Assignment 1

Tim But, Mehmet Bedirhan Gursoy, Vincentas Ryliskis, group 035

6 November 2023

Setting a Seed

In order to ensure that the results in this report are reproducible, seed (123) was used when generating random numbers.

```
set.seed(123)
```

Task 1

a)

Suppose the following:
- P(D) is the probability of a positive cancer diagnosis
- P(C) is the probability of an individual having cancer

Then, given the information given in exercise 1.3:

$$P(D|C) = 0.95$$

$$P(D|C') = 0.05$$

$$P(C) = 0.004$$

To calculate the probability of a random person being given a positive cancer diagnosis, we must add the probabilities that a positive cancer diagnosis is given to a person with cancer, as well as that a positive cancer diagnosis is given to a person without cancer:

$$P(D) = P(D|C) * P(C) + P(D|C') * P(C')$$

$$= 0.95 * 0.04 + 0.05 * 0.996$$

$$\approx 0.054$$

This differs from probability we are asked to calculate in the Exercise 1.3, as there we are asked to calculate the probability of someone having cancer, given that they have a cancer diagnosis, or: P(C|D). These probabilities refer to different events, as the latter already assumes P(D) as being true.

b)

As mentioned above, we need to calculate P(C|D). We can calculate this value using Bayes' Theorem

$$P(C|D) = \frac{P(D|C) * P(C)}{P(D)} = \frac{0.95 * 0.004}{0.0536} \approx 0.071$$

c)

The two events, that a person has cancer and that their test is positive, are dependent. This is shown by the fact that probabilities P(C|D) and P(C) are different, as shown below (based on calculations done in parts a) and b)):

$$P(C|D) = 0.071$$

$$P(C) = 0.004$$

The fact that P(C|D) is larger than P(C) shows that a cancer diagnosis leads to an increased risk in cancer, when compared to the risk of cancer without accounting for a diagnosis.

Task 2

a)

Below is sample space Ω, for which each outcome is equally likely:

Ω = {0-minutes, 1-minute, 2-minutes, 3-minutes, 4-minutes, 5-mimutes, 6-minutes, 7-minutes, 8-minutes, 9-minutes, 10-minutes, 11-minutes, 12-minutes, 13-minutes, 14-minutes}

b)

Let W be the wait time. P(W>=5) = 1-(P(W)<5).
P(W = 0) is 5/15 (as it "consumed" what was previously P(0<W<=4)
P(W = 1) occurs when the individual is 14 minutes late, or 1/15.
P(W = 2) occurs when the individual is 13 minutes late, or 1/15.
P(W = 3) occurs when the individual is 12 minutes late, or 1/15.
P(W = 4) occurs when the individual is 11 minutes late, or 1/15.
P(W<0) = 0, as one can not be negatively late.

The sum of all of these probabilities is 9/15, therefore P(W>=5) is (1 - 9/15) = 6/15

c)

Let X be the delay to the bus stop P(X = 0) is 5/15, as discussed above. P(0<X<5) = 0, as discussed above. Every other individual probability where X equals an integer between (and including) 5 and 14 is equal to 1/15.

Therefore, in order to calculate the expectation, we need to take the sum of the products of each probability and it's X value.

$$E(X) = \mu = \frac{(4 * 0) + 5 + 6 + 7 + 8 + 9 + 10 + 11 + 12 + 13 + 14}{15} = \frac{95}{15} = 6.333...$$

The expectation of the delay μ is equal to 6.333

d)

The variance of the delay is equal to E(Xˆ2) - E(X)ˆ2, and we already know that E(X) = 6.333. Therefore:

$$\sigma^2 = \frac{25 + 36 + 49 + 64 + 81 + 100 + 121 + 144 + 169 + 196}{15} - \mu^2 = \frac{985}{15} - \mu^2$$

$$= 25.555...$$

e)

The approximate average distribution of the average waiting time is as follows, in accordance with the Central Limit Theorem:

$$X_160 \sim N(\mu, \frac{\sigma^2}{160})$$

Task 3

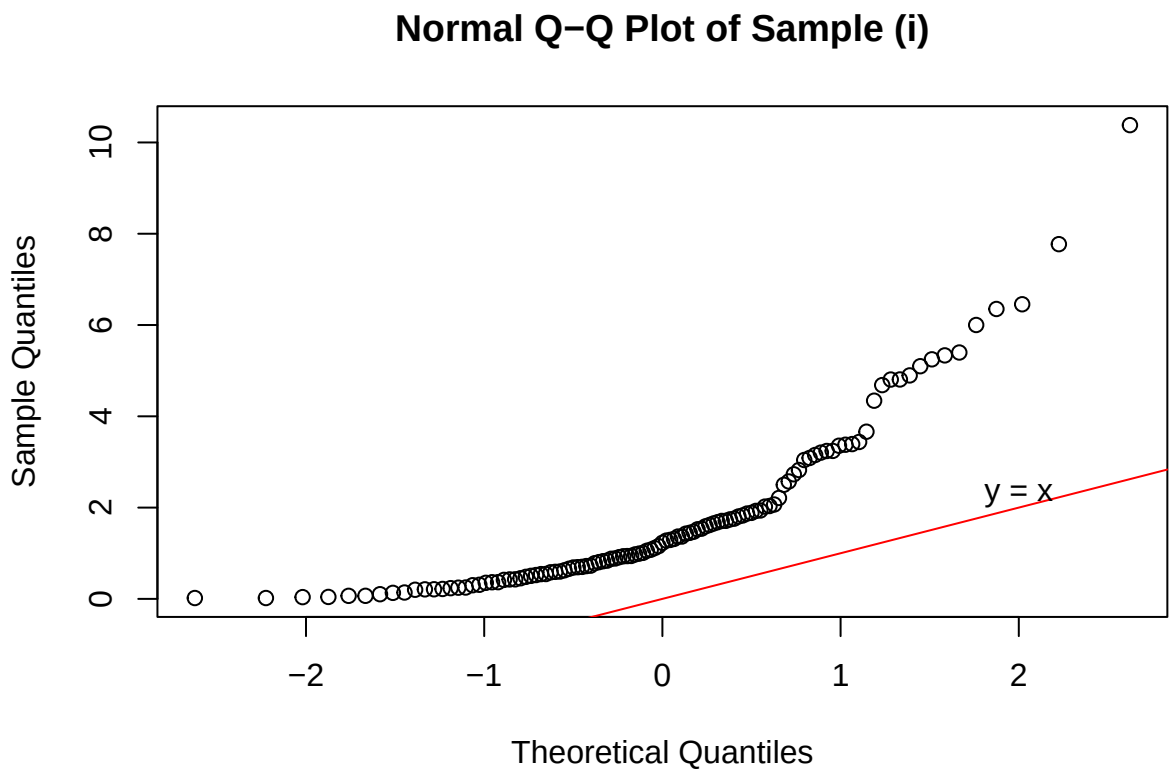
a)

