Worksheet3

STAT414

2024-09-28

```
# #1 - Describe some experiment or observational study, specifying two characteristics (preferably
# one categorical and the other continuous) of interest. Define the population, and how you
# will take physical samples. What are the random variables of interest in your study? What
# kind of shape and characteristics do you think the probability distributions of these random
# variables will have? Why?

cat("
    * Observational study: age distribution of male video game players across different game
    genres. The continuous variable is the age of the players, the categorical
    variable is the genre/type of video game, i.e shooter, moba, puzzle.
    * The population will be male players that interact with video games, thus
    a global scale of male gamers at varying ages.
    * The samples will be taken through either social media or general gaming forums,
    where randomly selected players will be asked to fill a survey of their age
    and preference of video game. To remove potential sampling bias, forums or platforms
    that only host a single or large majority of a demographic, i.e all from a single country,
    will not be the only sample, the sample will include different countries and demographic
    groups.
    * The random variables of the study are the categorical variable and continuous variable;
    the genre/type of game, and the age of the player, respectively.
    * I believe that for different types of genres of video games, the age distribution
    will be skewed towards different age groups compared to other genres. For example,
    the previously mentioned genres: the distribution of age for male players in shooters
    will have a distribution skewed towards the right, where age is measured from 0 and
    ascending; predicting that younger players favor shooter games more than older players.
    MOBA games could have a normal distribution of age for its male players, as their appeals
    are multi-layered and attract a large variety of age groups for different aspects.
    Puzzle games might have a distribution that is left skewed, as older players may find
    mind-training games more stimulating than younger players.
    ")
```

```
##
##     * Observational study: age distribution of male video game players across different game
##     genres. The continuous variable is the age of the players, the categorical
##     variable is the genre/type of video game, i.e shooter, moba, puzzle.
##     * The population will be male players that interact with video games, thus
##     a global scale of male gamers at varying ages.
##     * The samples will be taken through either social media or general gaming forums,
##     where randomly selected players will be asked to fill a survey of their age
##     and preference of video game. To remove potential sampling bias, forums or platforms
##     that only host a single or large majority of a demographic, i.e all from a single country,
##     will not be the only sample, the sample will include different countries and demographic
```

```
##     groups.
##     * The random variables of the study are the categorical variable and continuous variable;
##     the genre/type of game, and the age of the player, respectively.
##     * I believe that for different types of genres of video games, the age distribution
##     will be skewed towards different age groups compared to other genres. For example,
##     the previously mentioned genres: the distribution of age for male players in shooters
##     will have a distribution skewed towards the right, where age is measured from 0 and
##     ascending; predicting that younger players favor shooter games more than older players.
##     MOBA games could have a normal distribution of age for its male players, as their appeals
##     are multi-layered and attract a large variety of age groups for different aspects.
##     Puzzle games might have a distribution that is left skewed, as older players may find
##     mind-training games more stimulating than younger players.
##
```

```r
# #2 - Suppose that a population is adequately described by a Gamma distribution with shape=3
# and scale=2.
#install.packages("moments")
library("moments")

# a. Generate three samples, of size n = 10, 100 and 1000, respectively. For each sample,
# create a density histogram. Overlay the pdf of Gamma(shape=3,scale=2) on the created
# histogram. Comment on how increasing the sample size improves the fit between the
# sample histogram and its theoretical expectation.
shape <- 3
scale <- 2
sample_10 <- rgamma(10, shape=shape, scale=scale)
sample_100 <- rgamma(100, shape=shape, scale=scale)
sample_1000 <- rgamma(1000, shape=shape, scale=scale)
mean(sample_10)
```

```
## [1] 5.708756
```

```r
mean(sample_100)
```

```
## [1] 6.498493
```

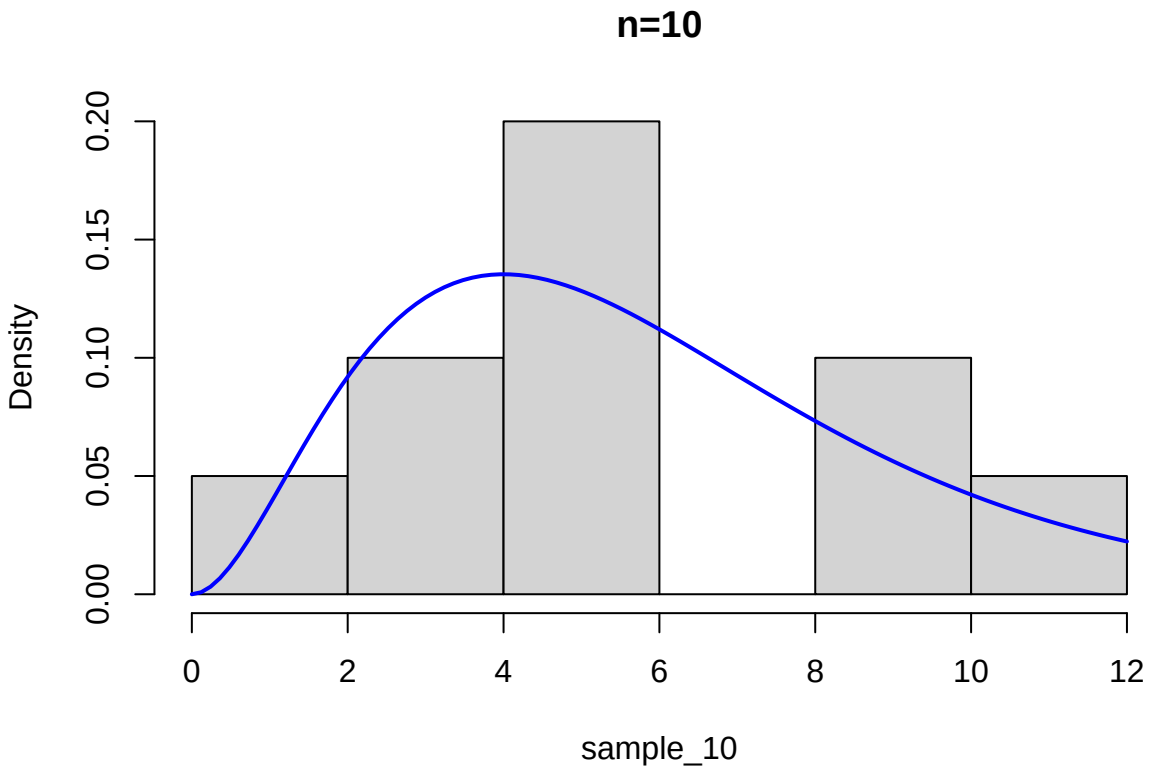
```r
mean(sample_1000)
```

```
## [1] 5.972694
```

