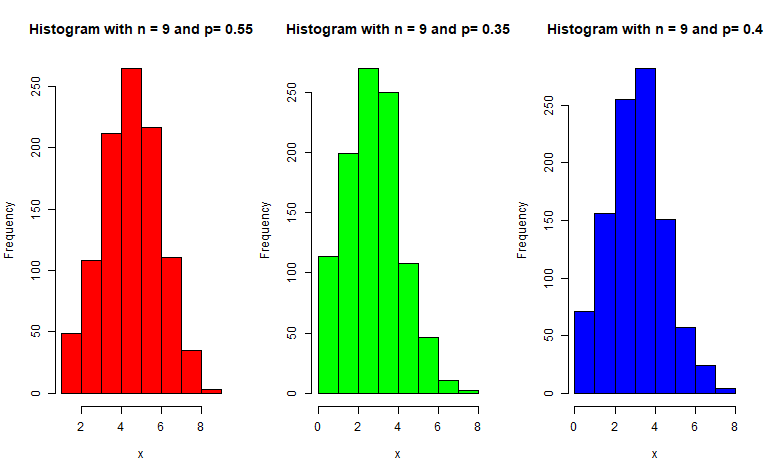
Practical Session 01

*Sampling Distribution and Order Statistics*

|  |  |  |
| --- | --- | --- |
| Name | : | Ashik E Elahi |
| ID | : | 2210724113 |
| Course Title | : | Sampling Distributions and Order Statistics |
| Course Code | : | B.Stat.-209 (Session I) |

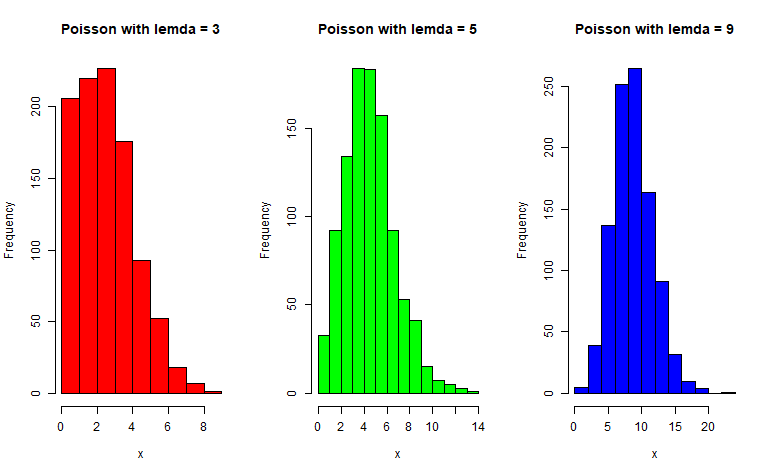
**Binomial Distribution**

set.seed(11)  
par(mfrow=c(1, 3))  
n = c(9, 9, 9) # size  
p = c(0.55, 0.35, 0.4) # Probability  
color = c("red", "green", "blue")  
for(i in 1:3) {  
 x = rbinom(1000, n[i], p[i])  
 hist(x, col=color[i], main = paste("Histogram with n =", n[i], "and p=", p[i]))  
}



**Poisson Distribution**

set.seed(12)  
par(mfrow=c(1,3))  
p <- c(3,5,9)  
color <- c("red", "green", "blue")  
for(i in 1:3) {  
 x = rpois(1000, p[i])  
 hist(x, col=color[i], main=paste("Poisson with lemda =", p[i]))  
}