(i) The sample of size 115 from the chisquared distribution with degrees of freedom 2 was generated using the code below:

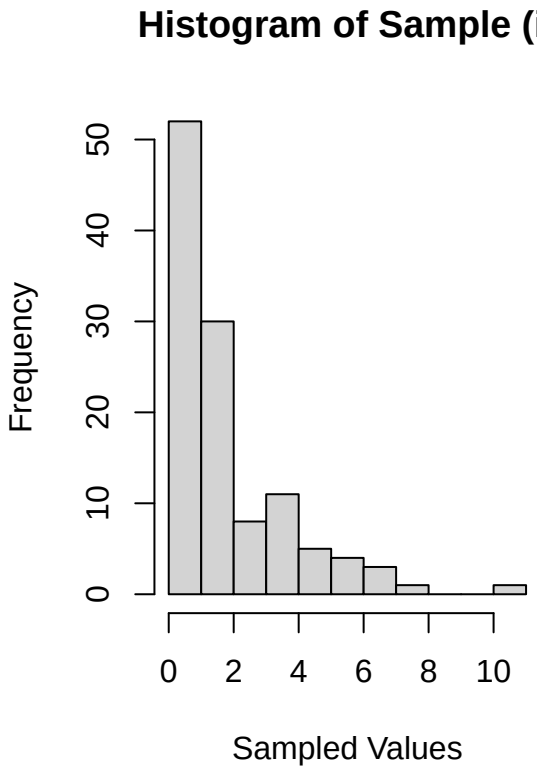
```
chi_n = 115
chi_df = 2
chi_x = rchisq(sequence(chi_n), df=chi_df)
```

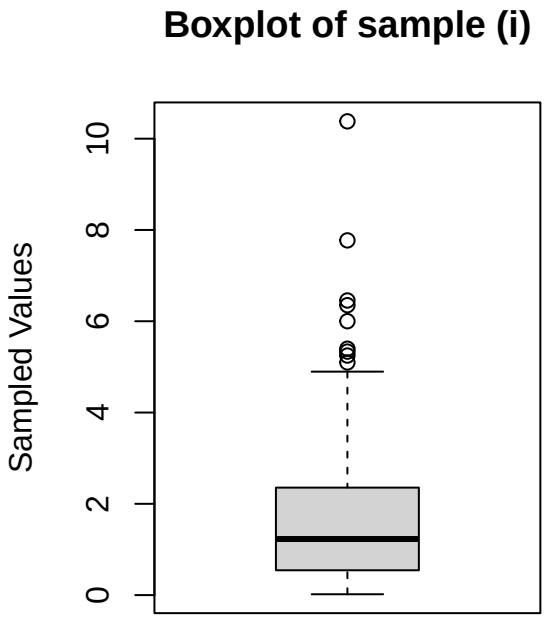
Below are the resulting qqplot, histogram, and boxplot, with the code used to generate the respective graphs.

```
qqnorm(chi_x, main = "Normal Q-Q Plot of Sample (i)")
abline(0, 1, col = 'red')
text(2, 2.25, "y = x")
```




```
par(mfrow = c(1,2))
hist(chi_x, main = "Histogram of Sample (i)", xlab = "Sampled Values")
boxplot(chi_x, main = "Boxplot of sample (i)", ylab ="Sampled Values")
```





