Chapter 3

Random Number Generation & Curve Fitting

3.1 Using R

► A function $f(x)$ is said to be a polynomial of degree 1 if $f(x) = a_0x + a_1$ with $a_0 \neq 0$.

3.1.1 Plot Distributions

■ Usually in R, we can find $f(x)$ or, $P(X = x)$ of a random variable X for various distributions.

► For discrete distribution example: i) '*dbinom*(x, n, p)' finds $P(X = x)$ with $X \sim Bin(n, p)$; ii) '*dpois*(x, λ)' finds $P(X = x)$ with $X \sim Poi(\lambda)$; iii) '*dnbinom*(x, r, p)' finds $P(X = x)$ with $X \sim$ *Negative Binomial*(r, p); iv) '*dgeom*(x, p)' finds $P(X = x)$ with $X \sim Geo(p)$.

► For continuous distribution example: i) '*dnorm*(x, μ, sd)' finds $f(x)$ with $X \sim N(\mu, sd)$; ii) '*dgamma*$(x, shape, rate)$' finds $f(x)$ with $X \sim Gamma(shape, rate)$; iii) '*dchisq*$(x, n)$' finds $f(x)$ with $X \sim \chi^2_n$; iv) '*dbeta*$(x, shape1, shape2)$' finds $f(x)$ with $X \sim Beta(shape1, shape2)$.

Example 3.1. *Plot the graph of binomial distribution with parameters* $n = 10, p = 0.2$ *i.e.* $Bin(10, 0.2)$. *Here, X is a random variable denotes the number of success in n^{th} trial then the probability mass function (pmf) of X is* $\boxed{P(X = x) = \binom{n}{x}p^x q^{n-x},\ x = 0, 1, \cdots, n}$; *$p$ is the success probability and $q = 1 - p$. Here we say that $X \sim Bin(n, p)$.*

► $\boxed{\textit{R Code}}$:

```
n=10
```

```
p=0.2
x=0:n # Range of Binomial
f=dbinom(x, n, p, log = FALSE) # Value of PMF
plot(x,f,type="h",xlim=c(min(x),max(x)),ylim=c(0,1),lwd=2,col="blue",ylab="P(X=x)")
points(x,f,pch=16,cex=1,col="dark red")
```

► $\boxed{\textit{R Plot}}$: *See the plot Fig. 3.1.*

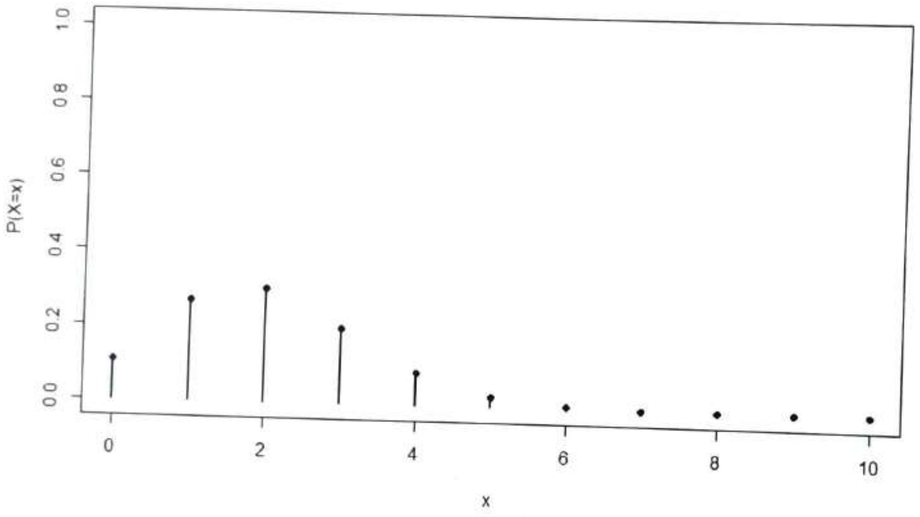


Figure 3.1: PMF of Binomial Distribution for $n = 10,\ p = 0.2$.