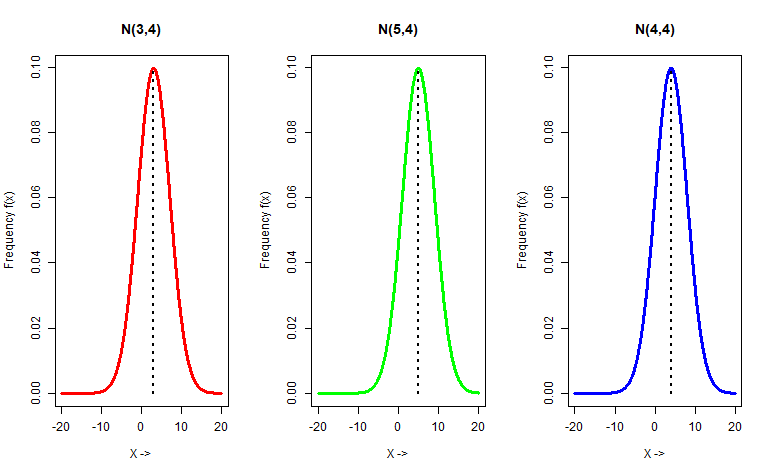
**Problem: 1 - Suppose X follows a normal distribution with mean and variance .**

**(i) Plot the distributions on a graph paper separately with (, and () and compare the graphs.**

**(ii) Plot the distributions on same graph paper with (, and (**

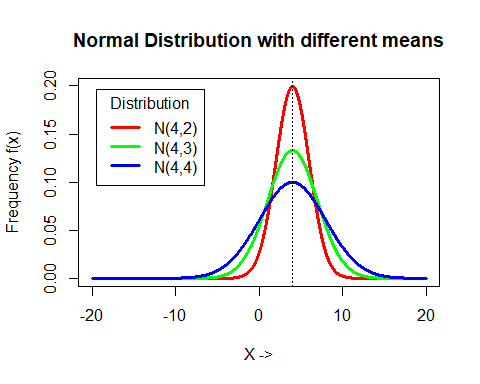
#### **Solution 1 (i):**

set.seed(21)  
par(mfrow=c(1, 3))  
  
mu = c(3, 5, 4)  
sigma = c(4, 4, 4)  
colors = c("red", "green", "blue")  
labels = paste0("N(", mu, ",", sigma, ")")  
x = seq(-20, 20, length.out=1000)  
  
for(i in 1:length(mu)) { # length(mu) = 3  
 fx <- dnorm(x, mean= mu[i], sd = sigma[i])  
 plot(x, fx, type = "l", main = labels[i], col=colors[i], lwd=3,  
 xlab = "X ->", ylab = "Frequency f(x)")  
   
 segments(x0=mu[i], x1=mu[i], y=0, y1 = max(fx), lwd=2, lty=3)  
}



#### **Solution 1 (ii):**

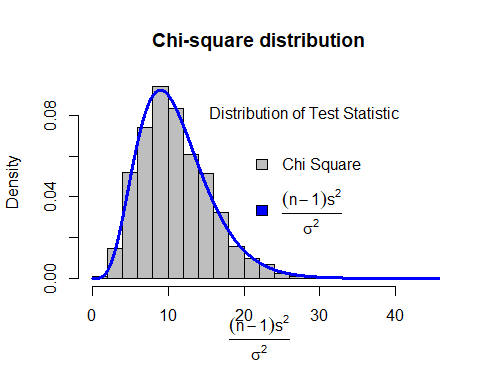
set.seed(21)  
par(mfrow=c(1, 1))  
  
mu = c(4, 4, 4)  
sigma = c(2, 3, 4)  
colors = c("red", "green", "blue")  
labels = paste0("N(", mu, ",", sigma, ")")  
  
x = seq(-20, 20, length.out=1000)  
  
for(i in 1:length(mu)) {  
 fx <- dnorm(x, mean= mu[i], sd = sigma[i])  
 if(i==1) {  
 plot(x, fx, type = "l", main = "Normal Distribution with different means", col=colors[i], lwd=3,  
 xlab = "X ->", ylab = "Frequency f(x)")  
 } else {  
 lines(x, fx, col=colors[i], lwd=3)  
 }  
}  
abline(v=mu[1], lty=3)   
  
legend("topleft", inset = 0.05, title = "Distribution", labels, lty = 1, lwd=3, col = colors)



**Problem 2: Generate 1200 random samples of size n = 12 with and . Suppose . Using sitable graph, justify that follows a chi-square distribution with (n-1) degress of freedom.**

