Theorem

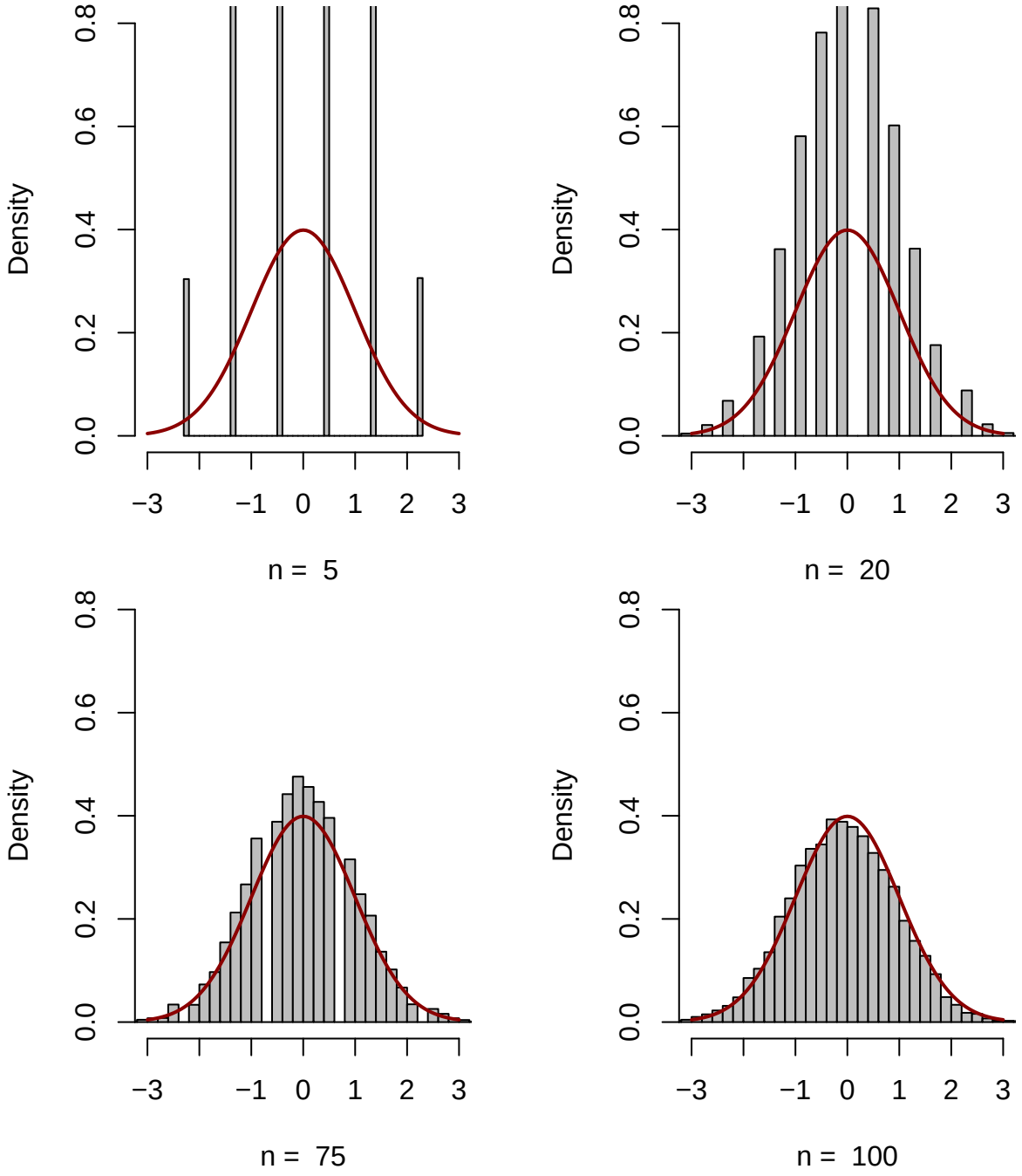
If we sample identically and independently distributed random variables with finite mean and variance, then the CLT states that as sample size tends to infinity, the distribution of the **normalized sample average** converges in distribution to the **standard normal distribution**. This can be denoted as follows:

$$\frac{\bar{Y} - \mu}{\sigma/\sqrt{n}} \xrightarrow{D} N(0,1)$$

If we consider our previous example itself of tossing a coin, this theorem tells us that the distribution of the normalized sample mean of the Bernoulli random variables would be approximately represented by the standard normal distribution. To show this, we will sample repeatedly many times, with varying sample sizes from the Bernoulli distribution and compute corresponding sample averages. Then plot the sample averages as a histogram and compare it with the standard normal distribution. We will then get a clear idea of how these histograms would increasingly be shaped like a standard normal distribution as the sample size increases.

```
par(mfrow=c(1,2))

# number of reps and sample sizes

reps <- 10000
```

```r
sample_sizes <- c(5,20,75,100)

set.seed(123)

# first loop over sample size vector

for(n in sample_sizes){

  # initialize sample mean and standard sample mean

  sample_mean <- rep(0, reps)
  stdsample_mean <- rep(0, reps)

  # now loop over repititions

  for(i in 1:reps){
    x <- rbinom(n, 1, 0.5)
    sample_mean[i] <- mean(x)
    stdsample_mean[i] <- sqrt(n)*(mean(x)-0.5)/0.5
  }

  hist(stdsample_mean, col='gray', freq=FALSE,
     breaks=40, xlim=c(-3,3), ylim=c(0,0.8),
     xlab=paste('n = ',n),
     main = "")

  curve(dnorm(x), lwd=2, col='darkred', add=TRUE)
}
```



We can clearly see that as the sample size is increasing, the distribution is tending to standard normal.

References

1. *Econometrics with R*
2. *Data Analysis for the Life sciences - Rafael A Irizarry*
3. *Learning Statistics with R*
4. *Stochastic Processes Lecture notes*