Question Solving

Ashik E Elahi

2024-01-21

Sampling Distribution and Order Statistics

R Programming

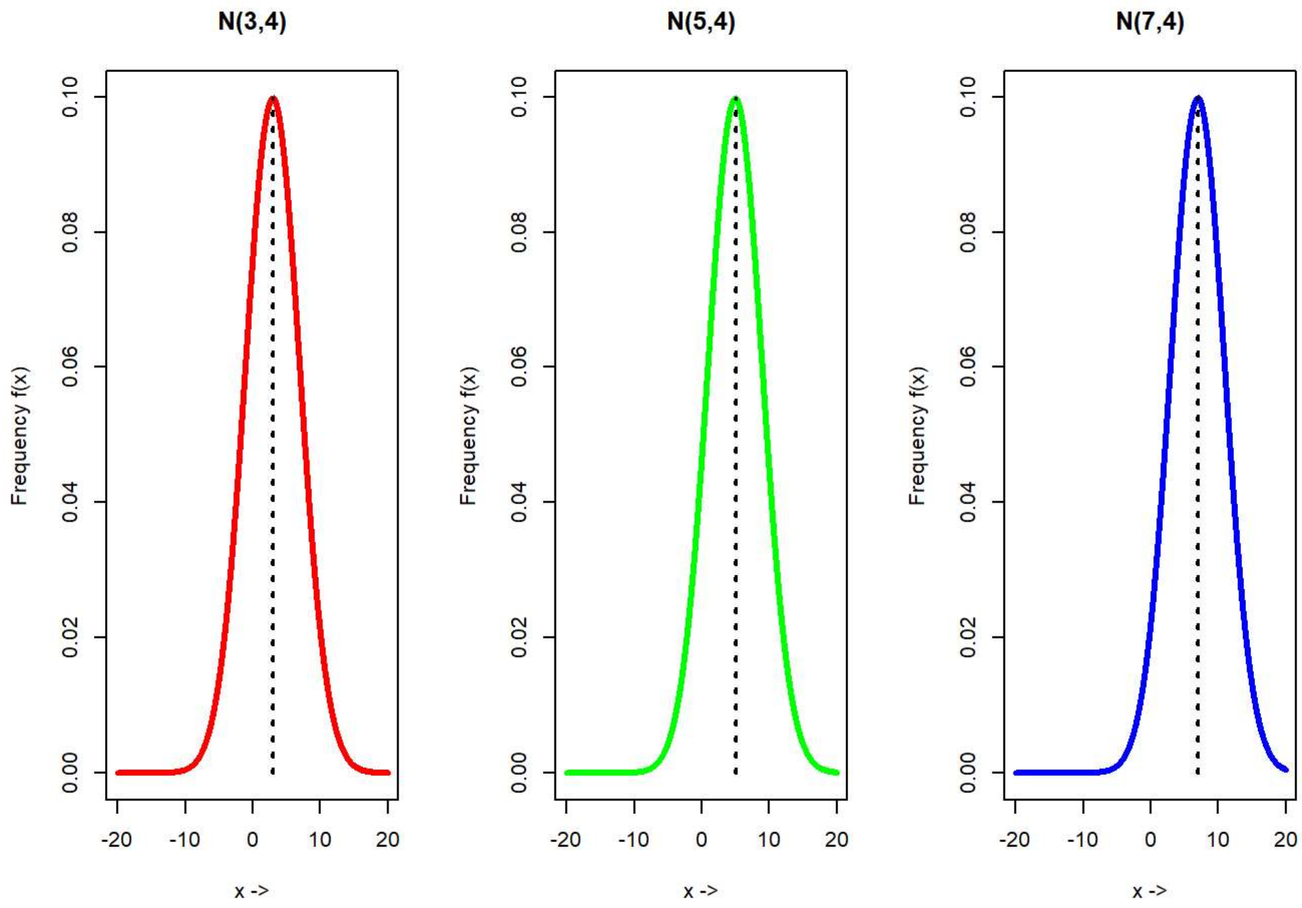
Problem: 1 - Suppose X follows a normal distribution with mean μ and variance σ^2.