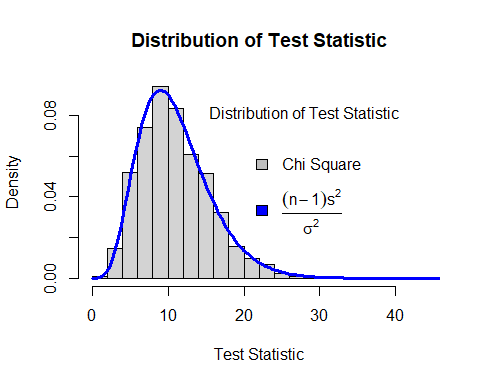
### Method: 01

set.seed(11)  
n = 12  
mu = 4  
sigma = 3  
it = 1200  
  
xvar = array()  
for(i in 1:it) {  
 x = rnorm(n, mu, sigma)  
 xvar[i] = var(x) # list of variances   
}  
  
equation <- expression(frac((n-1)\*s^2,sigma^2))  
  
test\_statistic = ((n-1)\*xvar)/sigma^2   
hist(test\_statistic, breaks = 15, freq = F, col="gray", xlab = equation, main="Chi-square distribution")  
  
x2 = seq(0, 120, 0.01)   
chisq\_pdf = dchisq(x2, df=n-1)  
lines(x2, chisq\_pdf, col="blue", lwd=3)  
  
legend("topright", inset = 0.1, title = "Distribution of Test Statistic", c("Chi Square", equation), fill = c("gray", "blue"), bty = "n")



### Method: 02

set.seed(11)  
n = 12  
mu = 2  
sigma = sqrt(9)  
  
samples <- replicate(1200, rnorm(n, mu, sigma))  
  
test\_statistic <- (n-1)\*(apply(samples, 2, var)/sigma^2)  
  
hist(test\_statistic, breaks = 25, probability = TRUE,  
 main = "Distribution of Test Statistic",   
 xlab="Test Statistic",  
 ylab = "Density")  
  
x <- seq(0, max(test\_statistic)+5, length.out = 100)  
  
lines(x, dchisq(x, df=n-1), col="blue", lwd=3)  
  
legend("topright", inset = 0.1, title = "Distribution of Test Statistic", c("Chi Square", equation), fill = c("gray", "blue"), bty = "n")

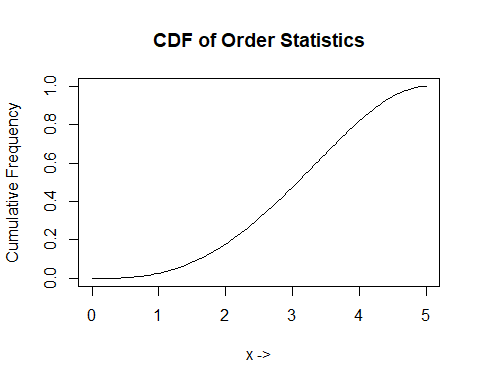


**Problem 3: A random sample of size 4 is drawn from a population that has a uniform distribution on the interval (0, 5). The resulting order statistics are and . Determine the cumulative distribution function (CDF) of the order statistics and plot the CDF of the order statistic. Evaluate the probability**

a = 0  
b = 5  
x = seq(a, b, length.out=1000)  
  
# pdf\_unif = expression(1/(b-a))  
# eval(pdf\_unif)  
  
Fr\_n\_value = function(x, r, n, a, b) {  
 # pdf\_unif = dunif(x, a, b)  
 cdf\_unif = punif(x, a, b)  
 sum\_value = 0  
   
 for(j in r:n) {  
 sum\_value = sum\_value + choose(n, j)\*(cdf\_unif^j)\*(1-cdf\_unif)^(n-j)  
 }  
   
 return(sum\_value)  
}  
  
Fr\_n\_eqn = function(r, n) {  
 sum\_eqn <- array()  
 for(j in r:n) {  
 term <- paste0("(", n, "C", j, ") F(x)^", j, " [1-F(x)]^", (n-j))  
 sum\_eqn[j-r+1] = term   
 }  
 # print(sum\_eqn)  
 return(paste0(sum\_eqn, collapse ="+"))  
}  
  