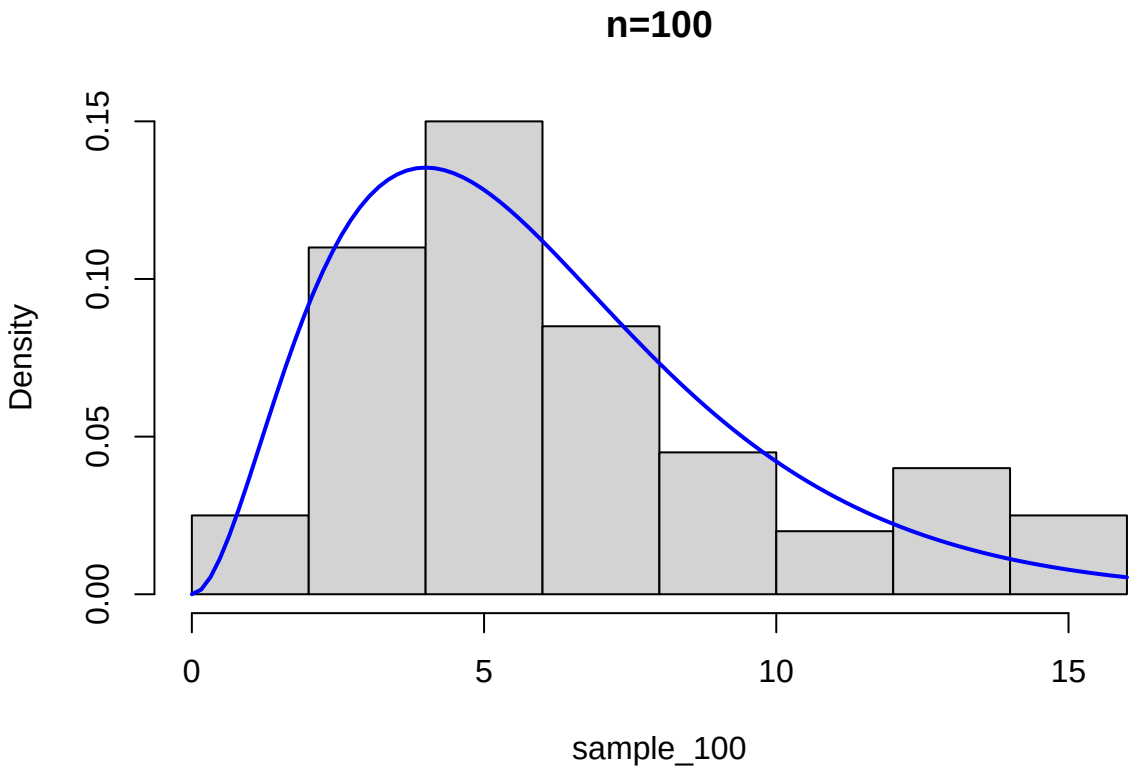
```r
mean(rgamma(100000, shape = shape, scale = scale))
```

```
## [1] 5.999246
```

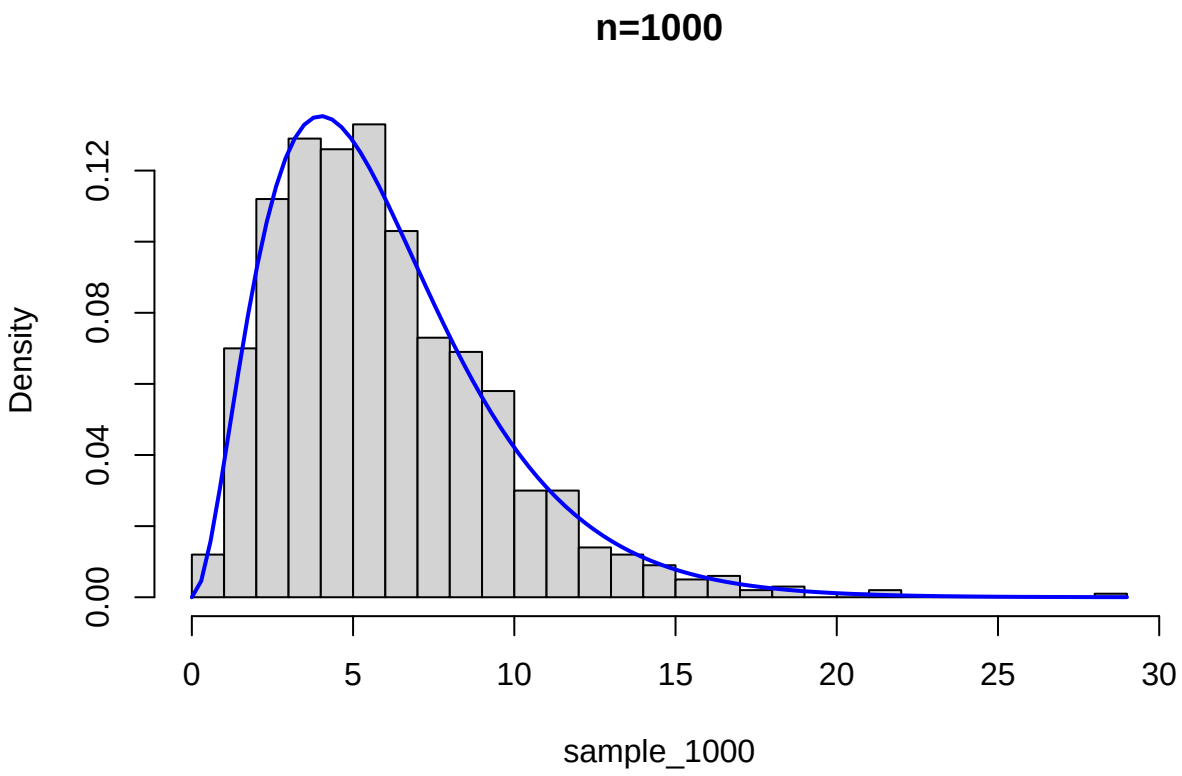
```r
hist(sample_10,probability=TRUE,main="n=10",breaks=5)
curve(dgamma(x, shape=shape, scale=scale), lwd=2, col="blue", add=TRUE)
```




```
hist(sample_100,probability=TRUE,main="n=100",breaks=10)
curve(dgamma(x, shape=shape, scale=scale), lwd=2, col="blue", add=TRUE)
```




```
hist(sample_1000,probability=TRUE,main="n=1000",breaks=20)
curve(dgamma(x, shape=shape, scale=scale), lwd=2, col="blue", add=TRUE)
```




```
cat("As n increases from 10 to 100, to 1000, the fit of the histogram gradually begins
    to more accurately match the theoretical curve of the pdf. This is due to
    the Central Limit Theorem and Law of Large Numbers.")
```

```
## As n increases from 10 to 100, to 1000, the fit of the histogram gradually begins
##     to more accurately match the theoretical curve of the pdf. This is due to
##     the Central Limit Theorem and Law of Large Numbers.
```

```
# b. Compute the mean, Variance, Skew and CV of the sample with 1000 observations
# generated above and compare them to their true values given in equations (4.47) of
# Millard and Neerchal.
```

$$\text{Mean} = \mu = k \cdot \theta$$

$$\text{Variance} = k \cdot \theta^2$$

$$\text{Skewness} = \frac{2}{\sqrt{k}}$$

$$\text{CV} = \frac{1}{\sqrt{k}}$$

$$k = \text{Shape}$$

$$\theta = \text{Scale}$$

```
mean_sample_1000 <- mean(sample_1000)
var_sample_1000 <- var(sample_1000)
skewness_sample_1000 <- skewness(sample_1000)
cv_sample_1000 <- sd(sample_1000) / mean(sample_1000)

true_mean_1000 <- shape * scale
true_var_1000 <- shape * scale**2
true_skewness_1000 <- 2/sqrt(shape)
true_cv_1000 <- 1/sqrt(shape)

cat("Sample mean:", mean_sample_1000, " vs. True:", true_mean_1000)
```

```
## Sample mean: 5.972694  vs. True: 6
```

```
cat("Sample var:", var_sample_1000, " vs. True:", true_var_1000)
```

```
## Sample var: 11.98002  vs. True: 12
```

```
cat("Sample skew:", skewness_sample_1000, " vs. True:", true_skewness_1000)
```

```
## Sample skew: 1.284179  vs. True: 1.154701
```

```
cat("Sample CV:", cv_sample_1000, " vs. True:", true_cv_1000)
```

```
## Sample CV: 0.5795068  vs. True: 0.5773503
```

```
cat("At n=1000, the sample statistics are extremely close to the population parameters,
    differing by a small margin of about less than .4 at most.")
```

```
## At n=1000, the sample statistics are extremely close to the population parameters,
##     differing by a small margin of about less than .4 at most.
```

```
# c. Generate 100000 random numbers from this distribution and create a density histogram. Overlay the
# mean, Variance, Skew and CV of this sample and compare them to their true values
# given in equations (4.47) of Millard and Neerchal.

sample_100000 <- rgamma(100000,shape=shape, scale=scale)

mean_sample_100000 <- mean(sample_100000)
var_sample_100000 <- var(sample_100000)
skewness_sample_100000 <- skewness(sample_100000)
cv_sample_100000 <- sd(sample_100000) / mean(sample_100000)

true_mean_100000 <- shape * scale
true_var_100000 <- shape * scale**2
```

```r
true_skewness_100000 <- 2/sqrt(shape)
true_cv_100000 <- 1/sqrt(shape)

cat("Sample mean:", mean_sample_100000, " vs. True:", true_mean_100000)
```

```
## Sample mean: 6.008528  vs. True: 6
```

```r
cat("Sample var:", var_sample_100000, " vs. True:", true_var_100000)
```

```
## Sample var: 12.0112  vs. True: 12
```

```r
cat("Sample skew:", skewness_sample_100000, " vs. True:", true_skewness_100000)
```

```
## Sample skew: 1.157838  vs. True: 1.154701
```

```r
cat("Sample CV:", cv_sample_100000, " vs. True:", true_cv_100000)
```

```
## Sample CV: 0.5767998  vs. True: 0.5773503
```

```r
cat("At n=100000, the sample statistics are even closer to the population parameters
    compared to n=1000. The statistics differ by at most .03. As n grows larger, the
    closer the sample statistics will match their true population parameters.")
```

```
## At n=100000, the sample statistics are even closer to the population parameters
##     compared to n=1000. The statistics differ by at most .03. As n grows larger, the
##     closer the sample statistics will match their true population parameters.
```