(i) Plot the distributions on a graph paper separately with ($\mu = 3, \sigma^2 = 16), (\mu = 5, \sigma^2 = 16)$, and ($\mu = 4, \sigma^2 = 16$) and compare the graphs.

(ii) Plot the distributions on same graph paper with ($\mu = 4, \sigma^2 = 4), (\mu = 4, \sigma^2 = 9)$, and ($\mu = 4, \sigma^2 = 16$)

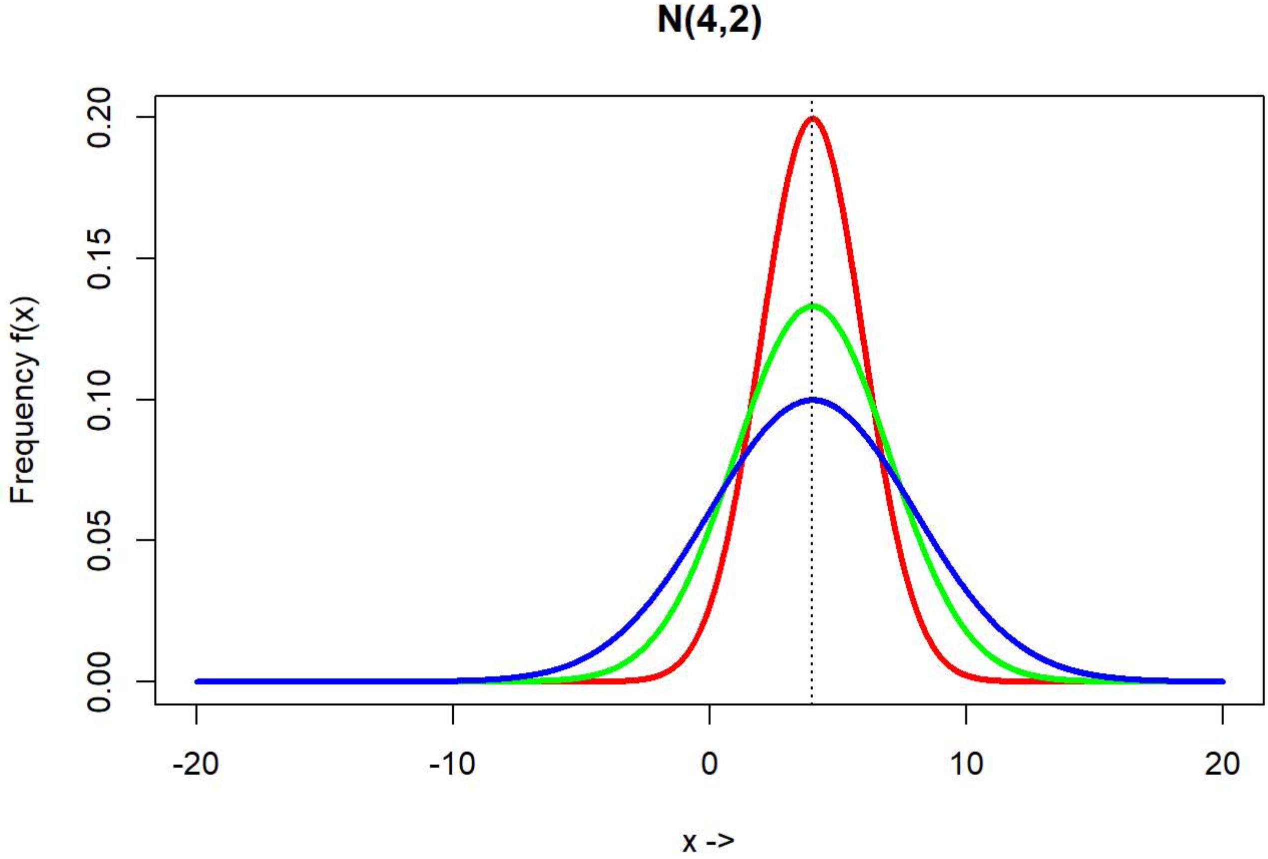
Solution 1 (i):

```
set.seed(2111)
par(mfrow = c(1, 3))
mu <- c(3, 5, 7)
sigma.2 <- c(16, 16, 16)
sigma <- sqrt(sigma.2)
colors <- c("red", "green", "blue")
labels <- paste0("N(", mu, ",", sigma, ")")
x <- seq(-20, 20, 0.01)

for(i in 1:length(mu)) {
  fx <- dnorm(x, mean = mu[i], sd = sigma[i])
  plot(x, fx, type="l", col = colors[i], lwd=3,
      main = labels[i], xlab = "x ->",
      ylab = "Frequency f(x)")

  segments(x0 = mu[i], y0 = 0, x1 = mu[i], y1 = max(fx), lwd = 2, lty=3)
}
```