# Results   
r <- 3  
n <- 4  
a <- 0  
b <- 5  
x <- seq(a, b, length.out=100)  
Fr\_n\_values <- Fr\_n\_value(x, r, n, a, b)  
Fr\_n\_eqn(r, n)

## [1] "(4C3) F(x)^3 [1-F(x)]^1+(4C4) F(x)^4 [1-F(x)]^0"

plot(x, Fr\_n\_values, main="CDF of Order Statistics", type="l", xlab = "x ->", ylab = "Cumulative Frequency")



# P(x\_3:4 > 2)  
x\_threshold <- 2  
p\_x3\_gt\_x <- 1 - Fr\_n\_value(x\_threshold, r, n, a, b)  
p\_x3\_gt\_x

## [1] 0.8208

**Problem 4: A random sample of size 7 is drawn from a population that has a uniform distribution on the interval (0, 5). The resulting order statistics are and . Determine the PDF, mean and variance of order (median) statistic. Also find mean and variance of smallest and largest order statistics**

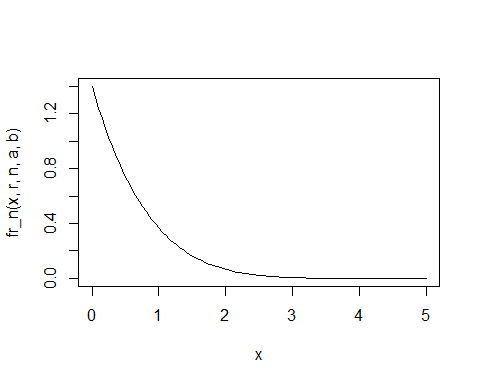
fr\_n = function(x, r, n, a, b) {  
 pdf\_x <- dunif(x, min=a, max=b)  
 cdf\_x <- punif(x, min=a, max=b)  
   
 numerator <- factorial(n) / (factorial(r-1)\*factorial(n-r))  
 prob\_dis <- numerator \* cdf\_x^(r-1) \* (1 - cdf\_x)^(n-r) \* pdf\_x  
   
 return(prob\_dis)  
}  
  
moment\_k <- function(k, r, n, a, b) {  
 integrand <- function(x) x^k \* fr\_n(x, r, n, a, b)  
 moment <- integrate(integrand, a, b)$value  
   
 return(moment)  
}  
a = 0  
b = 5  
n = 7  
# median, r = 4  
r = n %% 2  
x = seq(a, b, length.out = 100)  
  
mean\_median <- moment\_k(k=1, r, n, a, b)  
var\_median <- moment\_k(k=2, r, n, a, b) - mean\_median^2  
  
cat("Mean of median order statistics:", mean\_median, "\n")

## Mean of median order statistics: 0.625

cat("Variance of median order statistics:", var\_median, "\n")

## Variance of median order statistics: 0.3038194

plot(x, fr\_n(x, r, n, a, b), type = "l")



mean\_x1\_n = moment\_k(k=1, r=1, n, a, b)  
var\_x1\_n = moment\_k(k=2, r=1, n, a, b) - mean\_x1\_n^2  
  
mean\_xn\_n = moment\_k(k=1, r=n, n, a, b)  
var\_xn\_n = moment\_k(k=2, r=n, n, a, b) - mean\_xn\_n^2  
  
cat("Smallest order statistics: mean =", mean\_x1\_n, "and variance =", var\_x1\_n, "\n")

## Smallest order statistics: mean = 0.625 and variance = 0.3038194

cat("Largest order statistics: mean =", mean\_xn\_n, "and variance =", var\_xn\_n, "\n")

## Largest order statistics: mean = 4.375 and variance = 0.3038194

**Problem 5: Suppose denote the order statistics based on a random sample of size 7 from the distribution having probability density function (pdf) . Evaluate the mean and variance of the order statistic and plot the CDF of the order statistic.**