[Do It Yourself] 3.1. *Plot the graph of Poisson distribution with parameters $\lambda = 2$ i.e. $Poi(2)$. Here, X is a random variable denotes the number of occurrence in the interval of interest then the probability mass function (pmf) of X is* $\boxed{P(X = x) = \frac{e^{-\lambda}\lambda^x}{x!},\ x = 0, 1, \cdots, \infty}$; *$\lambda$ is the average number of events in the given interval. Here we say that $X \sim Poi(\lambda)$.*

[Do It Yourself] 3.2. *Plot the graph of Negative Binomial distribution with parameters $r = 2,\ p = 0.3$ i.e. $NB(2, 0.3)$. Here, X is a random variable denotes the number of failures that precede the r^{th} success then the probability mass function (pmf) of X is* $\boxed{P(X = x) = \binom{x+r-1}{x} p^r q^x,\ x = 0, 1, \cdots, \infty}$; *$r \geq 1$. Here we say that $X \sim NB(r, p)$.*

[Do It Yourself] 3.3. *Plot the graph of Geometric distribution with parameters $p = 0.3$ i.e. $Geo(0.3)$. Here, X is a random variable denotes the number of failures that precede the 1^{st} success then the probability mass function (pmf) of X is* $\boxed{P(X = x) = pq^x,\ x = 0, 1, \cdots, \infty}$. *Here we say that $X \sim Geo(p)$.*

■ Geometric distribution is a special case of negative binomial distribution for $r = 1$.

Example 3.2. *Plot the graph of normal distribution with parameters* $\mu = 0.4, sd = \sigma = 1.2$ *i.e.* $N(0.4, 1.2)$. *Here,* X *is a random variable with probability density function (pdf) of* X *is* $\boxed{f(x) = \frac{1}{\sqrt{2\pi}\sigma}e^{-\frac{1}{2\sigma^2}(x-\mu)^2},\ -\infty < x < \infty}$; μ *is mean and* σ *is sd. Here we say that* $X \sim N(\mu, \sigma)$.

► *R Code*:

```
x=seq(-5,5,0.1) # Generate sequence of x values
f=dnorm(x, mean = 0.4, sd = 1.2) # Generate f(x)
plot(x,f,type="l",xlim=c(min(x),max(x)),ylim=c(0,1),lwd=2,col="blue",ylab="f(x)")
```

► *R Plot*: *See the plot Fig. 3.2.*

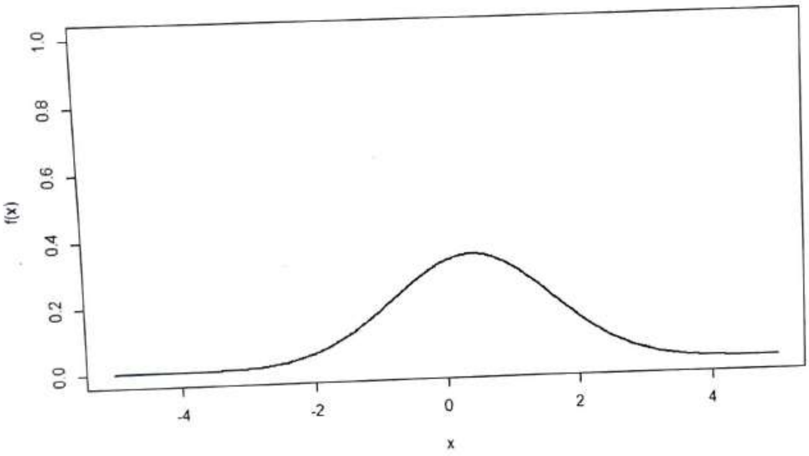


Figure 3.2: PDF of Normal Distribution for $\mu = 0.4,\ sd = \sigma = 1.2$.

Example 3.3. *Plot the multiple graphs of normal distribution from* $N(0.4, 1.2)$, $N(0.4, 0.6)$ *and* $N(0.4, 1.8)$.

► *R Code*:

```
x=seq(-5,5,0.1) # Generate sequence of x values
f=dnorm(x, mean = 0.4, sd = 1.2) # Generate f(x)
f1=dnorm(x, mean = 0.4, sd = 0.6) # Generate f(x)
f2=dnorm(x, mean = 0.4, sd = 1.8) # Generate f(x)
plot(x,f,type="l",xlim=c(min(x),max(x)),ylim=c(0,1),lwd=2,col="blue",ylab="f(x)")
lines(x,f1,type="l",xlim=c(min(x),max(x)),ylim=c(0,1),lwd=2,col="red")
lines(x,f2,type="l",xlim=c(min(x),max(x)),ylim=c(0,1),lwd=2,col="green")
legend(3,0.8,legend=c("N (0.4,1.2)","N (0.4,0.6)","N (0.4,1.8)"),
col=c("blue","red","green"),lty=c(1,1,1))
```