Solution 1 (ii):

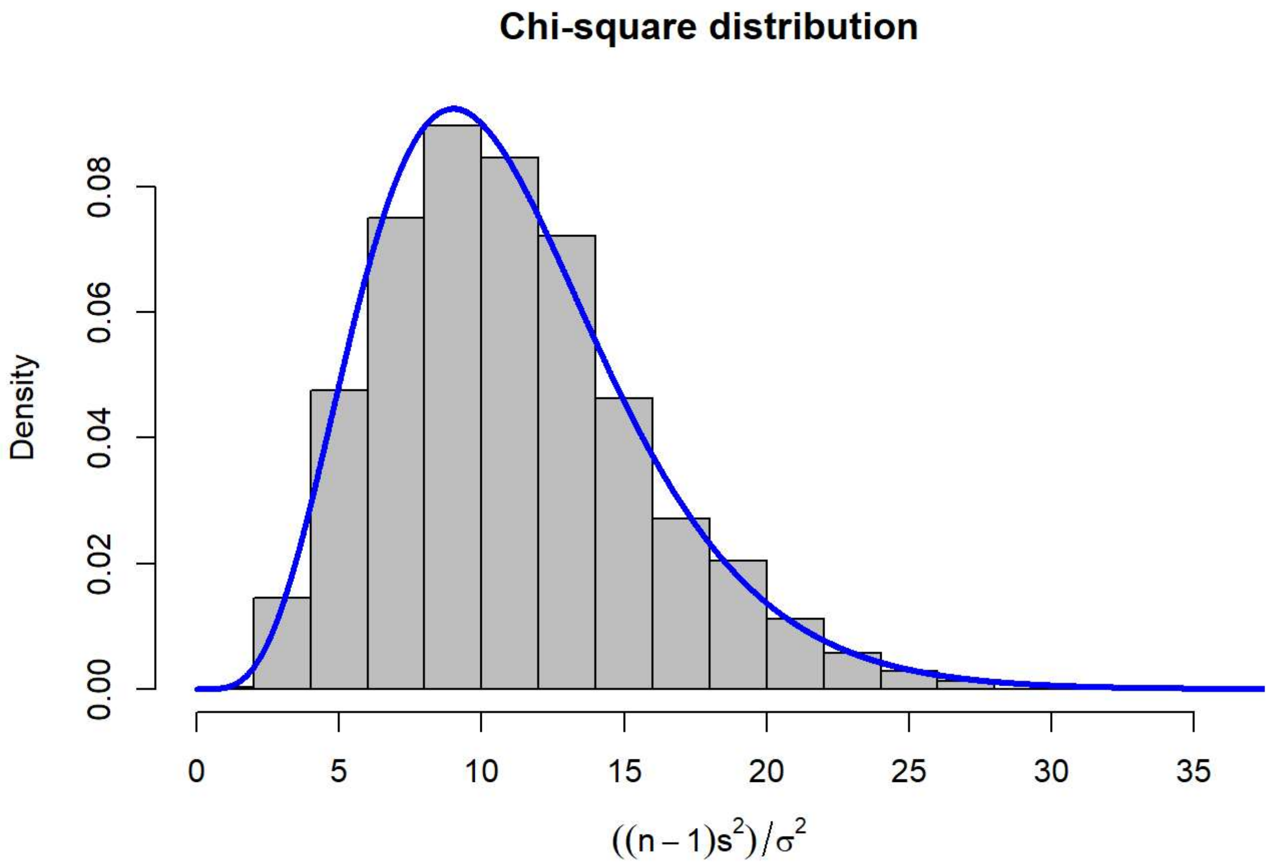
```
set.seed(2112)
mu <- c(4, 4, 4)
sigma.2 <- c(4, 9, 16)
sigma <- sqrt(sigma.2)
colors <- c("red", "green", "blue")
labels <- paste0("N(", mu, ",", sigma, ")")
x <- seq(-20, 20, 0.01)

for(i in 1:length(mu)) {
  fx <- dnorm(x, mean = mu[i], sd = sigma[i])
  if(i == 1) {
    plot(x, fx, type="l", col = colors[i], lwd=3,
        main = labels[i], xlab = "x ->",
        ylab = "Frequency f(x)")
  } else {
    lines(x, fx, lwd=3, col=colors[i])
  }
}
abline(v=mu[1], lty=3)
```



