Normal and Binomial Model Test 2023 Answer Sheet

YOUR NAME HERE

2023-12-13

```
set.seed(200)
```

On my honor, I have neither given nor received unauthorized aid. Please write your name here to acknowledge that you have lived up to the honor pledge: Ryan Cheng

1. Write the code and answers for Part 1 here. Make sure to give all answers as complete sentences.

a.

```
dbinom(2, 9, 0.4)
```

```
## [1] 0.1612431
```

The probability of this occurring is 16.1%.

b.

```
pbinom(2, 9, 0.4)
```

```
## [1] 0.231787
```

The probability that at most 2 people in a group of 9 receive persale codes is 23.2%.

c.

```
9*0.4
```

```
## [1] 3.6
```

```
sqrt(9*0.4*0.6)
```

```
## [1] 1.469694
```

Y'all should have expected to receive 3.6 presale codes with a standard deviation of 1.47 codes.

d.

```
0.6^9
```

```
## [1] 0.0100777
```

Yes, I would suspect that Ticketmaster's 40% claim was false. The chance that no one in your group of 9 received a presale code is the same as the chance that everyone did not receive a code, which can be calculated as 0.6 (the probability of not receiving a code) to the ninth (nine people in a row). This turns out to be a 1% chance, making Ticketmaster's claim very suspicious.

e.

```
pbinom(5, 11, 0.6)
```

```
## [1] 0.2465019
```

```r
1 - pbinom(5, 11, 0.4)
```

```
## [1] 0.2465019
```

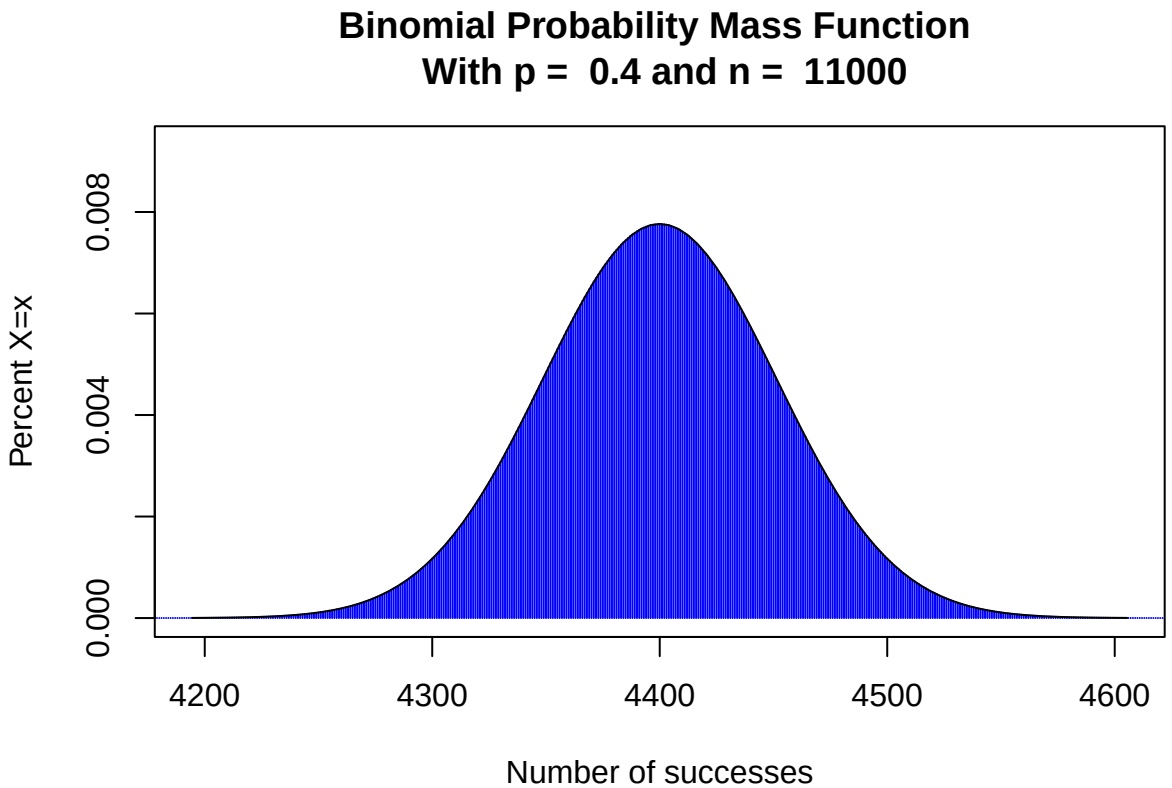
The probability that 6 or more of these 11 people got presale codes is 24.7%.