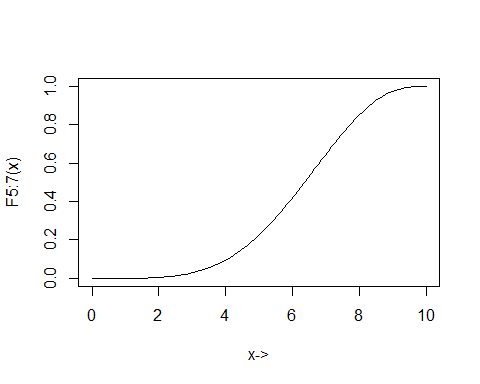
fr\_n = function(x, r, n, a, b) {  
 pdf\_x <- dunif(x, min=a, max=b)  
 cdf\_x <- punif(x, min=a, max=b)  
   
 numerator <- factorial(n) / (factorial(r-1)\*factorial(n-r))  
 prob\_dis <- numerator \* cdf\_x^(r-1) \* (1 - cdf\_x)^(n-r) \* pdf\_x  
   
 return(prob\_dis)  
}  
  
moment\_k <- function(k, r, n, a, b) {  
 integrand <- function(x) x^k \* fr\_n(x, r, n, a, b)  
 moment <- integrate(integrand, a, b)$value  
   
 return(moment)  
}  
a = 0  
b = 10  
r = 5  
n = 7  
x = seq(a, b, length.out = 100)  
  
mean\_x5 <- moment\_k(k=1, r, n, a, b)  
var\_x5 <- moment\_k(k=2, r, n, a, b) - mean\_x5^2  
  
cat("Mean of X\_{(5)}:", mean\_x5, "\n")

## Mean of X\_{(5)}: 6.25

cat("Variance of X\_{(5)}:", var\_x5, "\n")

## Variance of X\_{(5)}: 2.604167

Fr\_n\_values <- array()  
for(i in 1:100) {  
 Fr\_n\_values[i] = integrate(fr\_n, a, x[i], r=r, n=n, a=a, b=b)$value  
}  
  
plot(x, Fr\_n\_values, type = "l", xlab = "x->", ylab=paste0("F", r, ":", n, "(x)"))



**Problem 6: A random sample of size 3 is drawn from a population that has a uniform distribution on the interval (0, 1). The resulting order statistics are and . Determine the mean and variance of smallest and largest order statistic.**

### Method 1:

***kth raw moment of rth order statistics of uniform distribution on the interval [0, 1] is***

rm\_unif\_r\_n <- function(k, r, n) {  
 result <- (factorial(n)\*factorial(k+r-1))/(factorial(r-1)\*factorial(k+3))  
 return(result)  
}  
  
n = 3  
mean\_x1\_3 = rm\_unif\_r\_n(k=1, r=1, n)  
var\_x1\_3 = rm\_unif\_r\_n(k=2, r=1, n) - mean\_x1\_3^2  
  
mean\_x3\_3 = rm\_unif\_r\_n(k=1, r=3, n)  
var\_x3\_3 = rm\_unif\_r\_n(k=2, r=3, n) - mean\_x3\_3^2  
  
cat("Smallest order statistics: mean =", mean\_x1\_3, "and variance =", var\_x1\_3, "\n")

## Smallest order statistics: mean = 0.25 and variance = 0.0375

cat("Largest order statistics: mean =", mean\_x3\_3, "and variance =", var\_x3\_3, "\n")