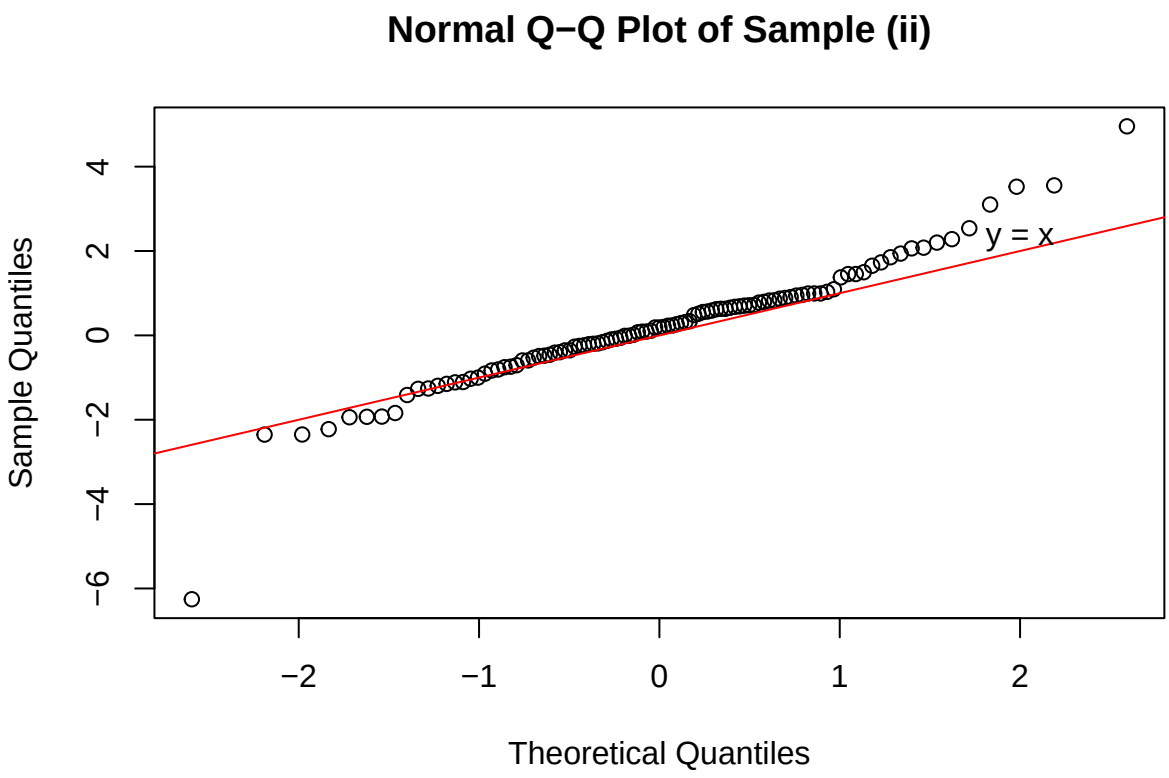
The results confirm that the sample for part (i) should not use a normal distribution as a model distribution. The Q-Q plot for this sample has a very distinctive "S" shape, and is above and to the left of the line y=x, indicating that it has light tails, and is heavily right skewed. Looking at the histogram and boxplot, the heavy asymmetry predicted by the Q-Q plot (skewed right) can be seen, as the left tails are nearly non-existent. This results in all outliers existing in the right tail, which is best seen in the boxplot.

(ii) The sample of size 105 from the t-distribution with 4 degrees of freedom was generated using the code below:

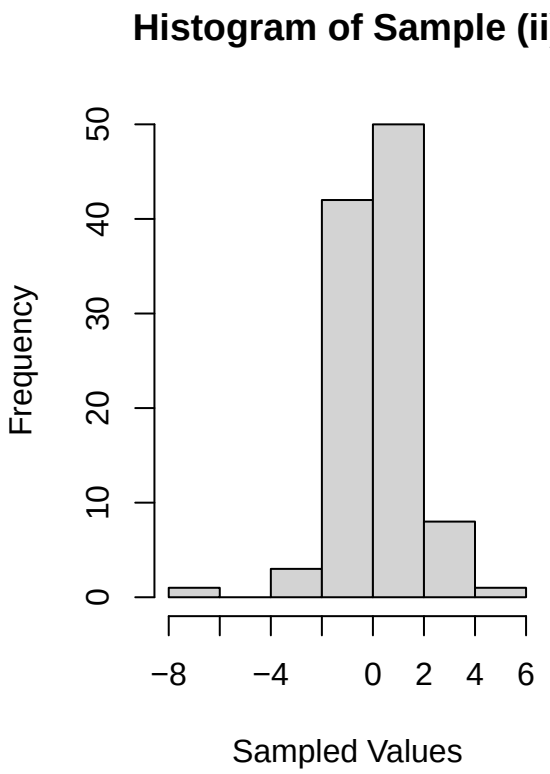
```
par(mfrow = c(1,3))

t_n = 105
t_df=4
t_x = rt(sequence(t_n), df=t_df)
```

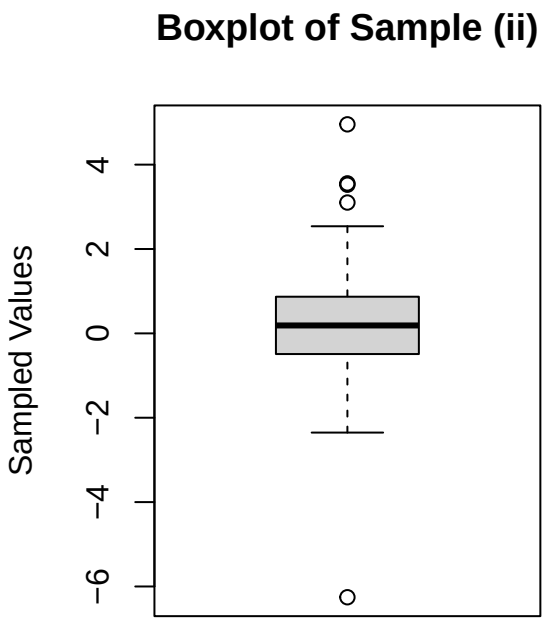
Below are the resulting qqplot, histogram, and boxplot, with the code used to generate the respective graphs.

```
qqnorm(t_x, main = "Normal Q-Q Plot of Sample (ii)")
abline(0, 1, col = 'red')
text(2, 2.25, "y = x")
```




```r
par(mfrow = c(1,2))

hist(t_x, main = "Histogram of Sample (ii)", xlab = "Sampled Values")
boxplot(t_x, main = "Boxplot of Sample (ii)", ylab = "Sampled Values")
```