▶ $\boxed{\textit{R Plot}}$: *See the plot Fig. 3.3.*

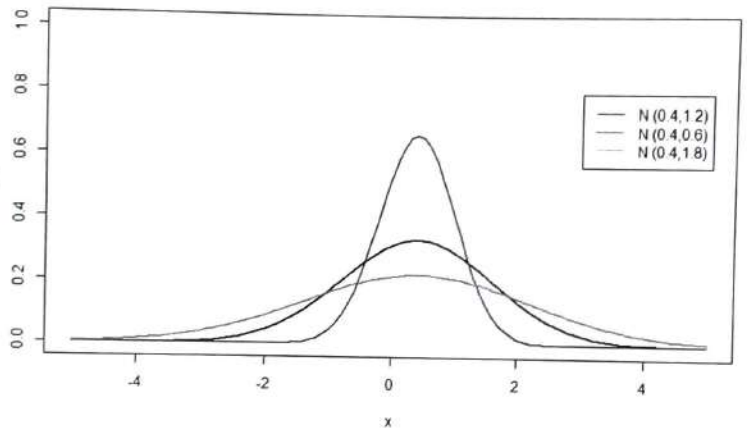


Figure 3.3: PDF of Normal Distribution for $\mu = 0.4,\ sd = \sigma = 1.2$.

[Do It Yourself] 3.4. *Plot the graph of gamma distribution with parameters shape* $=\alpha = 2$, *rate* $= \beta = 0.5$ *i.e.* $Gamma(2, 0.5)$. *Here,* X *is a random variable with probability density function (pdf) of* X *is* $\boxed{f(x) = \frac{1}{\Gamma(\alpha)\beta^{\alpha}} x^{\alpha-1} e^{-x/\beta},\ x > 0}$; α *is sahpe and* β *is rate. Here we say that* $X \sim Gamma(\alpha, \beta)$. *Plot the multiple graphs of gamma distribution from* $Gamma(2, 0.5)$, $Gamma(2, 2.5)$ *and* $Gamma(1.2, 2)$.

[Do It Yourself] 3.5. *Plot the graph of Chi-square distribution with parameters df* $= n = 3$ *i.e.* χ^2_3. *Here,* X *is a random variable with probability density function (pdf) of* X *is* $\boxed{f(x) = \frac{1}{\Gamma(\frac{n}{2})2^{n/2}} x^{n/2-1} e^{-x/2},\ x > 0}$; n *is degrees of freedom. Here we say that* $X \sim \chi^2_n$. *Plot the multiple graphs of Chi-square distribution from* χ^2_3, χ^2_6 *and* χ^2_8.

[Do It Yourself] 3.6. *Plot the graph of Beta distribution with parameters shape1* $= a = 1.1$, *shape2* $= b = 1.3$ *i.e.* $Beta(1.1, 1.3)$, *(Type I). Here,* X *is a random variable with probability density function (pdf) of* X *is* $\boxed{f(x) = \frac{1}{B(a,b)} x^{a-1}(1-x)^{b-1},\ 0 < x < 1}$; a, b *are shape parameters. Here we say that* $X \sim Beta(a, b)$. *Plot the multiple graphs of Beta distribution from* $Beta(1.5, 1.4)$, $Beta(1.1, 1.4)$ *and* $Beta(1.5, 1.1)$.

■ Exponential distribution is a special case of Gamma distribution for $\alpha = 1$.
■ You can easily plot any distribution function i.e. $F(x) = P(X \leq x)$ instead of PMF/

PDF by using 'q' instead of 'p' command. For example, 'qbinom','qnorm' for binomial and normal distribution respectively.

► Also theoretically the range of x is $(-\infty, \infty)$, so just use feasible range for a distribution.

3.1.2 Random Number Generation

■ Usually in R, we can generate random numbers from various distributions. This is a very strong feature of 'R Programming'.

► For discrete distribution example: i) '$rbinom(y, n, p)$' generate y number of random samples from $X \sim Bin(n, p)$; ii) '$rpois(y. \lambda)$' generate random samples from $X \sim Poi(\lambda)$; iii) '$rnbinom(y, r, p)$' generate random samples from $X \sim Negative\ Binomial(r, p)$; iv) '$rgeom(x, p)$' generate random samples from $X \sim Geo(p)$.

► For continuous distribution example: i) '$rnorm(y, \mu, sd)$' generate random samples from $X \sim N(\mu, sd)$; ii) '$rgamma(y, shape, rate)$' generate random samples from $X \sim Gamma(shape, rate)$; iii) '$rchisq(y, n)$' generate random samples from $X \sim \chi^2_n$; iv) '$rbeta(y, shape1, shape2)$' generate random samples from $X \sim Beta(shape1, shape2)$; v) '$runif(y, a, b)$' generate random samples from $X \sim U[a, b]$.