f.

```r
n <- 11000
p <- 0.4

plot( 0:n, dbinom( 0:n, n, p),
      main = paste("Binomial Probability Mass Function \n", "With p = ", p, "and n = ", n),
      xlim = c(n*p - 4* sqrt(n*p*(1-p) ), (n*p + 4* sqrt(n*p*(1-p) )) ),
      ylim = c(0, dbinom( round(n*p), n, p)*1.2 ),
      col = "blue",
      pch = 20,
      cex = 2,
      xlab = "Number of successes",
      ylab = "Percent X=x",
      type = "h",
            )

#add Normal model curve using the mean and sd of Binomial model
curve( dnorm( x, n*p, sqrt(n*p*(1-p) )  ),
       add = TRUE)
```



As the number of trials increases, the binomial model converges to the normal model, as shown in the graph above. The blue represents the binomial model, and the black outline represents the normal model. As you can see, they are a perfect fit. Therefore, we can now use the normal model to answer the rest of this question.

```r
1 - pnorm(6000, n*p, sqrt(n*p*(1-p)))
```

```
## [1] 0
```

The probability that at least 6000 people receive presale codes out of 11000 people is 0%.

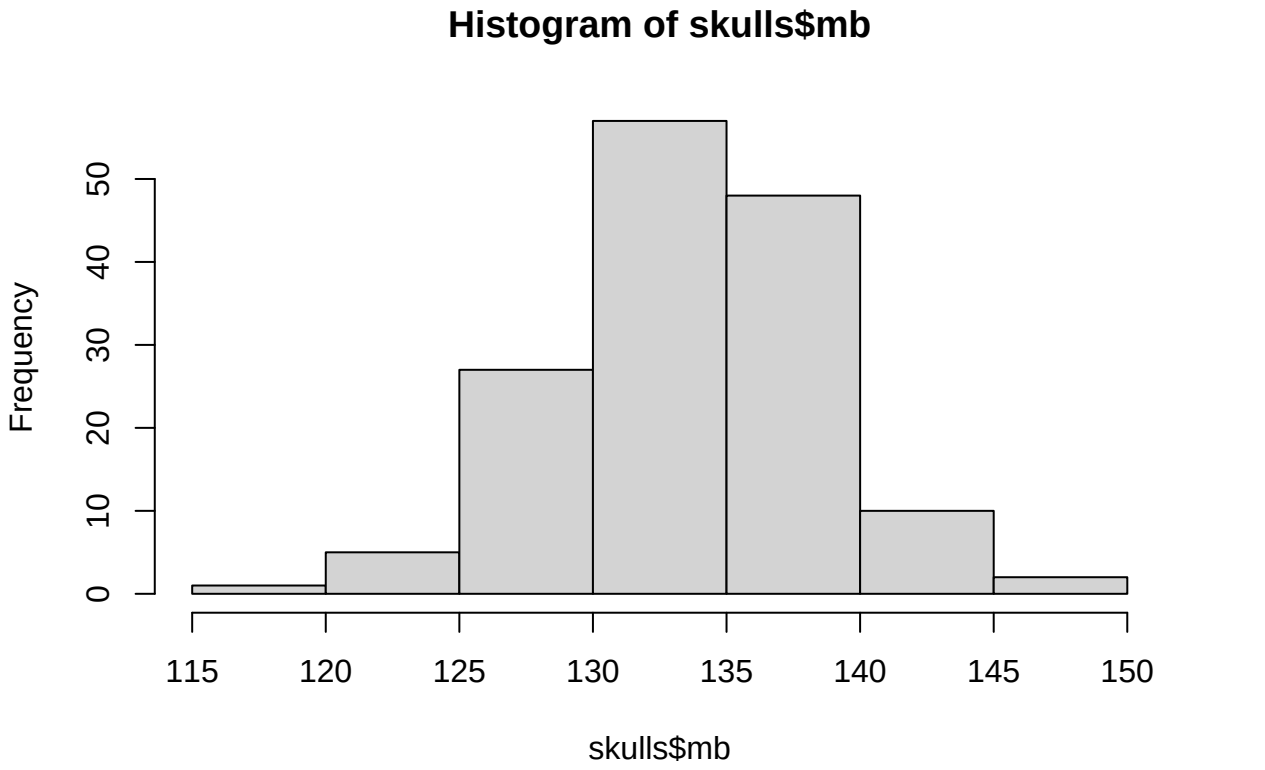
2

. Write the code and answers for Part 2 here. Make sure to give all answers as complete sentences.