The results provide limited insight into whether sample (ii) should use a normal distribution as a model distribution. The Q-Q plot for this sample largely stays on the line y=x, with somewhat heavy tails. Looking at the histogram and boxplot, the sample is largely symmetrical, if ever-so-slightly skewed right (as can be seen by the mean being to the right of zero, and tails and outliers being more prevalent in that direction). However, this asymmetry could potentially be attributed to the limited sample size. Ultimately, a normal distribution as a model distribution would be more appropriate in this case than in sample (i), however, it is still not a perfect fit due to the heavy tails of the sample.

b)

(i) For a standard normal distribution :
Probability of an outcome smaller than 3: results in roughly 0.999

```
smallerThan3 = pnorm(3)
print(smallerThan3)
```

```
## [1] 0.9986501
```

Probability of an outcome larger than -0.5: results in roughly 0.691

```r
biggerthanM0.5 = 1-pnorm(-0.5)
print(biggerthanM0.5)
```

```
## [1] 0.6914625
```

Probability of an outcome between -1 and 2: results in roughly 0.819

```r
betweenM1and2 = pnorm(2) - pnorm(-1)
print(betweenM1and2)
```

```
## [1] 0.8185946
```

(ii) For a normal distribution with mean 3 and variance 4 :
Since the variance is 4, the standard distribution will be 2. Probability of an outcome smaller than 3: results in roughly 0.5

```r
smallerThan3 = pnorm(3, mean = 3, sd = 2)
print(smallerThan3)
```

```
## [1] 0.5
```

Probability of an outcome larger than -0.5: results in roughly 0.960

```r
biggerthanM0.5 = 1-pnorm(-0.5, mean = 3, sd = 2)
print(biggerthanM0.5)
```

```
## [1] 0.9599408
```

Probability of an outcome between -1 and 2: results in roughly 0.286

```r
betweenM1and2 = pnorm(2, mean = 3, sd = 2) - pnorm(-1, mean = 3, sd = 2)
print(betweenM1and2)
```

```
## [1] 0.2857874
```

Value for which 95% of outcomes are smaller: results in roughly 6.290

```r
NFpercSmaller = qnorm(0.95, mean = 3, sd = 2)
print(NFpercSmaller)
```

```
## [1] 6.289707
```

Below is a table describing the differences between the values calculated in part (ii), and those found in the book. Note that those found in the book have not been rounded (by us), while the values calculated in R have.

Assuming a normal distribution with mean 3 and variance 4...		
Value	Book Value	R Value
Probability of an outcome smaller than 3	z = ((3-3)/2) = 0 Z(0) = 0.5	0.5
Probability of an outcome larger than -0.5	z = -1*((-0.5-3)/2) = 1.75 Z(1.75) = 0.9599	0.960
Probability of an outcome between -1 and 2	z1 = (-1-3)/2 = -2 z2 = (2-3)/2 = -1/2 Z(-1/2) - Z(-2) = = 0.3085-0.0228 = 0.2857	0.286
Value for which 95% of outcomes are smaller	Z((x-3)/2) = 0.95 (Z^(-1)(0.95) *2) + 3 = x x = 1.645*2 + 3 = 6.29	6.290

(iii) With the following method, we take a sample from the standard normal distribution, and convert it to a distribution with mean -1 and standard deviation 5:

```
mean = -1
sd = 5

samples = rnorm(1000)

corrected_samples = (samples * sd) +mean

sample_mean = mean(corrected_samples)
sample_sd = sd(corrected_samples)
```

After the correction, the sample mean is roughly -0.940, and the sample standard deviation is roughly 5.006

```
cat("Sample Mean:", sample_mean, "| Sample std. dev.:",sample_sd, "\n")
```

```
## Sample Mean: -0.9403835 | Sample std. dev.: 5.005643
```

(iv) Below is a verification function, made to take a min and max value, and compare the proportion of samples within a sample size of 100 and a sample size of 100000 that fall in that range. The default for min is -inf, and for max is inf, so if a min or max aren't given, they are assumed to be infinite. Below are our results

```
verification_function <- function(min = -Inf, max = Inf){

  sample_100 = rnorm(100)
  sample_100k = rnorm(100000)

  prob_sample_100 = mean(sample_100 > min & sample_100 < max)
```

```
  prob_sample_100k = mean(sample_100k > min & sample_100k < max)

  cat("Probability (Sample 100) for value between", min, max, ":", prob_sample_100, "\n")
  cat("Probability (Sample 100k) for value between", min, max, ":", prob_sample_100k, "\n")
}

verification_function(max = 3)
```

```
## Probability (Sample 100) for value between -Inf 3 : 1
## Probability (Sample 100k) for value between -Inf 3 : 0.99868
```

```
verification_function(min = -0.5)
```

```
## Probability (Sample 100) for value between -0.5 Inf : 0.66
## Probability (Sample 100k) for value between -0.5 Inf : 0.69379
```

```
verification_function(min = -1, max = 2)
```

```
## Probability (Sample 100) for value between -1 2 : 0.8
## Probability (Sample 100k) for value between -1 2 : 0.81713
```