Example 3.4. *Generate* 50 *random samples from binomial distribution with parameters* $n = 10, p = 0.2$ *i.e.* $Bin(10, 0.2)$. *Also draw the histogram.*

► $\boxed{R\ Code}$:

```
y=rbinom(50, 10, 0.2) # Generate 50 Random Samples from Bin(10,0.2)
y
hist(y,col='green') # Plot may be different for each run
```

► $\boxed{R\ Plot}$: *See the plot Fig. 3.4.*

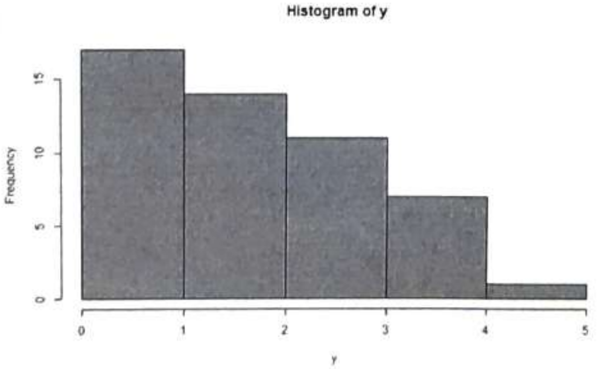


Figure 3.4: Histogram of $Bin(10, 0.2)$ for 50 random samples.

[Do It Yourself] 3.7. *Generate 100 random samples from $Poi(2)$, $NB(2,0.3)$, $Geo(0.3)$. Also draw the respective histograms.*

Example 3.5. *Generate 500 random samples from Normal distribution with parameters $\mu = 1$, $sd = \sigma = 1.2$ i.e. $N(1,1.2)$. Also draw the histogram.*

► *R Code*:

```
y=rnorm(500, 1, 1.2) # Generate 500 Random Samples from N(1,1.2)
y
round(y,3) # To round off numbers
hist(y,col='green') # Plot may be different for each run
hist(y,col='red',breaks=15) # Different intervals
```

► *R Plot*: *See the plot Fig. 3.5.*

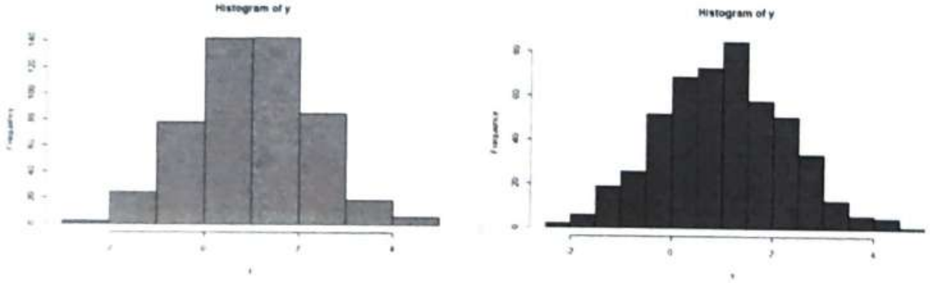


Figure 3.5: Histogram of $N(1,1.2)$ for 500 random samples: a) Default Breaks b) 15 Breaks.

[Do It Yourself] 3.8. *Generate 1000 random samples from $Gamma(2,2.5)$, χ^2_3, $Beta(1.1,1.4)$, $U[2,5]$. Also draw the respective histograms.*

3.1.3 Sampling Procedures

■ Population is a large group and usually we want to draw conclusion about it. Practically it is quite difficult to work with the population due to various associated costs.

■ Sample is a part of the population based on which you will draw the conclusion about the population.

► As we will use the sample instead of population, it is very important to know how we will draw the sample from a population.

► It is also important how many samples (i.e. sample size) we will draw from the population to get a better result. However, it is beyond the scope of this book.

► The sampling frame is the actual list of individuals that the sample will be drawn from. Ideally, it should include the entire target population. For Example, if you have a 1 sq.

km. garden consists with coconut, mango, apple and palm tree and you want to sample mango, coconut tree then the sampling frame will be the whole list of mango, coconut tree.

■ Here we will use R to draw n number of samples using various sampling methods.