Problem 2: Generate 1200 random samples of size n = 12 with $\mu = 4$ and $\sigma^2 = 9$. Suppose $s^2 = \frac{1}{n-1}\sum_i^n (x_i - \bar{x})^2$. Using sitable graph, justify that $\frac{(n-1)s^2}{sigma^2}$ follows a chi-square

distribution with (n-1) degress of freedom.

```
par(mfrow=c(1, 1))
n = 12
mu = 4
sigma.2 = 9
sigma = sqrt(sigma.2)
it = 1200

xvar = array()
for(i in 1:it) {
  x = rnorm(n, mu, sigma)
  xvar[i] = var(x)
}
# xvar -> list of variances

x = ((n-1)*xvar)/sigma.2
hist(x, breaks = 15, freq = F, col = "gray", xlab = expression(((n-1)*s^2)/sigma^2), main="Ch
i-square distribution")

x.2 = seq(0,120, 0.01)
chisq.pdf = dchisq(x.2, df=n-1)
lines(x.2, chisq.pdf, col="blue", lwd=3)
```



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 $X_{1:4}$, $X_{2:4}$, $X_{3:4}$ and $X_{4:4}$.

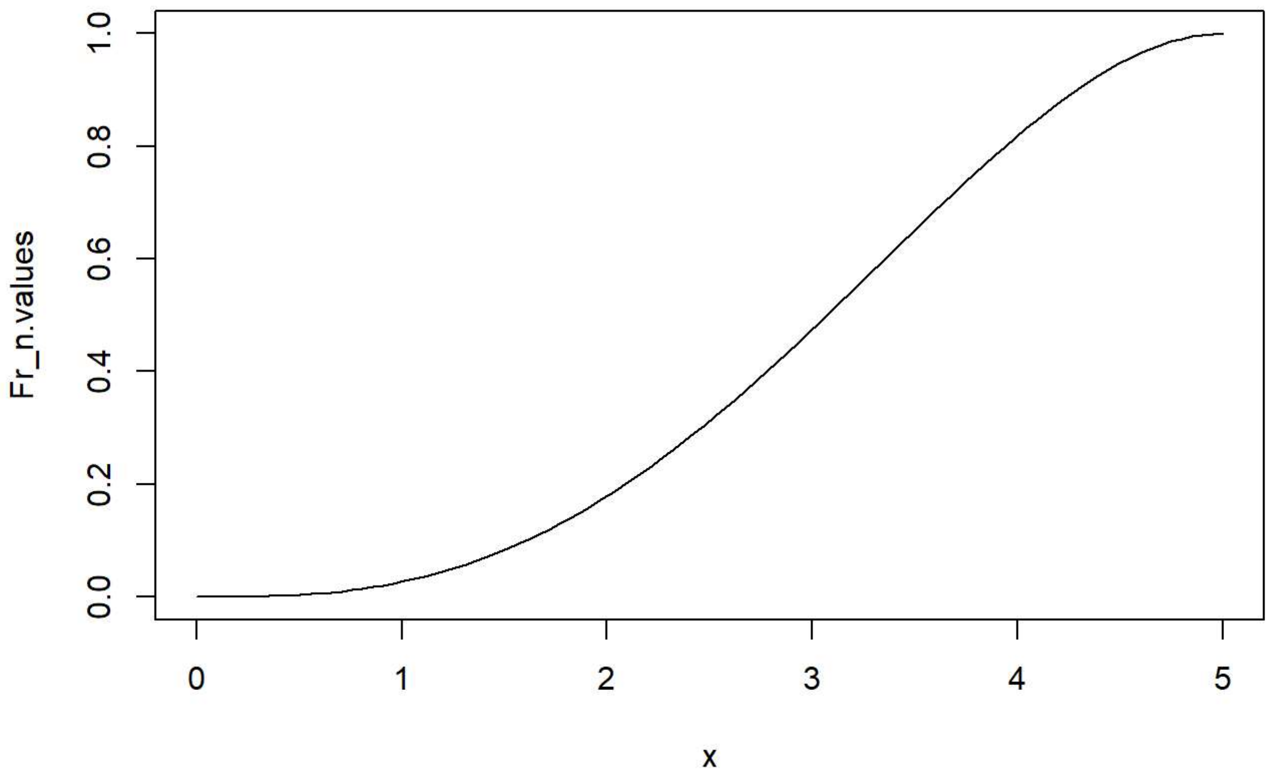
Determine the cumulative distribution function (CDF) of the 3^{rd} order statistics $(X_{3:4})$ and plot the CDF of the 3^{rd} order statistic. Evaluate the probability $P[X_{3:4} >= 2]$

```
# Uniform Distribution
# pdf_uniform = dunif(x, min=0, max=5)
# cdf_uniform = punif(x, min=0, max=5)

# PDF of uniform (User Defined Function)
pdf.uniform <- function(x, min, max) {
  ifelse(x < min | x > max, 0, 1/(max - min))
}