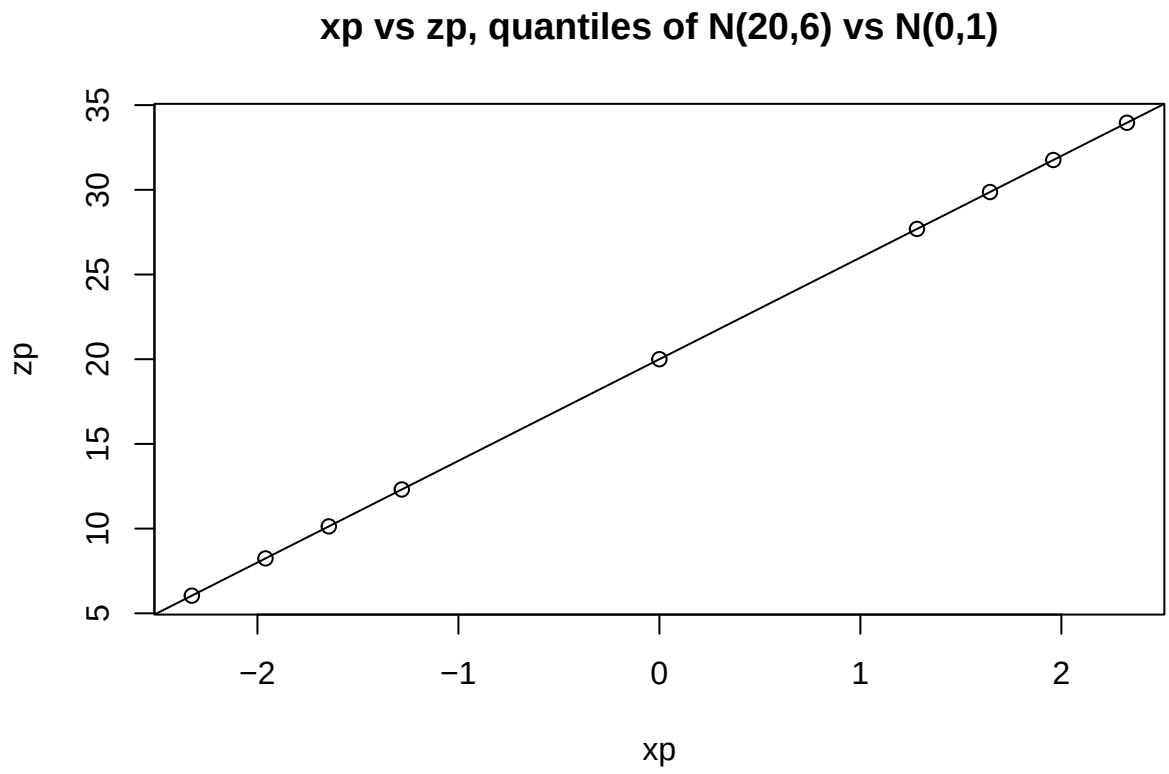
```r
# #3 - Suppose the population is well-approximated by a Normal distribution N(μ = 20, sigma = 6).

# (a) Use the appropriate R function and compute the quantiles (xp) corresponding to p =
# 1%,2.5%, 5%, 10% 50% 90% 95%, 97.5% and 99%. Compute the corresponding quantiles
# (zp) for the standard normal distribution.
mu <- 20
sig <- 6
p <- c(.01, .025, .05, .1, .5, .9, .95, .975, .99)

#generate quantiles using qnorm
xp <- qnorm(p, mean=mu, sd=sig)
zp <- qnorm(p, mean=0, sd=1) #standard normal N(0,1)

xp
```

```
## [1]  6.041913  8.240216 10.130878 12.310691 20.000000 27.689309 29.869122
## [8] 31.759784 33.958087
```

```r
zp
```

```
## [1] -2.326348 -1.959964 -1.644854 -1.281552  0.000000  1.281552  1.644854
## [8]  1.959964  2.326348
```

```r
# (b) Plot xp vs zp. What do you see? What is the slope and intercept of this line?
plot(zp, xp, main="xp vs zp, quantiles of N(20,6) vs N(0,1)",
     ylab="zp", xlab="xp")
abline(lm(xp~zp)) #add line of linear regression to plot
```




```r
summary(lm(xp~zp))
```

```
## Warning in summary.lm(lm(xp ~ zp)): essentially perfect fit: summary may be
## unreliable

##
## Call:
## lm(formula = xp ~ zp)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -1.023e-14 -1.180e-15  2.065e-15  2.922e-15  3.884e-15
##
## Coefficients:
##              Estimate Std. Error   t value Pr(>|t|)
## (Intercept) 2.000e+01  1.716e-15 1.165e+16   <2e-16 ***
## zp          6.000e+00  9.872e-16 6.078e+15   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 5.149e-15 on 7 degrees of freedom
## Multiple R-squared:      1,  Adjusted R-squared:      1
## F-statistic: 3.694e+31 on 1 and 7 DF,  p-value: < 2.2e-16
```

```
cat("The slope of xp~zp (xp is the is dependent, zp is the independent) is 6, and
    the intercept is 20. Notice that the slope is equal to mu and the intercept
    is equal to sigma. The line is completely linear with no outliers. The datapoints
    are also symmetrical from (0,20) because of the quantiles probabilities selected.
    It is a fact that all normal distributions regardless of their mean or SD, are
    linear transformations of the standard normal distribution.")
```

```
## The slope of xp~zp (xp is the is dependent, zp is the independent) is 6, and
##     the intercept is 20. Notice that the slope is equal to mu and the intercept
##     is equal to sigma. The line is completely linear with no outliers. The datapoints
##     are also symmetrical from (0,20) because of the quantiles probabilities selected.
##     It is a fact that all normal distributions regardless of their mean or SD, are
##     linear transformations of the standard normal distribution.
```