■ Usually we use two types of sampling procedure i) Probability sampling methods (e.g. Simple random sampling, Systematic sampling, Stratified sampling, Cluster sampling) and ii) Non-probability sampling (e.g. Convenience sampling, Voluntary response sampling, Purposive/ Judgement sampling, Quota sampling).

■ Simple Random Sampling (SRS): Here we will draw sample from whole population and each member will select with equal probability.

▶ SRS can be drawn in two ways: a) With replacement i.e. SRSWR and b) Without replacement i.e. SRSWOR.

Example 3.6. *Suppose there are* 100 *people numbered from* 1 *to* 100. *You want select* 20 *among them to give movie ticket. Use SRSWR (one people may get multiple tickets) and SRSWOR to select these lucky people.*

▶ *R Code*:

```
y1=sample(c(1:100),size=20,replace =T) # Sample using SRSWR
y1
y2=sample(c(1:100),size=20,replace =F) # Sample using SRSWOR
y2
```

[Do It Yourself] 3.9. *Suppose there are* 10 *plant named: Areca Palm, English Ivy, Indian Basil, Spider Plant, Snake plant, Weeping Fig, Azalea, Dracaena, Aloe Vera and Small Coconut. Select four of these by using SRSWR and SRSWOR.*

■ Systematic Sampling: Just randomly draw a starting point and then select rest of the element at a regular interval.

▶ The best way is to draw random samples from Systematic Sampling is to use Circular Systematic Sampling. For example, you want to draw $n = 12$ samples from population of size $N = 90$. Then just draw any number r from 1 to N and find a number $k = N/n = 90/12 \approx 7$ (nearest integer). The samples are $r, r + k, \cdots$. Note that, if any sample cross the value 90 then we will divide that number by 90 and take '$remainder + 1$'.

Example 3.7. *Suppose there are* 90 *people numbered from* 1 *to* 90. *You want to select* 17 *among them to give movie ticket. Use Circular Systematic Sampling to select these lucky people.*

▶ *R Code*:

```
N=90
n=17
r=sample(c(1:90),size=1)  # One sample using SRS
```

```
r
k=N/n
k=5  # Check the above k and write the nearest integer
samp=seq(r,r+k*n,k)  # Generate systematic sample
samp=samp%%90+1  # Use modulo function i.e. ramainder
samp
```

[Do It Yourself] 3.10. *Suppose there are* 1000 *trees with numbering. Select* 140 *of these by using Circular Systematic Sampling.*

■ Stratified Sampling: Divide the whole population into strata (or, homogenous subsets) based on similar characteristics and then use SRS or, Systematic RS from each strata.

► Sample size may not be equal for each strata.

Example 3.8. *Suppose there are* 800 *students in Zoology. Among them* 500 *are girls and* 300 *are boys. You want to select* 50 *among them with equal number of boys and girls for a scholarship. Use Stratified Sampling to select these lucky people.*

► *R Code*:

```
fem=sample(c(1:500),size=25,replace =F) # Sample for Strata 1
fem
mal=sample(c(1:300),size=25,replace =F) # Sample for Strata 2
mal
```

[Do It Yourself] 3.11. *Suppose there are* 1200 *students in Zoology. Among them* 800 *are girls and* 400 *are boys. You want to select* 60 *among them for a conference with same proportion reflects the population. Use Stratified Sampling to select these lucky people.*

[Do It Yourself] 3.12. *Suppose you want to study the characteristics of students with various annual family income. Then you may create strata (groups) according to the annual family income i.e. based on some income range, draw SRS.*

■ Cluster Sampling: Divide the whole population into subgroups based on similar characteristics like the population (i.e. cluster) and then use RS to select one or, more cluster.

► In single-stage cluster sampling, all members of the chosen clusters are then included in the study.

► In two-stage cluster sampling, first select the cluster and then select its member randomly (instead of all like single-stage).

► In multi-stage cluster sampling, is an extension of two-stage sampling.

► For example, A research firm in the UK conducted a survey in which it divided the country into its counties and randomly selected some of these counties as a cluster sample (the first stage of sampling). Each county was then divided into its towns, and areas were chosen at random from each town (the second stage of sampling). Finally, within each town, each town was divided into small areas and households were selected at random from each area. These households formed the sample population for the research study (third stage of sampling).