# CDF of uniform (User Defined Function)
cdf.uniform <- function(x, min, max) {
  ifelse(x < min, 0,
         ifelse(x > max, 1,
                x/(max-min)))
}

# CDF of rth order statistics Fr:n
cdf.order.uniform.eqn <- function(x, r, n) {
  sum_eqn = ""

  for(j in r:n) {
    term = paste0(n,"C", j, "[F(x)]^", j, "[1-F(x)]^(", n, "-", j, ")")
    if(j < n)
      sum_eqn = paste0(sum_eqn, term, "+")
    else
      sum_eqn = paste0(sum_eqn, term)
  }

  return(sum_eqn)
}

# CDF value
cdf.order.uniform.value <- function(x, r, n, min, max) {
  cdf_x = cdf.uniform(x, min, max)
  sum_value = 0

  for(j in r:n) {
    sum_value = sum_value + choose(n, j)*(cdf_x^j)*(1-cdf_x)^(n-j)
  }
  return(sum_value)
}

# Results

r <- 3
n <- 4
x <- seq(0, 5, length.out=100)

Fr_n.eqn <- cdf.order.uniform.eqn(x, r, n)
```

```
Fr_n.values <- cdf.order.uniform.value(x, r, n, min=0, max=5)

cat("The distribution function F", r, ":", n, "(x) is given by :\n", Fr_n.eqn)
```

```
## The distribution function F 3 : 4 (x) is given by :
##  4C3[F(x)]^3[1-F(x)]^(4-3)+4C4[F(x)]^4[1-F(x)]^(4-4)
```

```
plot(x, Fr_n.values, type="l")
```




```
# P(x_3:4 > 2)
x_threshold <- 2
p_x3_gt_x <- 1 - cdf.order.uniform.value(x_threshold, 3, 4, min=0, max=5)
p_x3_gt_x
```

```
## [1] 0.8208
```