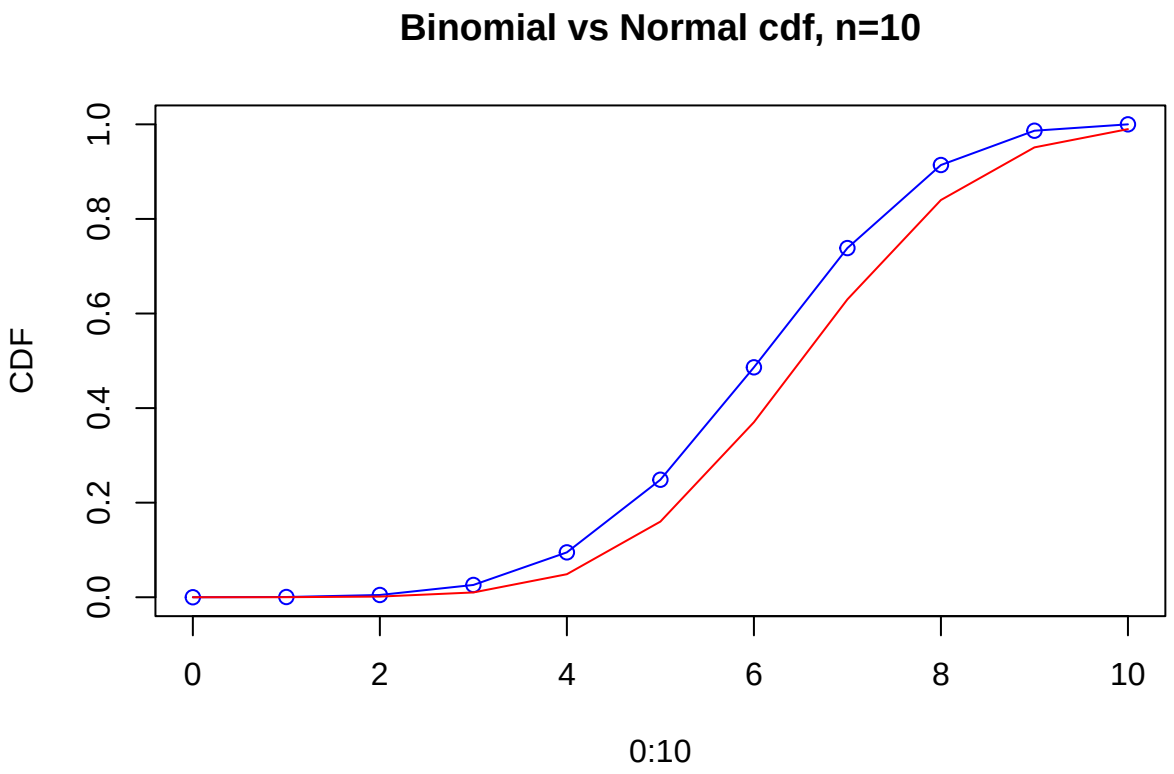
```
# #4 - Consider a random variable X ~ Binomial(n, p). It is known that, for large n, the Binomial
# probabilities are well-approximated by a Normal random variable with mean μX = E(X) and
# variance sigma^2X = Var(X). Let us find out how large n should be for this approximation to be
# acceptably accurate.

# (a) For p = 0.65, and n = 10, compute the mean (μX) and standard deviations (sigmaX.) Plot
# the CDF of this random variable and the plot the CDF of the approximating Normal
# random variable.
p <- .65
n <- 10
#mean = np, sd = sqrt((1-p)np)
bmean <- n*p
bsd <- sqrt((1-p)*n*p)

bmean
```

```
## [1] 6.5
```

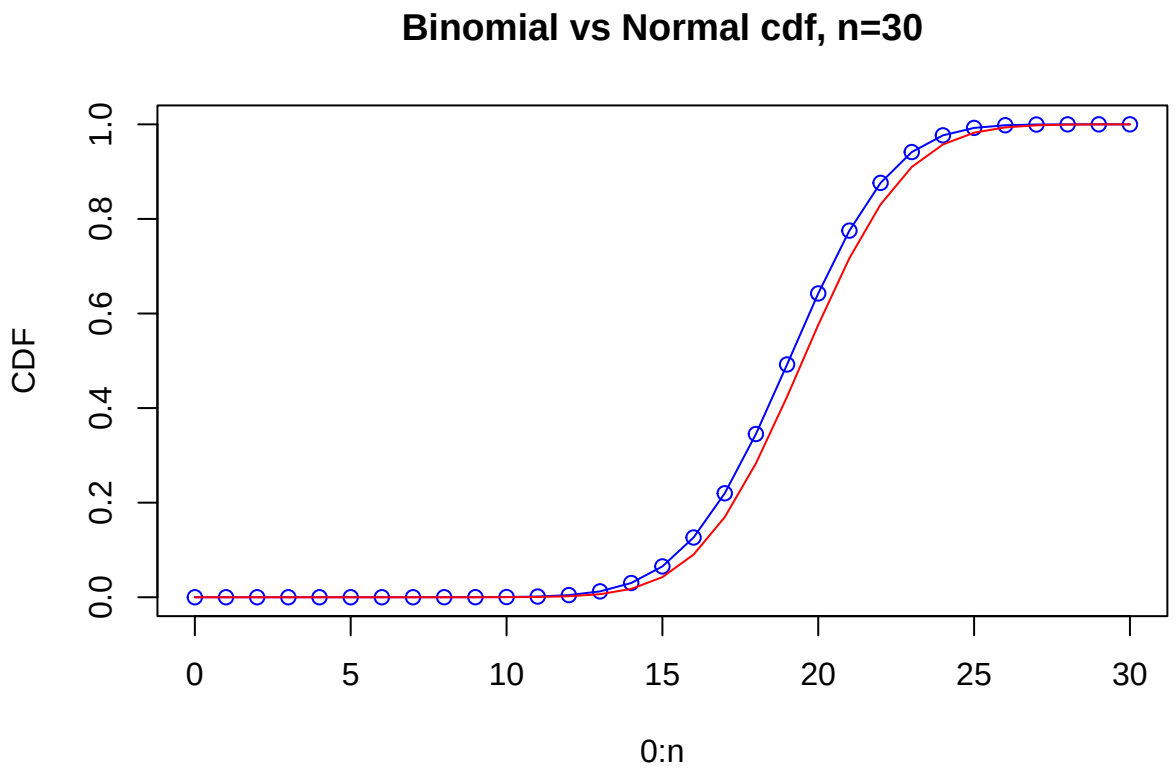
```
bsd
```

```
## [1] 1.50831
```

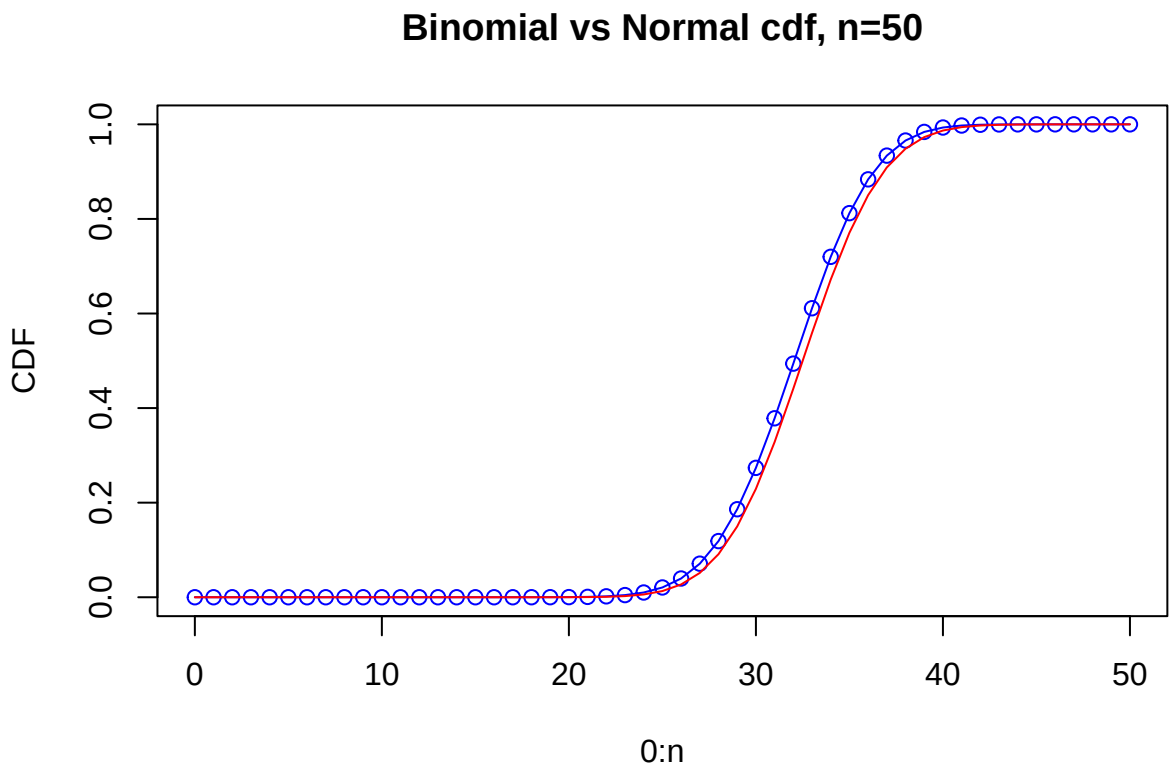
```
#plot the cdf of this binomial rv, and a normal rv
bcdf <- pbinom(0:n, size=n, prob=p)
ncdf <- pnorm(0:n, bmean, bsd)
plot(0:10, bcdf, type="o", main="Binomial vs Normal cdf, n=10", ylab="CDF", col="blue")
lines(0:10, ncdf,col="red")
```