Example 3.9. *The company has offices in* 50 *cities across the country (all with roughly the same number of employees in similar roles). Due to travel restriction, it is not possible to visit every office to collect the data. Now using R, find* 10 *office (stage 1 cluster). Then from these* 10 *offices select* 5 *employees from each office (stage 2 cluster).*

3.1.4 Fitting Exponential Curve

▶ The exponential regression model is defined as $y = ae^{bx}\epsilon$. Here a, b are unknown parameters will be estimated from the data.
▶ Taking log both sides we have, $\boxed{\ln(y) = a' + bx + \epsilon'}$, where $a' = \ln(a)$, $\epsilon' = \ln(\epsilon)$. Now we can fit simple linear regression with $\hat{a} = e^{\hat{a}'}$.

Example 3.10. *Fit an exponential curve $y = ae^{bx}$ based on the data: X: 0 0.01 0.03 0.05 0.07 0.09 0.11 0.13 0.15 0.17 0.19 0.21; Y: 1 1.03 1.06 1.38 2.09 3.54 6.41 12.6 22.1 39.05 65.32 99.78. Draw the regression line over the scatter plot.*

▶ $\boxed{\textit{R Code}}$:

```
x=c(0,0.01,0.03,0.05,0.07,0.09,0.11,0.13,0.15,0.17,0.19,0.21)
y=c(1,1.03,1.06,1.38,2.09,3.54,6.41,12.6,22.1,39.05,65.32,99.78)
y1=log(y)
fit=lm(y1 ~ x) # Lin Reg Fit
a=exp(fit$coefficients[1])
b=fit$coefficients[2]
summary(fit)
plot(x,y,xlab = 'X', ylab='Y',ylim=c(0,100),col='blue',main='Exponential Fit',
cex=1.2,pch=16)
x1 = seq(min(x),max(x)+0.1,0.01)
y1 = a*exp(b*x1)
lines(x1, y1, lty =1, col = "darkgreen", lwd =2)
```

▶ $\boxed{\textit{R Plot}}$: *See the plot Fig. 3.6.*

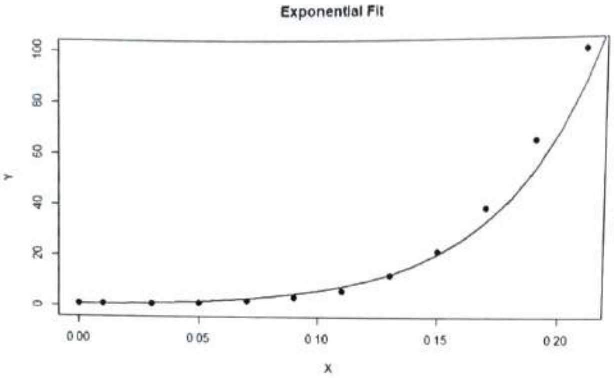


Figure 3.6: Fitting exponential curve over data.

[Do It Yourself] 3.13. *Fit an exponential curve $y = ae^{bx}$ based on the data: X: 0 0.01 0.03 0.05 0.07 0.09 0.11 0.13 0.15 0.17 0.19 0.21; Y: 1 1.03 1.06 1.38 2.09 3.54 6.41 12.6 22.1 39.05 65.32 99.78. Draw the regression line over the scatter plot.*

3.1.5 Fitting Distribution

Example 3.11. *Twelve dice were thrown* 2630 *times and each time the number of dice which had* 5 *or,* 6 *on the uppermost face was recorded. The result are as follows:*

No. of dice wih upper 5, 6	0	1	2	3	4	5	6
Frequency	18	115	326	548	611	519	307
No. of dice wih upper 5, 6	7	8	9	10	11	12	-
Frequency	133	40	11	2	0	0	-

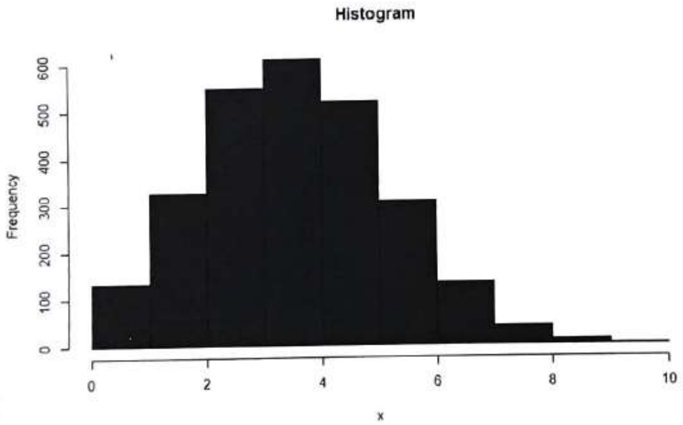
Using R, fit a binomial distribution when p is unknown. Also fit a binomial distribution when $p = 1/3$.