Problem 4: Suppose $X_{1:7} < X_{2:7} < X_{3:7} < X_{4:7} < X_{5:7} < X_{6:7} < X_{7:7}$ denote the order statistics based on a random sample of size 7 from the distribution having probability density function (pdf)

$f(x) = \frac{1}{10}, 0 <= x <= 10$. Evaluate the mean and variance of the 5^{th} order statistic $(X_{5:7})$ and plot the CDF of the 5^{th} order statistic.

```
pdf_uniform = dunif(x, min=0, max=10)
cdf_uniform = punif(x, min=0, max=10)
```

Problem 5 **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 $X_{1:3}, X_{2:3}$ and $X_{3:3}$.** Determine the mean and variance of smallest and largest order statistic.

```
# kth moment of rth order statistic  = (3!*(k+r-1)!) / ((r-1)! * (k+3)!)
rm_uniform_order <- function(k, r) {
  (factorial(3)*factorial(k+r-1)) / (factorial(r-1)*factorial(k+3))
}

mean_X_1.3 = rm_uniform_order(k = 1, r = 1)
 var_X_1.3 = rm_uniform_order(k = 2, r = 1) - mean_X_1.3^2
mean_X_3.3 = rm_uniform_order(k = 1, r = 3)
 var_X_3.3 = rm_uniform_order(k = 2, r = 3) - mean_X_3.3^2

mean_X_1.3
```

```
## [1] 0.25
```

```
var_X_1.3
```

```
## [1] 0.0375
```

```
mean_X_3.3
```

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

```
var_X_3.3
```

```
## [1] 0.0375
```

Method 2

```
# kth moment of rth roder statistic when sample size is n
# PI_i=1^k ((r+i) / (n+i))
rm_uniform_order_n <- function(n, r, k) {
  mult = 1
  for(i in 1:k) {
    term = (r+i-1) / (n+i)
    mult = mult * term
  }
  return(mult)
}
mean_X_1.3 = rm_uniform_order_n(n = 3, r = 1, k = 1)
 var_X_1.3 = rm_uniform_order_n(n = 3, r = 1, k = 2) - mean_X_1.3^2
mean_X_3.3 = rm_uniform_order_n(n = 3, r = 3, k = 1)
 var_X_3.3 = rm_uniform_order_n(n = 3, r = 3, k = 2) - mean_X_3.3^2

mean_X_1.3
```

```
## [1] 0.25
```

```
var_X_1.3
```

```
## [1] 0.0375
```

```
mean_X_3.3
```