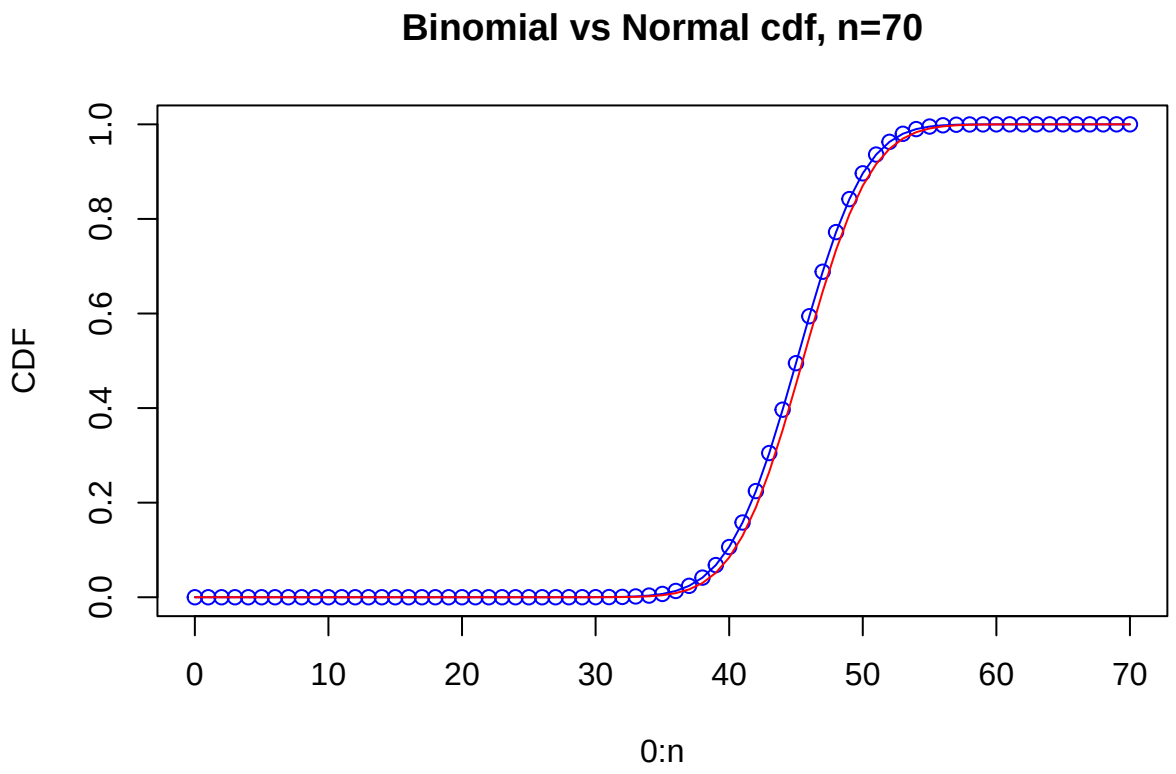

```r
# (b) Repeat the above with larger values of n and comment on the quality of approximation.
n <- 30
bmean <- n*p
bsd <- sqrt((1-p)*n*p)

bcdf <- pbinom(0:n, size=n, prob=p)
ncdf <- pnorm(0:n, bmean, bsd)
plot(0:n, bcdf, type="o", main="Binomial vs Normal cdf, n=30", ylab="CDF", col="blue")
lines(0:n, ncdf,col="red")
```



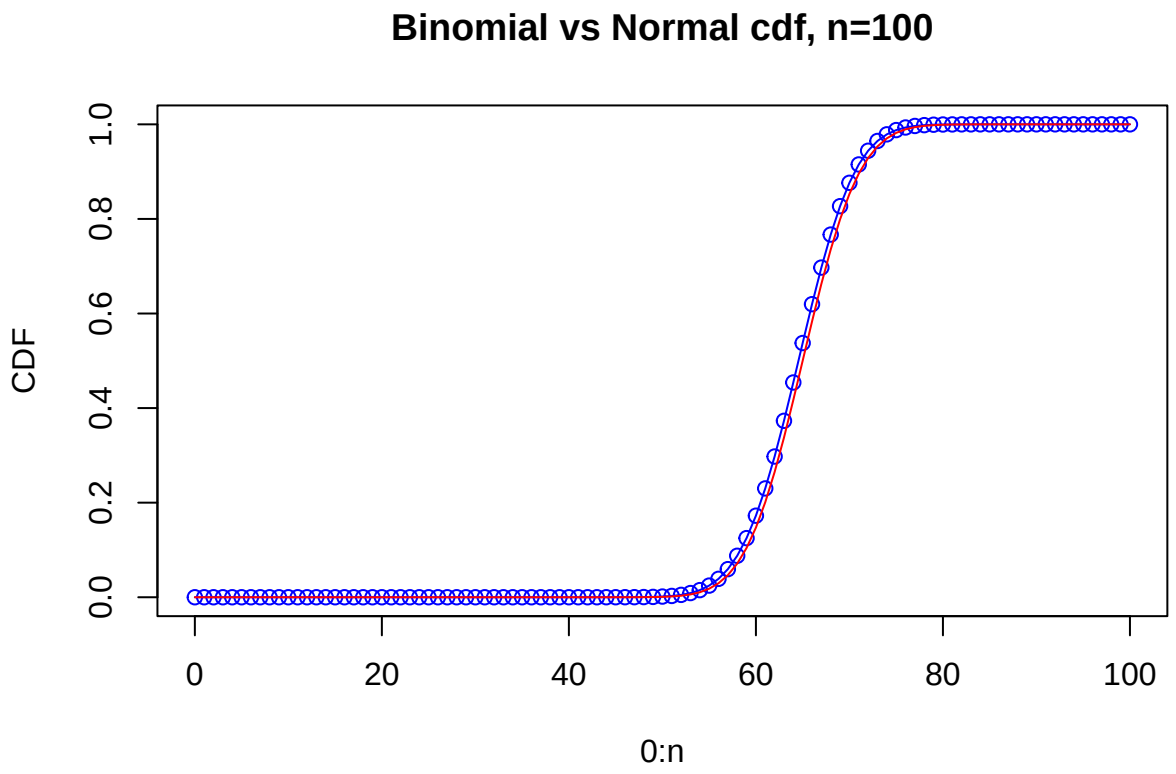

```
n <- 50
bmean <- n*p
bsd <- sqrt((1-p)*n*p)

bcdf <- pbinom(0:n, size=n, prob=p)
ncdf <- pnorm(0:n, bmean, bsd)
plot(0:n, bcdf, type="o", main="Binomial vs Normal cdf, n=50", ylab="CDF", col="blue")
lines(0:n, ncdf,col="red")
```