## Largest order statistics: mean = 0.75 and variance = 0.0375

### Method 2:

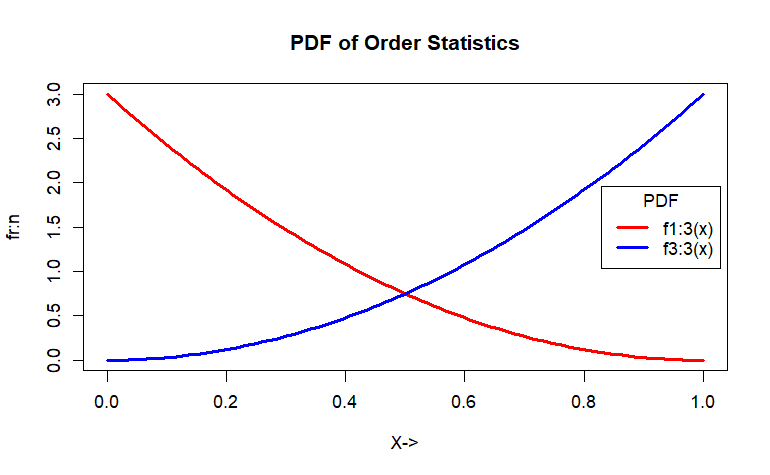
fr\_n = function(x, r, n, a, b) {  
 pdf\_x <- dunif(x, a, b)  
 cdf\_x <- punif(x, a, b)  
   
 numerator <- factorial(n) / (factorial(r-1)\*factorial(n-r))  
 prob\_dis <- numerator \* cdf\_x^(r-1) \* (1-cdf\_x)^(n-r) \* pdf\_x  
   
 return(prob\_dis)  
}  
  
moment\_k <- function(k, r, n, a, b) {  
 integrand <- function(x) x^k \* fr\_n(x, r, n, a, b)  
 moment <- integrate(integrand, a, b)$value  
 return(moment)  
}  
  
a = 0  
b = 1  
n = 3  
  
colors = c("red", "blue")  
  
x <- seq(a, b, length.out = 100)  
mean\_x1\_3 = moment\_k(k=1, r=1, n, a, b)   
var\_x1\_3 = moment\_k(k=2, r=1, n, a, b) - mean\_x1\_3^2   
mean\_x3\_3 = moment\_k(k=1, r=3, n, a, b)   
var\_x3\_3 = moment\_k(k=2, r=3, n, a, b) - mean\_x3\_3^2  
  
cat("Smallest order statistics: mean =", mean\_x1\_3, "and variance =", var\_x1\_3, "\n")

## Smallest order statistics: mean = 0.25 and variance = 0.0375

cat("Largest order statistics: mean =", mean\_x3\_3, "and variance =", var\_x3\_3, "\n")

## Largest order statistics: mean = 0.75 and variance = 0.0375

plot(x, fr\_n(x, r=1, n, a, b), type = "l", col=colors[1], lwd=3, main = "PDF of Order Statistics", xlab = "X->", ylab = "fr:n")  
lines(x, fr\_n(x, r=3, n, a, b), col = colors[2], lwd=3)  
  
legend("right", inset = 0.01, title = "PDF", c("f1:3(x)", "f3:3(x)"), lty=1, lwd=3, col=colors)



**Problem 7: A random sample of size 3 is drawn from a population that has a uniform distribution on the interval (0, 1). The resulting order statistics are . Determine the mean and variance of sample range.**

fr\_n = function(x, r, n, a, b) {  
 pdf\_x = dunif(x, min=a, max=b)  
 cdf\_x = punif(x, min=a, max=b)  
   
 numerator <- factorial(n) / (factorial(r-1)\*factorial(n-r))  
 prob\_dist <- numerator \* cdf\_x^(r-1) \* (1 - cdf\_x)^(n-r)\*pdf\_x  
   
 return(prob\_dist)  
}  
  
a = 0  
b = 1  
n = 3  
x = seq(0, 1, length.out=100)  
  
moment\_k <- function(k, r, n, a, b) {  
 integrand <- function(x) x^k \* fr\_n(x, r, n, a, b)  
 moment <- integrate(integrand, a, b)$value  
 return(moment)  
}  
  
exp\_xy <- function(n, a, b) {  
 integrand <- function(x) {  
 f\_y <- function(y) n\*(n-1)\*x\*y\*(y-x)  
 # integration from x to b (dependent range)  
 integrate(Vectorize(f\_y), x, b)$value   
 }  
   