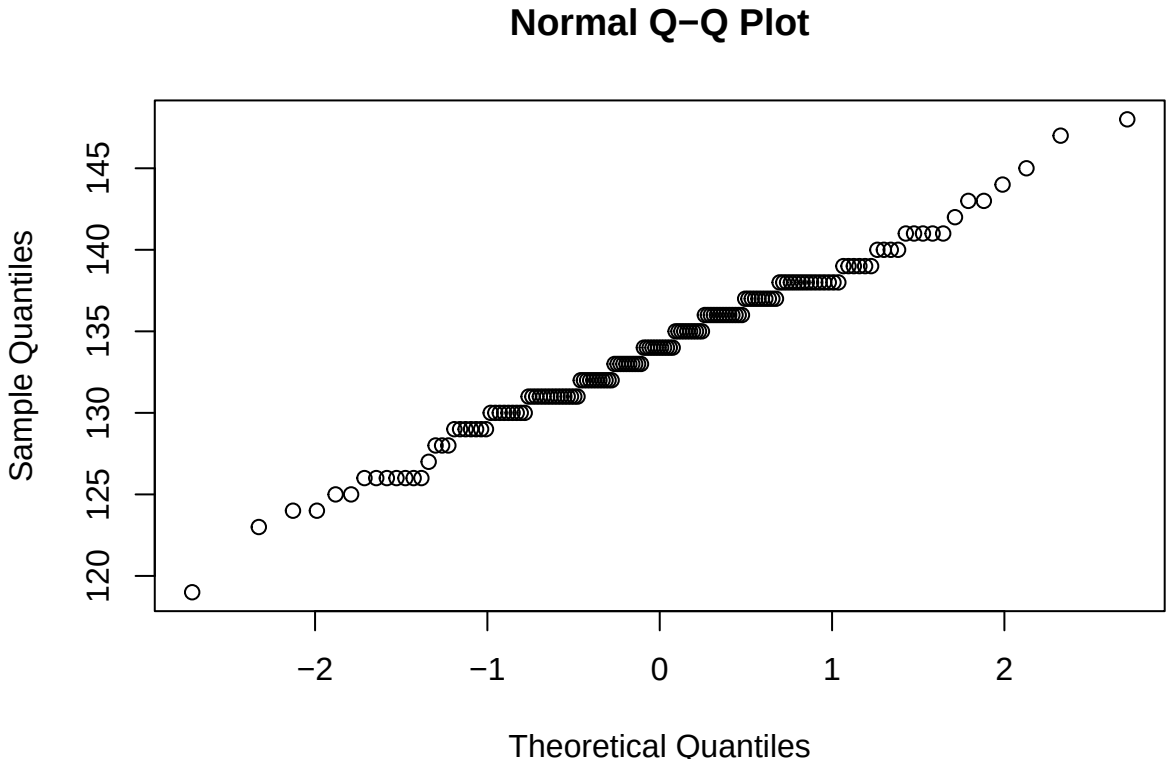
First read in the data:

```
skulls <- read.csv("skulls.csv")
```

```
hist(skulls$mb)
```




```
qqnorm(skulls$mb)
```



The maximum breadth of the skulls appears not to follow the Normal distribution. Looking at the histogram of the skulls' mb, it is evidently unimodal; however, it is not symmetrical, which it would be if it followed the Normal distribution. Now let's take a look at the Q-Q plot. A Q-Q plot of a Normal distribution should display a linear line. However, we see this is not the case. There are slight jumps that deviate from the "normal" linear line. Therefore, combined with what we know fom the histogram, we can conclude that the maximum breadth of the skulls does not follow the Normal distribution.