► $\boxed{\textit{R Code}}$:

```
x=c(0,1,2,3,4,5,6,7,8,9,10,11,12)
f=c(18,115,326,548,611,519,307,133,40,11,2,0,0)
n=sum(f)
m=12
dat=rep(x,f)
```

```
hist(dat,col="blue",xlab="x",main="Histogram") # Idea about the Distribution
p=mean(dat)/m # Estimate of p = bar(x)/m
q=1-p
f1=f*0  # Initialize expected frequency
f1[1]=q^m # Intial value is q^m
for (i in 1:11) {
f1[i+1]=((m-i+1)/i)*(p/q)*f1[i]
}
f1[13]=1-sum(f1) # Adjusting total probability
Ef=n*f1 # Expected frequency
Out=data.frame(x,Ef,f)
colnames(Out) =c("|x|", "|Expected Frequency|", "|Observed Frequency|")
format(Out, scientific = F,digit = 3)
```

▶ R Plot: *See the plot Fig. 3.7.*



```
    |x|  |Expected Frequency|  |Observed Frequency|
1    0             18.68824                     18
2    1            114.41520                    115
3    2            321.05568                    326
4    3            546.00052                    548
5    4            626.77191                    611
6    5            511.63879                    519
7    6            304.53990                    307
8    7            133.17766                    133
9    8             42.46639                     40
10   9              9.62935                     11
11  10              1.47385                      2
12  11              0.13672                      0
13  12              0.00581                      0
```

Figure 3.7: Fitting Binomial distribution over data.

■ *Second part is easy, only change the line $p = 1/3$ instead $p = mean(dat)/m$, rest are same.*

Example 3.12. *The following data represents number of telephone calls received in a particular hour*

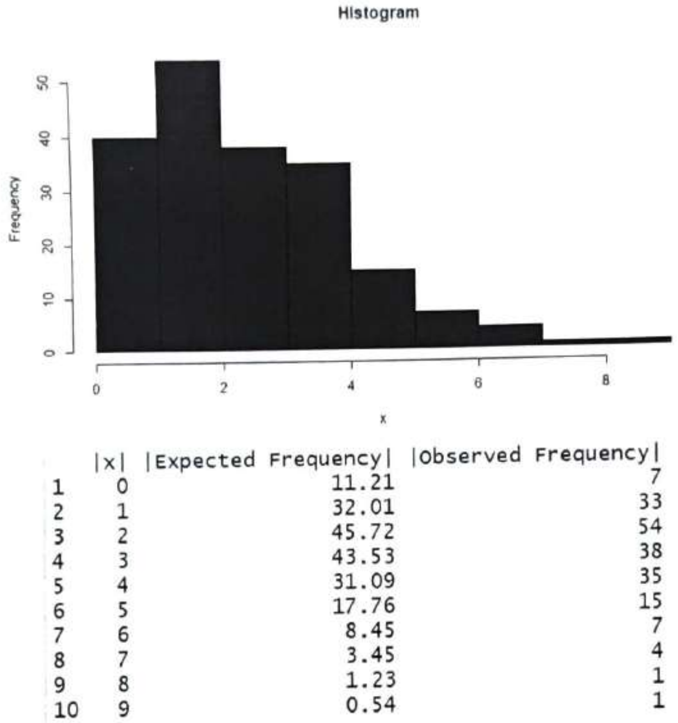
No. of Calls	0	1	2	3	4
Frequency	7	33	54	38	35
No. of Calls	5	6	7	8	9 & more
Frequency	15	7	4	1	1

Using R, fit a poisson distribution when λ is unknown.

► *R Code*:

```
x=c(0,1,2,3,4,5,6,7,8,9)
f=c(7,33,54,38,35,15,7,4,1,1)
n=sum(f)
dat=rep(x,f)
hist(dat,col="blue",xlab="x",main="Histogram") # Idea about the Distribution
lam=mean(dat) # Estimate of lambda = bar(x)
f1=f*0  # Initialize expected frequency
f1[1]=exp(-lam) # Intial value is e^(-lam)
for (i in 1:8) {
f1[i+1]=(lam/i)*f1[i]
}
f1[10]=1-sum(f1) # Adjusting total probability
Ef=n*f1 # Expected frequency
Out=data.frame(x,Ef,f)
colnames(Out) =c("|x|", "|Expected Frequency|", "|Observed Frequency|")
format(Out, scientific = F,digit = 3)
```

► *R Plot*: *See the plot Fig. 3.8.*



```
    |x| |Expected Frequency| |Observed Frequency|
1    0                11.21                    7
2    1                32.01                   33
3    2                45.72                   54
4    3                43.53                   38
5    4                31.09                   35
6    5                17.76                   15
7    6                 8.45                    7
8    7                 3.45                    4
9    8                 1.23                    1
10   9                 0.54                    1
```

Figure 3.8: Fitting Poisson distribution over data.