For context, the results we had gotten in part (i) were as follows:
- Probability of an outcome smaller than 3: results in roughly 0.999
- Probability of an outcome larger than -0.5: results in roughly 0.691
- Probability of an outcome between -1 and 2: results in roughly 0.819

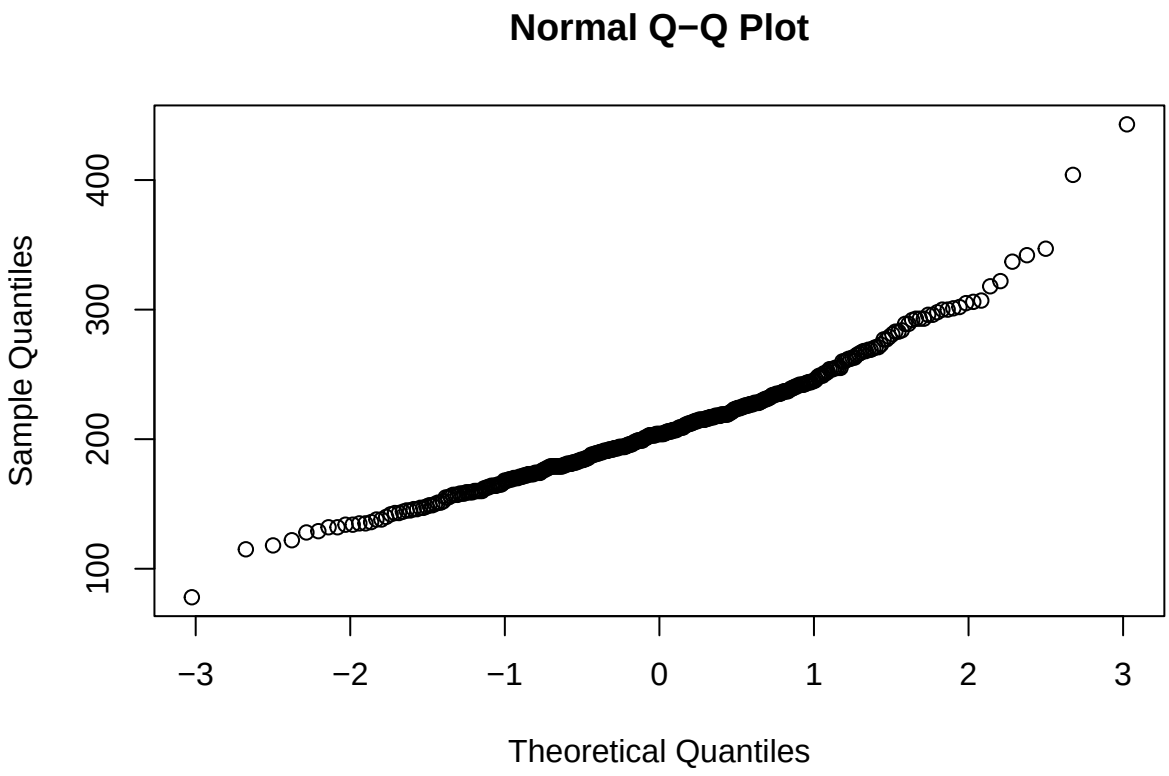
When evaluating the sample size of 100k, all results are off by less than 0.03, or 3%. Considering most statistical tests have a margin of error of 0.05, this is acceptable.

The sample of 100 performed worse, but this is expected with a lower sample size. This sample was off by over 0.3 at times.

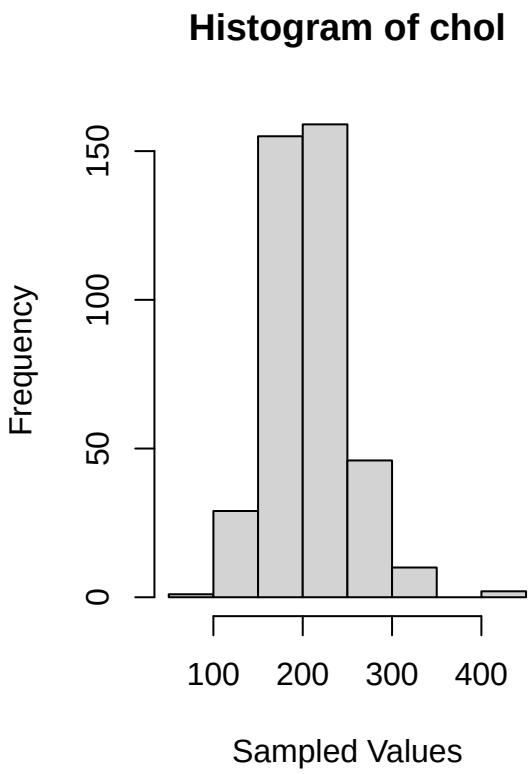
c)