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

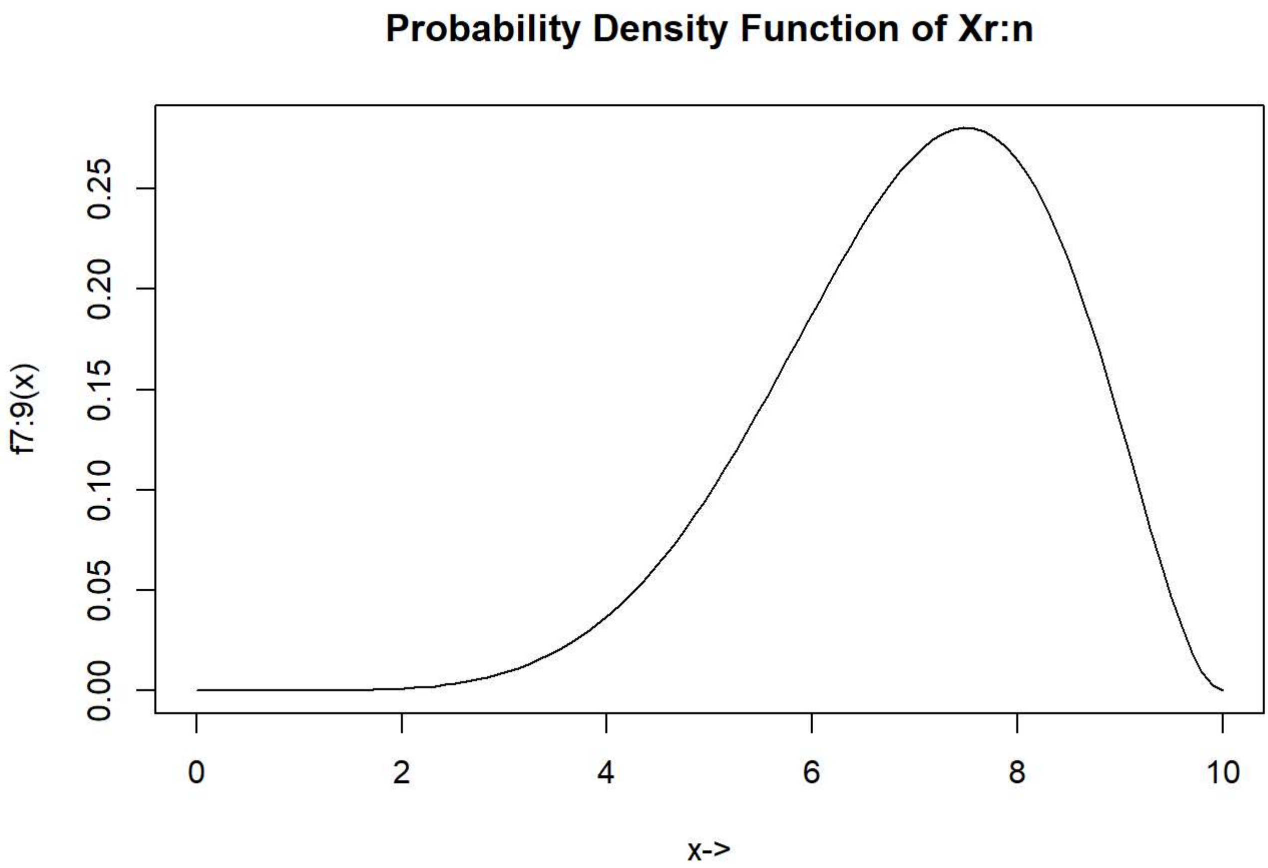
```
var_X_3.3
```

```
## [1] 0.0375
```

Problem 6: The random sample $x_1, x_2, \ldots, x_9$ 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.

```
# cdf
cdf_uniform <- function (x, a, b) {
    ifelse(x < a, 0, ifelse(x > b, 1, (x - a) / (b - a)))
}

# fr:n
fr_n <- function (x, r, n, a, b) {
    cdf_x <- cdf_uniform(x, a, b)
    pdf_x <- 1 / (b - a)

    numerator <- factorial(n) / (factorial(r-1)*factorial(n-r))
    prob_dist <- numerator * cdf_x^(r-1) * (1 - cdf_x)^(n-r)*pdf_x
    prob_dist
}


a <- 0
b <- 10
n <- 9
r <- 7

x_values <- seq(a, b, length.out=100)

fr_n_values <- sapply(x_values, fr_n, r=r, n=n, a=a, b=b)

plot(x_values, fr_n_values, type="l", ylim=c(0, max(fr_n_values)), xlab="x->", ylab=paste0
("f", r, ":", n, "(x)"), main=paste0("Probability Density Function of Xr:n"))
```




```
# 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
```

```
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
```