 # integration from a to b  
 result <- integrate(Vectorize(integrand), a, b)$value  
 return(result)  
}  
  
mean\_x1\_n = moment\_k(k=1, r=1, n, a, b)  
var\_x1\_n = moment\_k(k=2, r=1, n, a, b) - mean\_x1\_n^2  
  
mean\_xn\_n = moment\_k(k=1, r=n, n, a, b)  
var\_xn\_n = moment\_k(k=2, r=n, n, a, b) - mean\_xn\_n^2  
  
cat("Smallest order statistics: mean =", mean\_x1\_n, "and variance =", var\_x1\_n, "\n")

## Smallest order statistics: mean = 0.25 and variance = 0.0375

cat("Largest order statistics: mean =", mean\_xn\_n, "and variance =", var\_xn\_n, "\n")

## Largest order statistics: mean = 0.75 and variance = 0.0375

cov\_order = exp\_xy(n, a, b) - (mean\_x1\_n \* mean\_xn\_n)  
  
cat("Covariance of smallest and largest order statistics =", cov\_order, "\n")

## Covariance of smallest and largest order statistics = 0.0125

mean\_sample\_range = (mean\_xn\_n - mean\_x1\_n)  
var\_sample\_range = var\_x1\_n + var\_xn\_n - 2\*cov\_order  
  
cat("Mean of sample range:", mean\_sample\_range, "\n")

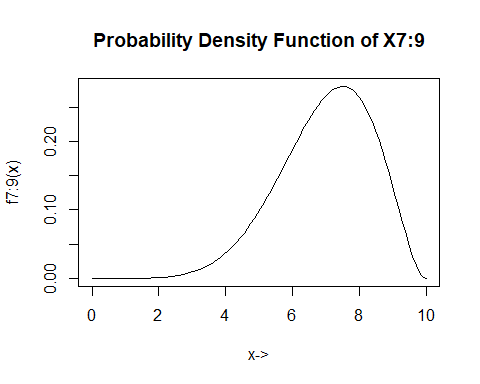
## Mean of sample range: 0.5

cat("Variance of sample range:", var\_sample\_range, "\n")

## Variance of sample range: 0.05

**Problem 8: The random sample of size 9 is drawn from a population that has a uniform distribution with the interval (0, 10). Evaluate the mean and variance of 7th order statistic. Also find the value of**

fr\_n = function(x, r, n, a, b) {  
 pdf\_x <- dunif(x, a, b)  
 cdf\_x <- punif(x, a, b)  
   
 numerator <- factorial(n) / (factorial(r-1)\*factorial(n-r))  
 prob\_dis <- numerator \* cdf\_x^(r-1) \* (1-cdf\_x)^(n-r) \* pdf\_x  
   
 return(prob\_dis)  
}  
  
a <- 0  
b <- 10  
n <- 9  
r <- 7  
  
x <- seq(a, b, length.out=100)  
  
fr\_n\_values <- sapply(x, fr\_n, r=r, n=n, a=a, b=b)  
  
plot(x, fr\_n\_values, type = "l", xlab = "x->",   
 ylab = paste0("f", r, ":", n, "(x)"),   
 main = paste0("Probability Density Function of X", r, ":", n))



moment\_k <- function(k, r, n, a, b) {  
 integrand <- function(x) x^k \* fr\_n(x, r, n, a, b)  
 moment <- integrate(integrand, a, b)$value  
 moment  
}  
  
mean\_x7 <- moment\_k(1, r, n, a, b)  
var\_x7 <- moment\_k(2, r, n, a, b) - mean\_x7^2  
  
cat("Mean of X\_{(7)}:", mean\_x7, "\n")

## Mean of X\_{(7)}: 7

cat("Variance of X\_{(7)}:", var\_x7, "\n")

## Variance of X\_{(7)}: 1.909091

# int 2 to 10  
p\_x7\_gt\_2 <- integrate(fr\_n, 2, b, r=r, n=n, a=a, b=b)$value  
cat("P(X\_{(7)} > 2):", p\_x7\_gt\_2, "\n")

## P(X\_{(7)} > 2): 0.9996861