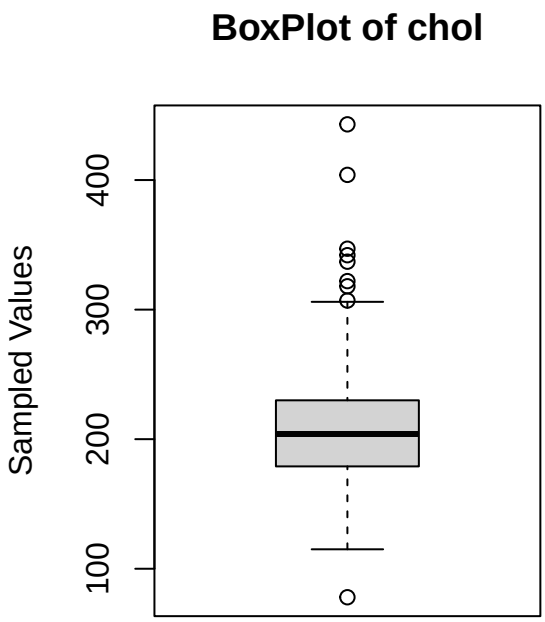
(i) For the data represented below, we hold the opinion that normality cannot be excluded. Due to the fact that the data is (for the most part) rather symmetrical, as can be seen in the histogram and boxplot, and due to the trend of the Q-Q plot being rather straight, it seems as though there is potential for normality to be an explanation. However, this dataset has a heavy right tail, leading to some asymmetry. This is best seen in the outliers on the boxplot, as well as at the top right of the Q-Q plot.

```
diabetes <- read.csv("diabetes.csv")
chol = diabetes$chol

qqnorm(chol)
```



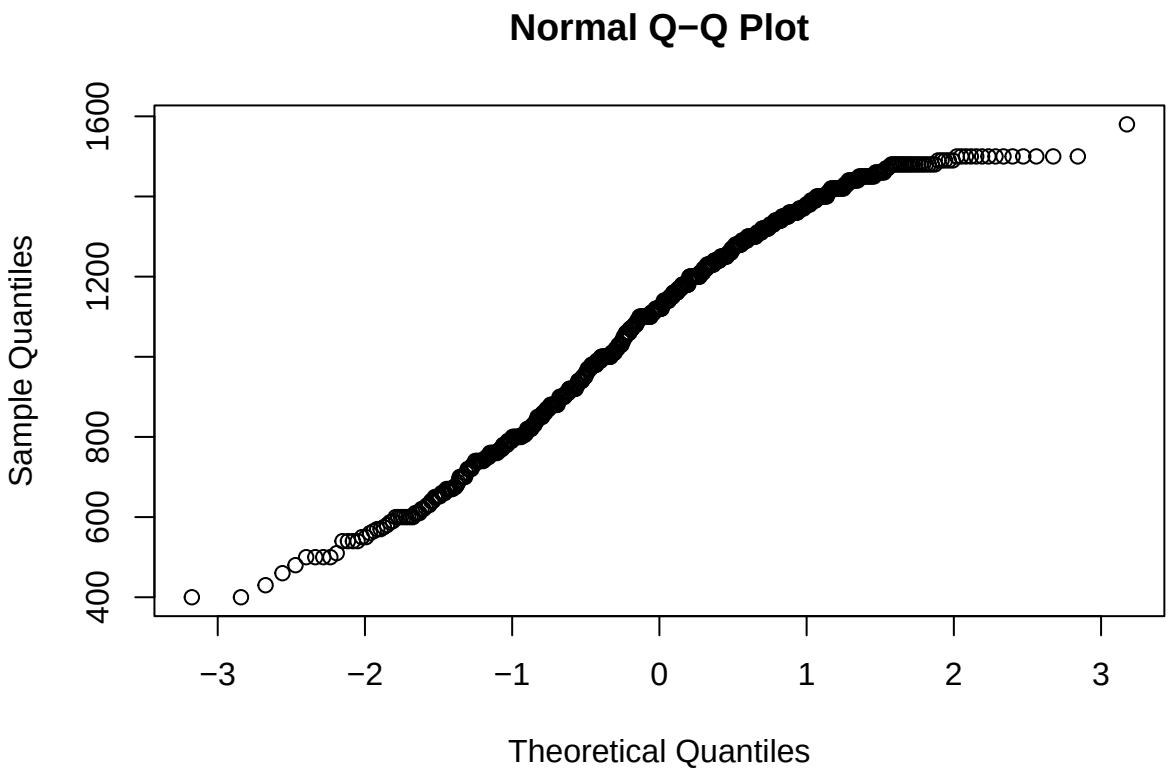

```
par(mfrow = c(1,2))

hist(chol, xlab = "Sampled Values")
boxplot(chol, ylab = "Sampled Values", main = "BoxPlot of chol")
```