3

. Write the code and answers for Part 3 here. Make sure to give all answers as complete sentences.

a.

```
qnorm(0.9, 8, 2)
```

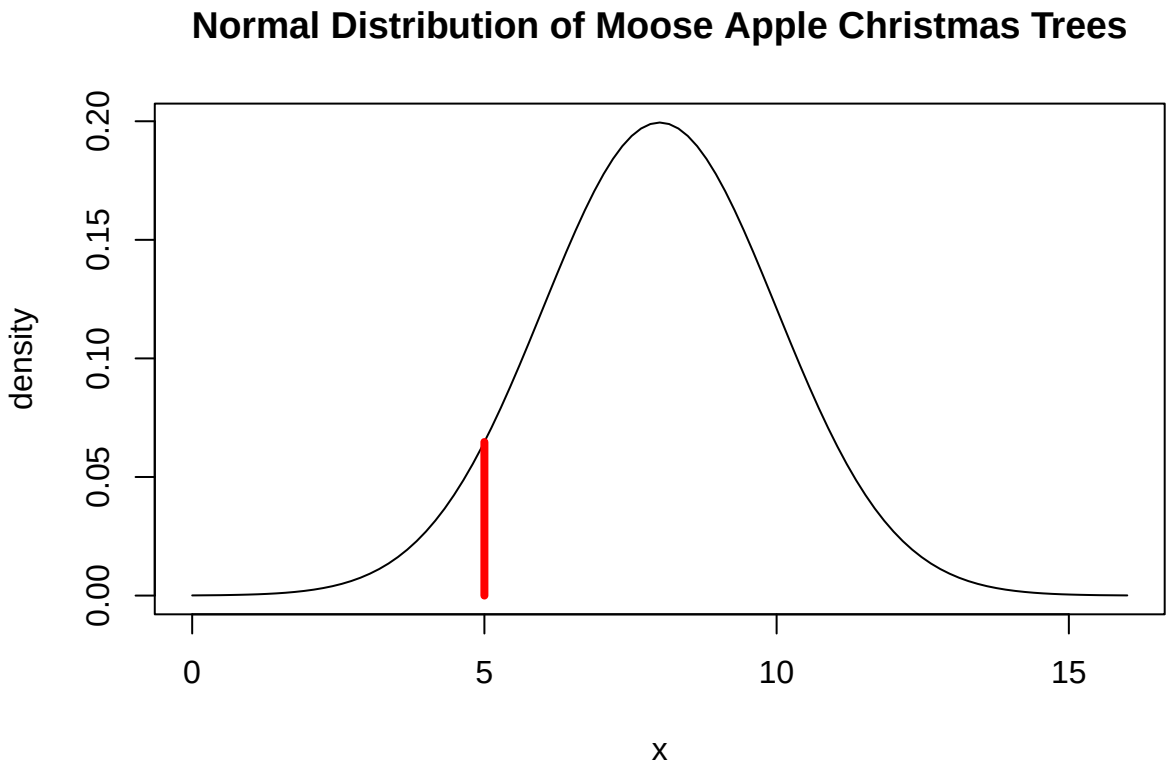
```
## [1] 10.5631
```

It would have to be at least 10.56 feet tall.

b.

```
#draw the Normal distribution
curve( dnorm(x, mean=8, sd=2),
       xlim=c(0, 16),  #pick your xlim by taking the mean - 4sd and mean +4sd
       main="Normal Distribution of Moose Apple Christmas Trees",
       ylab="density")

#add a segment at our value of note
segments( 5, 0,
          5, dnorm(5, mean=8, sd=2),
          col = "red",
          lwd = 4)
```



c.

```
1 - pnorm(5, 8, 2)
```

```
## [1] 0.9331928
```

93.3% of trees are able to be sold.

d.

```
sample.christmas.trees = rnorm(2000, mean=8, sd=2)
head(sample.christmas.trees)
```

```
## [1] 8.169513 8.452921 8.865113 9.116130 8.119511 7.770718
```

e.

```
sellable.christmas.trees = sample.christmas.trees[sample.christmas.trees > 5]
head(sellable.christmas.trees, 10)
```

```
##  [1]  8.169513  8.452921  8.865113  9.116130  8.119511  7.770718  5.958843
##  [8]  7.405897  8.336300 10.839745
```

```
length(sellable.christmas.trees)
```

```
## [1] 1877
```

1846 trees could have been sold.