[Do It Yourself] 3.14. *Given a hypothetical distribution with x : 0 1 2 3 4 5 and f : 210 125 36 21 5 2. Fit a negative binomial distribution on the data.*

Example 3.13. *The following data represents the height and frequency of* 180 *students*

Height	144 - 149	149-154	154 - 159	159 - 164
Frequency	1	3	24	58
Height	164 - 169	169 - 174	174 - 179	179 - 184
Frequency	60	27	2	2

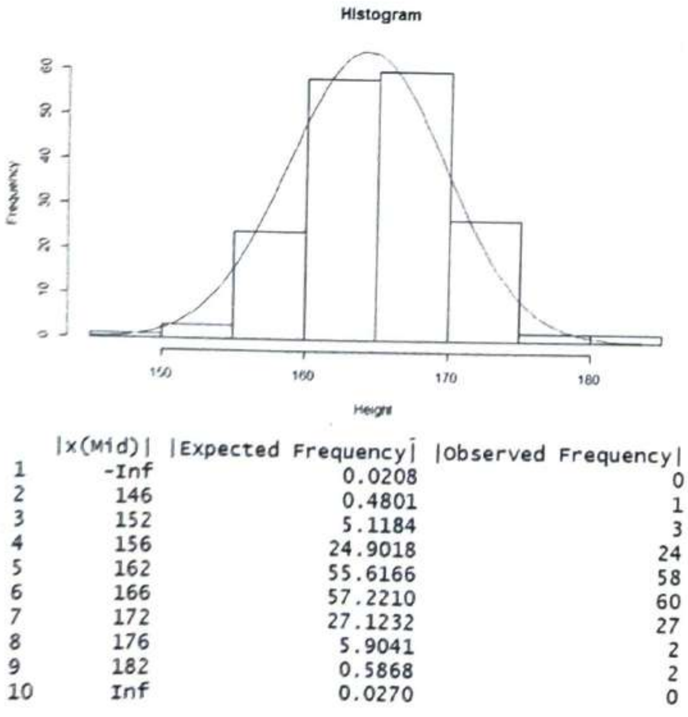
Using R, fit a normal distribution. Also draw the normal curve over the histogram.

► *R Code*:

```
x=seq(146.5,181.5,5) # Midvalues
f=c(1,3,24,58,60,27,2,2)
n=sum(f)
dat=rep(x,f)
```

```
hist(dat,xlab="Height",main="Histogram",ylim=c(0,max(f)+3)) # ylim needed for curve
xbar=mean(dat) # Estimate of mu = mean(x)
s=sd(dat)
curve(n*5*dnorm(x,xbar,s),col="blue",add=T) # Add curve
xi=seq(144,184,5) # Intervalvalues
x1=(xi-xbar)/s # Normalize
y1=dnorm(x1) # Standard normal value
F1=pnorm(x1) # Df values
F2=c(0,F1,1) # Extended DF, including -Inf, Inf
F3= diff(F2)     # Actual Probability values
Ef=n*F3  # Expected frequency
x1=c(-Inf,x,Inf) # Extended x, including -Inf, Inf
f1=c(0,f,0) # Extended f, including -Inf, Inf
Out=data.frame(x1,Ef,f1)
colnames(Out) =c("|x(Mid)|", "|Expected Frequency|", "|Observed Frequency|")
format(Out, scientific = F,digit = 3)
```

► R Plot: *See the plot Fig. 3.9.*



```
   |x(Mid)| |Expected Frequency| |Observed Frequency|
1      -Inf               0.0208                    0
2       146               0.4801                    1
3       152               5.1184                    3
4       156              24.9018                   24
5       162              55.6166                   58
6       166              57.2210                   60
7       172              27.1232                   27
8       176               5.9041                    2
9       182               0.5868                    2
10      Inf               0.0270                    0
```

Figure 3.9: Fitting Normal distribution over data. In first column, instead of x(Mid), it is better to write the class interval e.g. $-\infty - 144,\ 144 - 149,\ \cdots,\ 184 - \infty$.