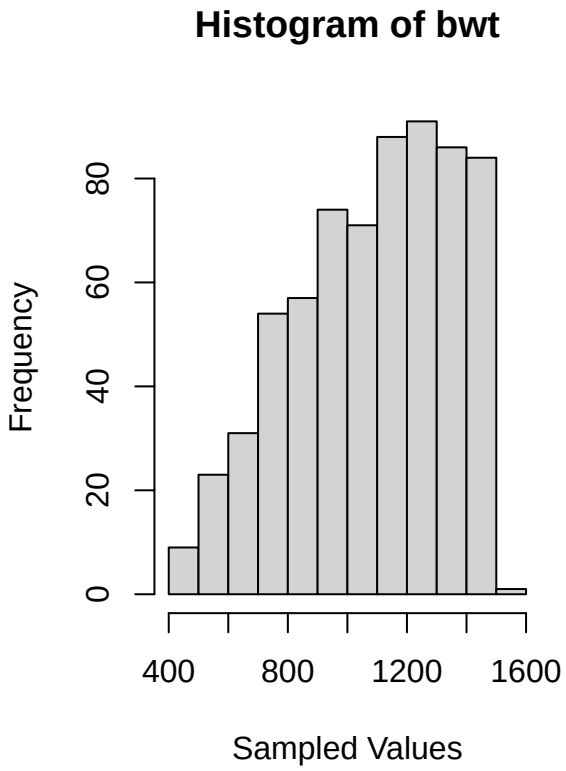


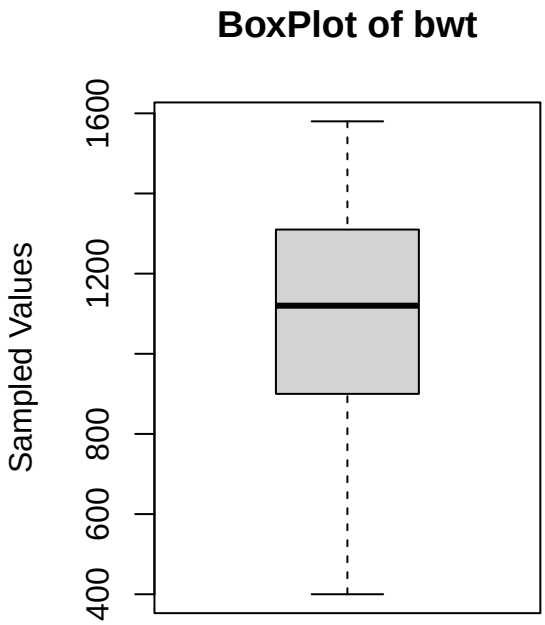
(ii) For the data represented below, we hold the opinion that the data is obviously not from a normal distribution. The shape of the Q-Q plot implies that the data is extremely uniform, and has almost no weight in the tails. The histogram appears to show a rather spread-out distribution, and while there is some falloff, it is not proportional to what would be expected in a normal distribution. This especially holds true for the right tail, which falls off incredibly quickly. The boxplot additionally implies that the three right quadrants are of roughly equal size, which would be unusual for a normal distribution, which tends to be very concentrated in the middle. This leads us to believe that this distribution is likely not normal.

```
vlbw <- read.csv("vlbw.csv")
bwt = vlbw$bwt

qqnorm(bwt)
```




```
par(mfrow = c(1,2))
hist(bwt,xlab = "Sampled Values")
boxplot(bwt, ylab = "Sampled Values", main = "BoxPlot of bwt")
```


Histogram of bwt
Frequency
0
20
40
60
80
400
800
1200
1600
Sampled Values

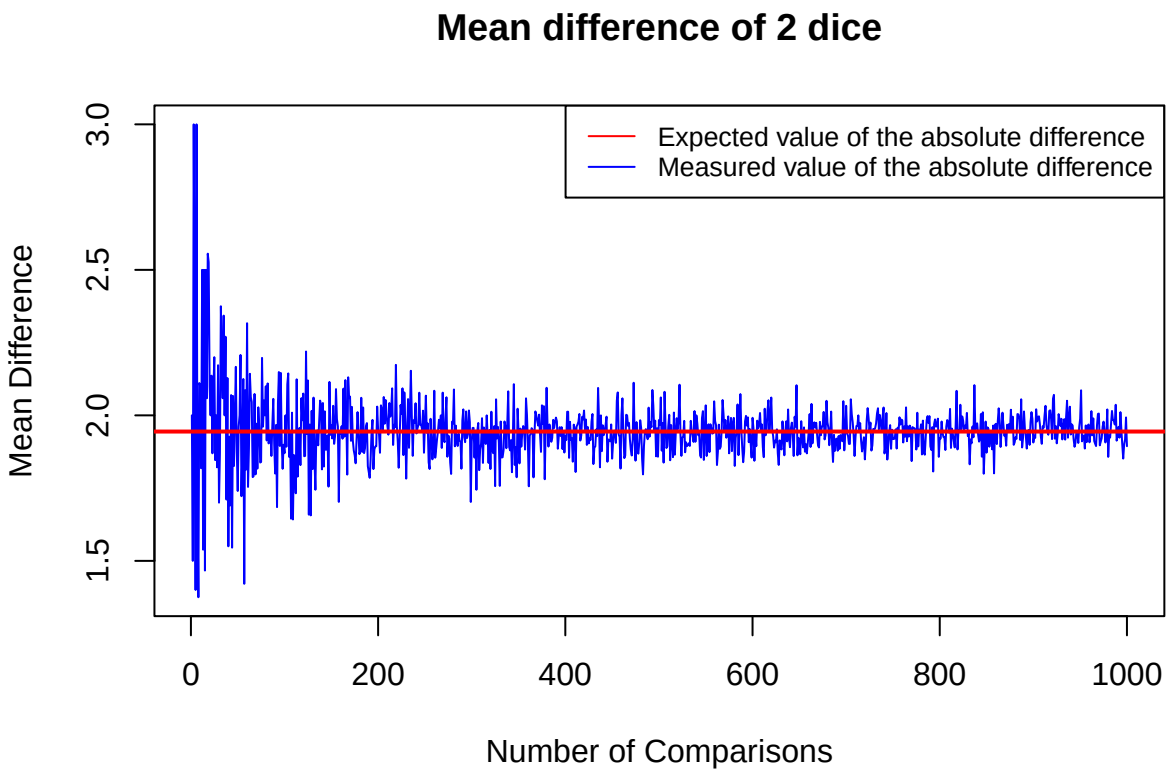

BoxPlot of bwt
Sampled Values
400
600
800
1200
1600

Task 4

a)

Below is a line chart illustrating the Law of Large Numbers using the mean difference of two dice rolls.

```
source("function02.txt")
n = 1000
meandiff = numeric(n)


for (i in 1:1000){
  meandiff[i] = mean(diffdice(i))

}

plot(meandiff, type = 'l', col = "blue", xlab = "Number of Comparisons",
     ylab = "Mean Difference", main = "Mean difference of 2 dice")

abline(h = 1.9444, col = "red", lwd = 2)

legend("topright", legend=c("Expected value of the absolute difference",
      "Measured value of the absolute difference"), col=c("red", "blue"),
      lty=1, cex=0.8)
```