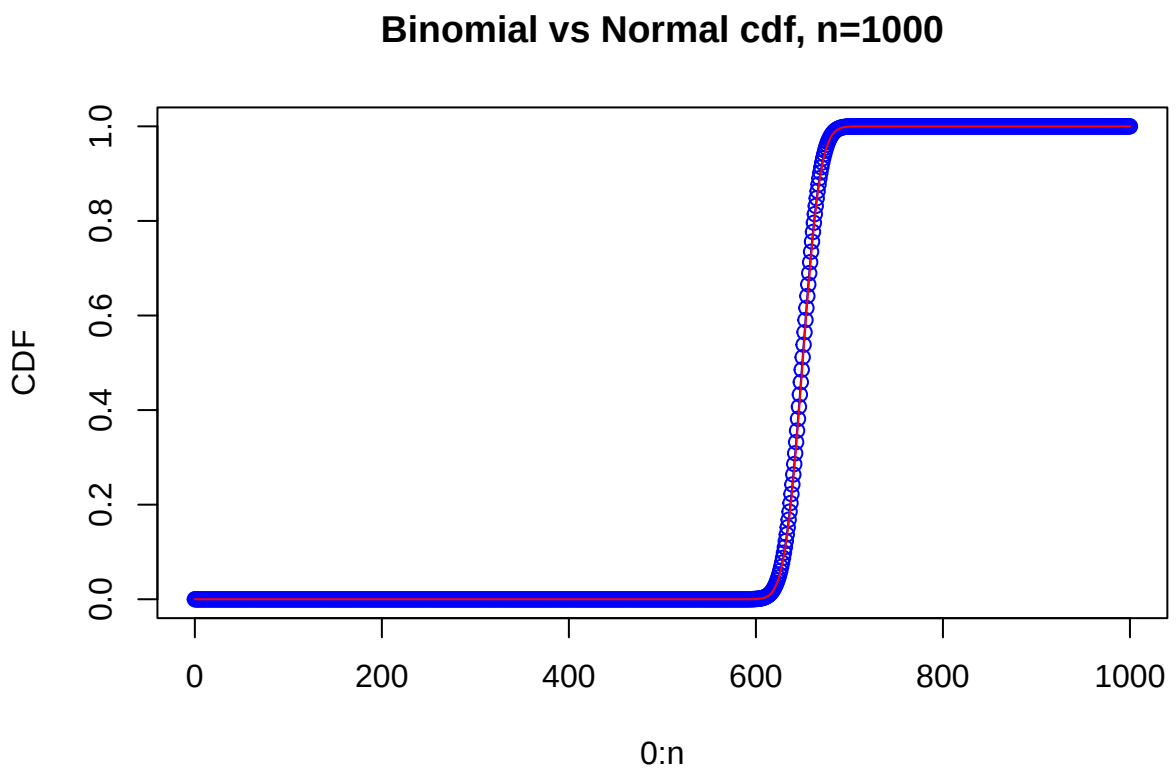

```
n <- 70
bmean <- n*p
bsd <- sqrt((1-p)*n*p)

bcdf <- pbinom(0:n, size=n, prob=p)
ncdf <- pnorm(0:n, bmean, bsd)
plot(0:n, bcdf, type="o", main="Binomial vs Normal cdf, n=70", ylab="CDF", col="blue")
lines(0:n, ncdf,col="red")
```



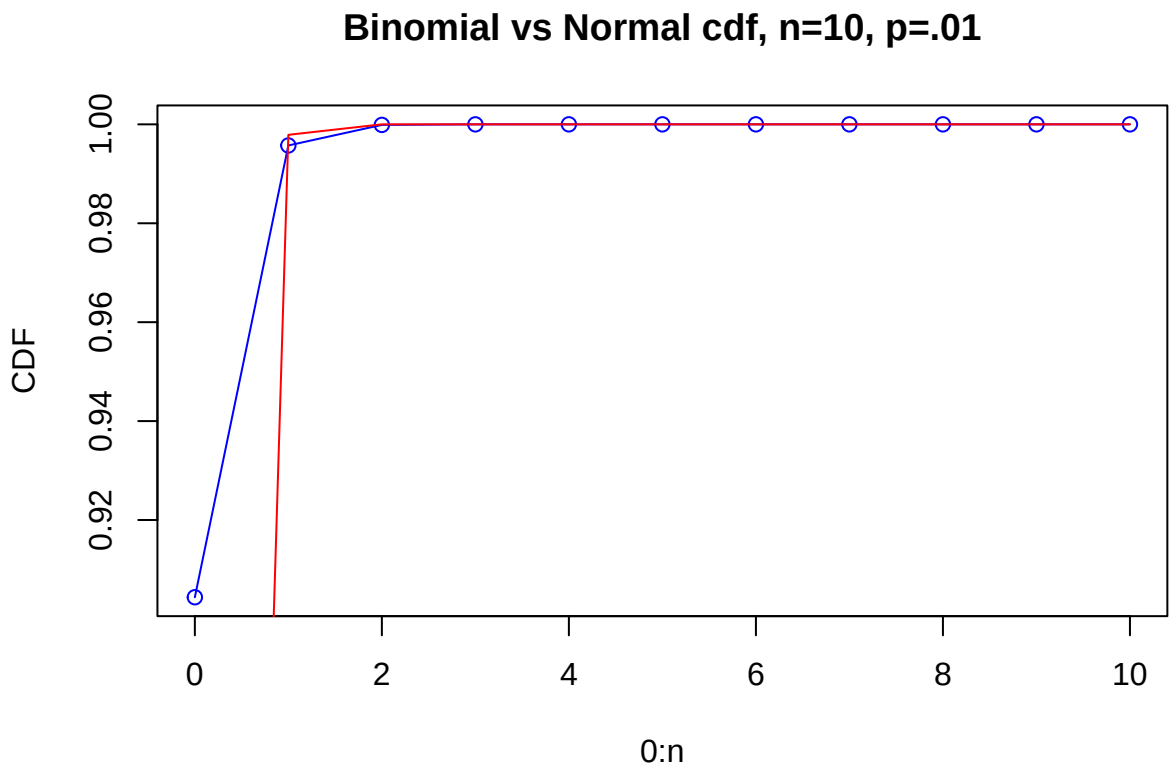

```
n <- 100
bmean <- n*p
bsd <- sqrt((1-p)*n*p)

bcdf <- pbinom(0:n, size=n, prob=p)
ncdf <- pnorm(0:n, bmean, bsd)
plot(0:n, bcdf, type="o", main="Binomial vs Normal cdf, n=100", ylab="CDF", col="blue")
lines(0:n, ncdf,col="red")
```



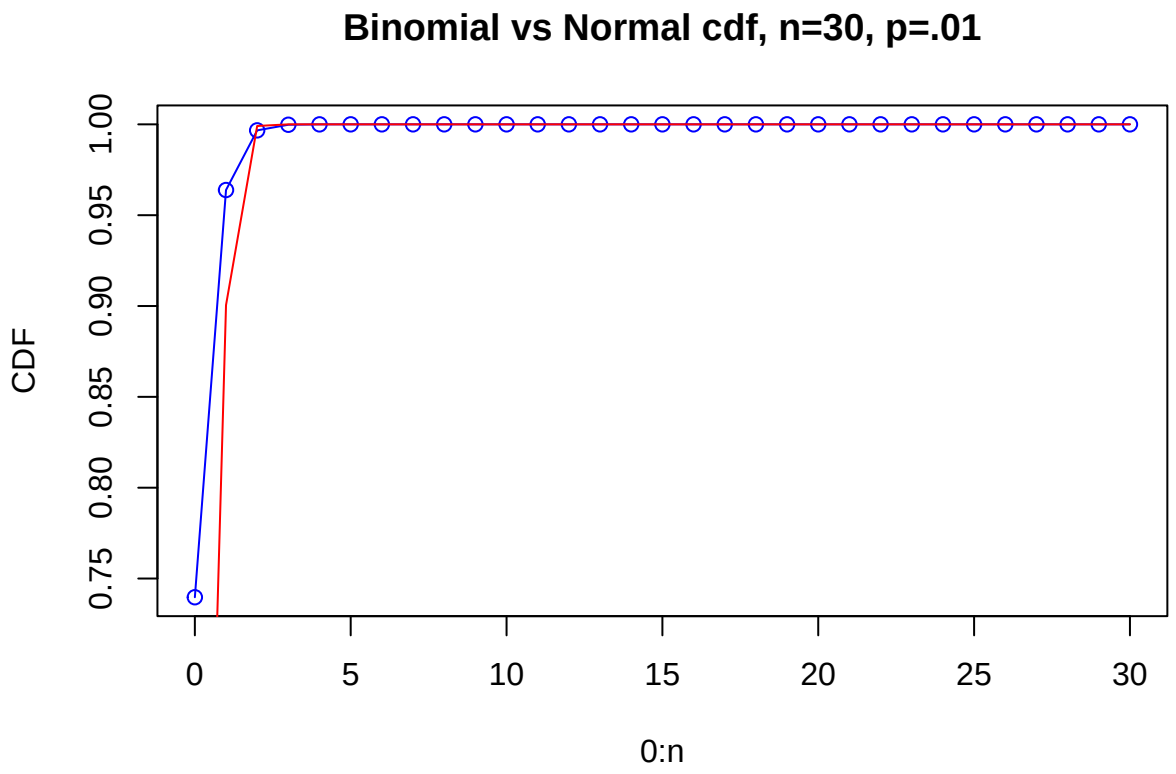

```
n <- 1000
bmean <- n*p
bsd <- sqrt((1-p)*n*p)

bcdf <- pbinom(0:n, size=n, prob=p)
ncdf <- pnorm(0:n, bmean, bsd)
plot(0:n, bcdf, type="o", main="Binomial vs Normal cdf, n=1000", ylab="CDF", col="blue")
lines(0:n, ncdf,col="red")
```