.

b)

We called the function diffdice with n = 100000, in order to generate 100k differences. Then, going through all possible values, we found the proportion of each value in these 100k differences. Ultimately, the expectation value that we got was roughly 1.938, which is just slightly short of the projected 1.9444 that was assumed in part a)

```
source("function02.txt")

n = 100000
Diffs100k = diffdice(n)
expectation = mean(Diffs100k)
print(expectation)
```

```
## [1] 1.93781
```

```
expectation = 0

for (i in 0 : 6){
```

```
  probability_of_i = length(which(Diffs100k == i))/n

  expectation = expectation + (probability_of_i*i)
}
expectation = expectation

print(expectation)
```

```
## [1] 1.93781
```

```
probabiliy_of_3 = length(which(Diffs100k == 3))/n
print(probabiliy_of_3)
```

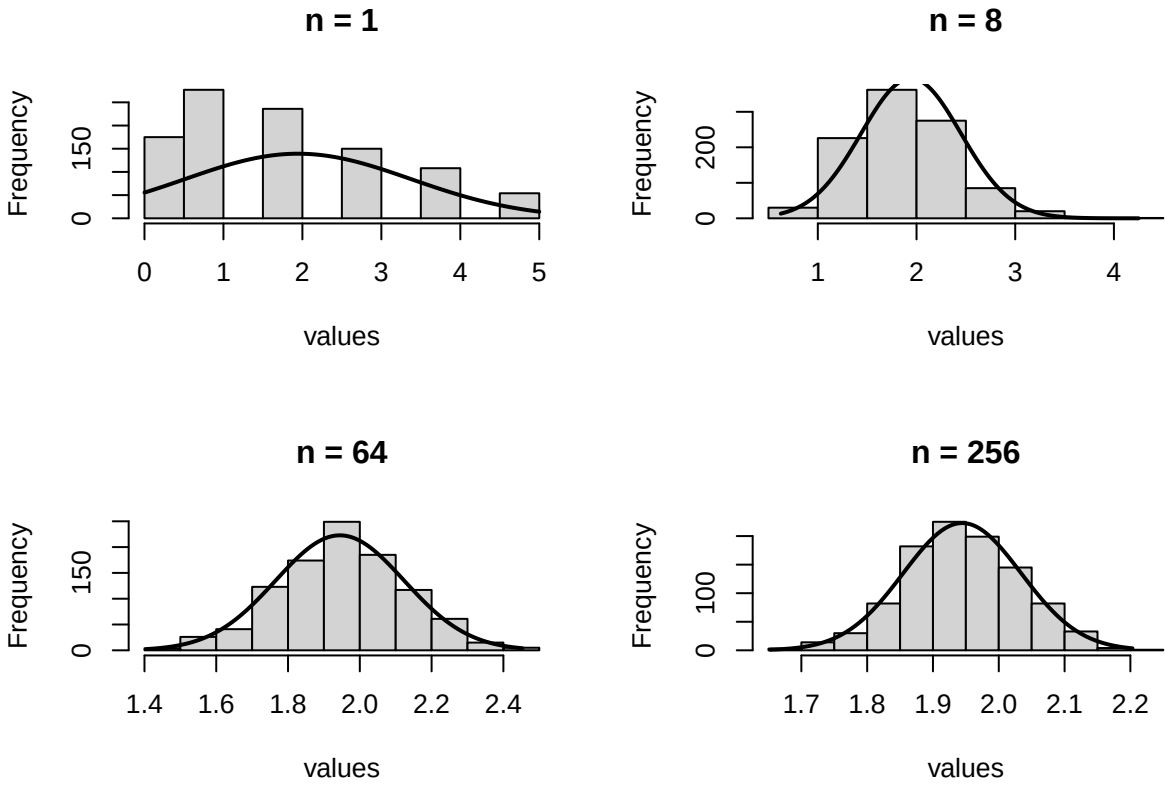
```
## [1] 0.16797
```

.

c)

Below are 4 plots visualizing the central limit theorem. Each graph uses 1000 means of diffdice, but the n value is different for each graph (the number of samples to find a mean).

```
source("function02.txt")
par(mfrow = c(2,2))
CLTGenerator = function(n = 1, sample_means = 1000){
  values = numeric(sample_means)
  for (i in 1:sample_means){
    values[i] = mean(diffdice(n))
  }
  h <- hist(values, main = sprintf("n = %s", n))
  xfit <- seq(min(values), max(values), length = 1000)
  yfit <- dnorm(xfit, mean = 1.9444, sd = sqrt(((1.4326)^2)/n))
  yfit <- yfit * diff(h$mids[1:2]) * length(values)
  lines(xfit, yfit, col = "black", lwd = 2)

}

CLTGenerator(n = 1)
CLTGenerator(n = 8)
CLTGenerator(n = 64)
CLTGenerator(n = 256)
```



d)

As the number of samples used to calculate each mean increases, the estimate of the standard deviation decreases. In the histograms created, we can see the predicted normal distribution considering the respective n as a black line. The sampled graphs match up quite well with their predicted normal distributions, illustrating the Central Limit Theorem. Theoretically, as n approaches infinity, the standard deviation approaches zero, which would result in an infinitely tall and thin distribution.