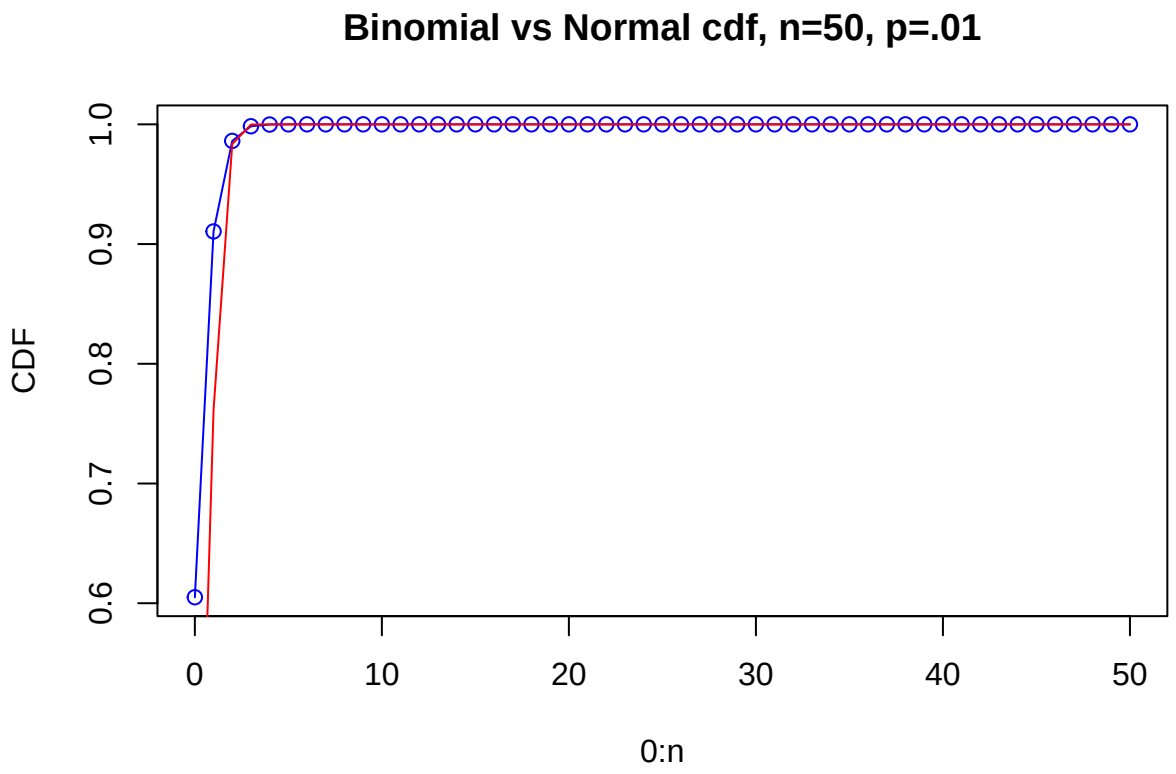

```r
# (c) Repeat the above exercise with a different value of p, for example p = 0.1.
n <- 10
p <- .01
bmean <- n*p
bsd <- sqrt((1-p)*n*p)

bcdf <- pbinom(0:n, size=n, prob=p)
ncdf <- pnorm(0:n, bmean, bsd)
plot(0:n, bcdf, type="o", main="Binomial vs Normal cdf, n=10, p=.01", ylab="CDF", col="blue")
lines(0:n, ncdf,col="red")
```



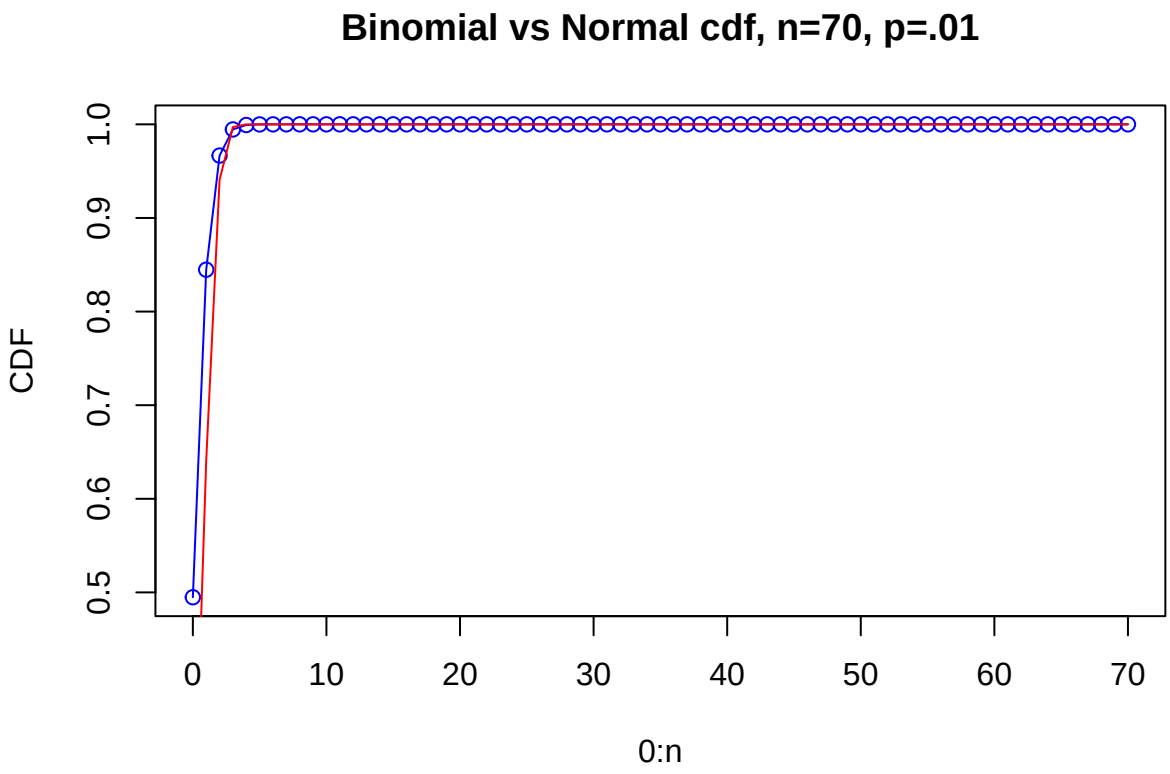

```
n <- 30
bmean <- n*p
bsd <- sqrt((1-p)*n*p)

bcdf <- pbinom(0:n, size=n, prob=p)
ncdf <- pnorm(0:n, bmean, bsd)
plot(0:n, bcdf, type="o", main="Binomial vs Normal cdf, n=30, p=.01", ylab="CDF", col="blue")
lines(0:n, ncdf,col="red")
```



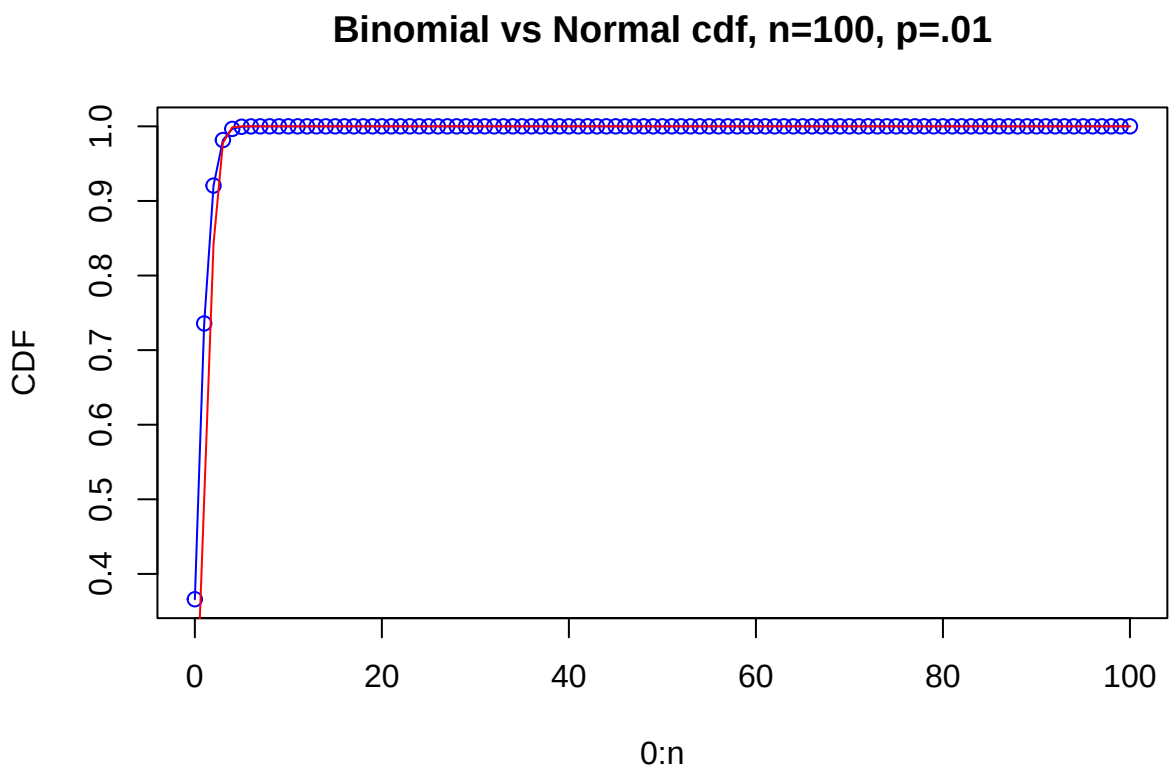

```
n <- 50
bmean <- n*p
bsd <- sqrt((1-p)*n*p)

bcdf <- pbinom(0:n, size=n, prob=p)
ncdf <- pnorm(0:n, bmean, bsd)
plot(0:n, bcdf, type="o", main="Binomial vs Normal cdf, n=50, p=.01", ylab="CDF", col="blue")
lines(0:n, ncdf,col="red")
```



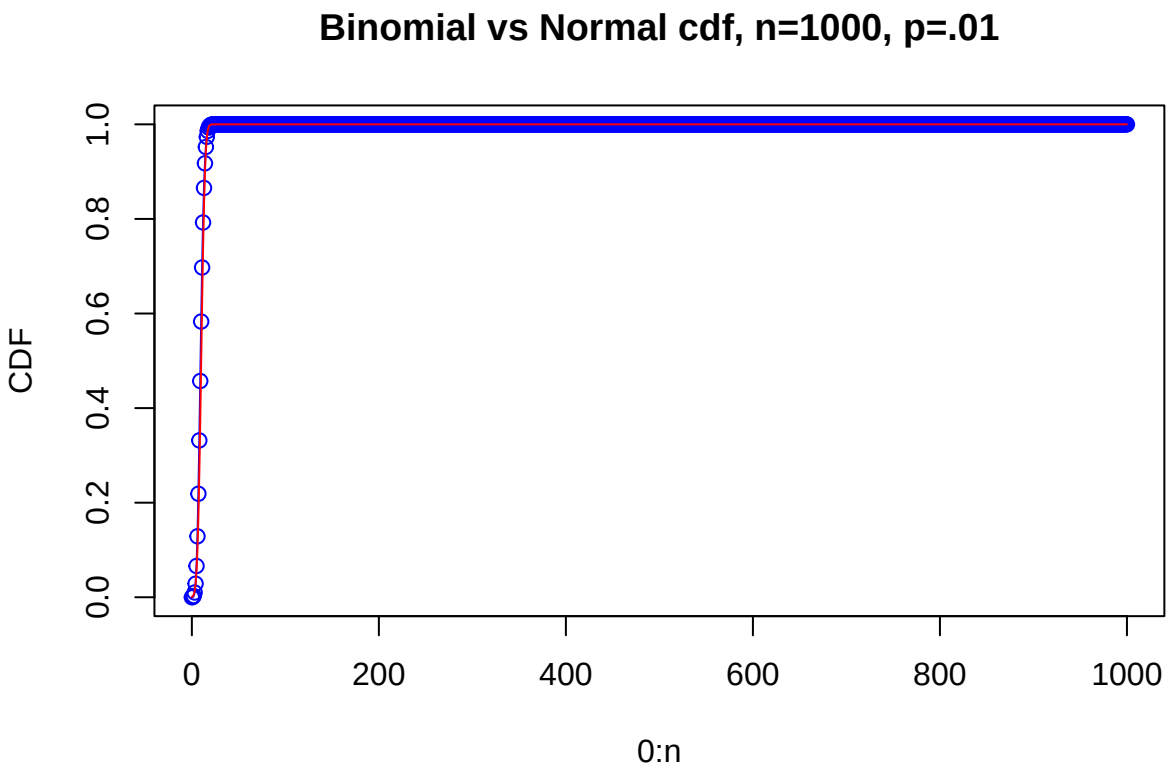

```
n <- 70
bmean <- n*p
bsd <- sqrt((1-p)*n*p)

bcdf <- pbinom(0:n, size=n, prob=p)
ncdf <- pnorm(0:n, bmean, bsd)
plot(0:n, bcdf, type="o", main="Binomial vs Normal cdf, n=70, p=.01", ylab="CDF", col="blue")
lines(0:n, ncdf,col="red")
```




```
n <- 100
bmean <- n*p
bsd <- sqrt((1-p)*n*p)

bcdf <- pbinom(0:n, size=n, prob=p)
ncdf <- pnorm(0:n, bmean, bsd)
plot(0:n, bcdf, type="o", main="Binomial vs Normal cdf, n=100, p=.01", ylab="CDF", col="blue")
lines(0:n, ncdf,col="red")
```




```
n <- 1000
bmean <- n*p
bsd <- sqrt((1-p)*n*p)

bcdf <- pbinom(0:n, size=n, prob=p)
ncdf <- pnorm(0:n, bmean, bsd)
plot(0:n, bcdf, type="o", main="Binomial vs Normal cdf, n=1000, p=.01", ylab="CDF", col="blue")
lines(0:n, ncdf,col="red")
```


Binomial vs Normal cdf, n=1000, p=.01
CDF
0.0
0.2
0.4
0.6
0.8
1.0
0
200
400
600
800
1000
0:n