

Chapter 2

2.1

Compute the mean \bar{x} and median, m , of the six numbers: 3, 5, 8, 15, 20, 21, 24, then apply the natural log to the data.

```
x <- c(3, 5, 8, 15, 20, 21, 24)
xt <- log(x)
```

Does $\bar{x} = \tilde{x}$?

```
log(mean(x)) == mean(xt)
```

```
[1] FALSE
```

Does $m = \tilde{m}$?

```
log(median(x)) == median(xt)
```

```
[1] TRUE
```

2.2

Compute the mean \bar{x} and median of the eight numbers: 1, 2, 4, 5, 6, 8, 11, 15.

Let $f(x) = \sqrt{x}$

Apply the transformation, then compute the mean, \tilde{x} and median, m , of the transformed data.

```
x <- c(1, 2, 4, 5, 6, 8, 11, 15)
xt <- sqrt(x)
```

- Is $f(\bar{x}) = \tilde{x}$?

```
sqrt(mean(x)) == mean(xt)
```

```
[1] FALSE
```

- Is $f(m) = \tilde{m}$?

```
sqrt(median(x)) == median(xt)
```

```
[1] FALSE
```

2.4

Import the flights data.

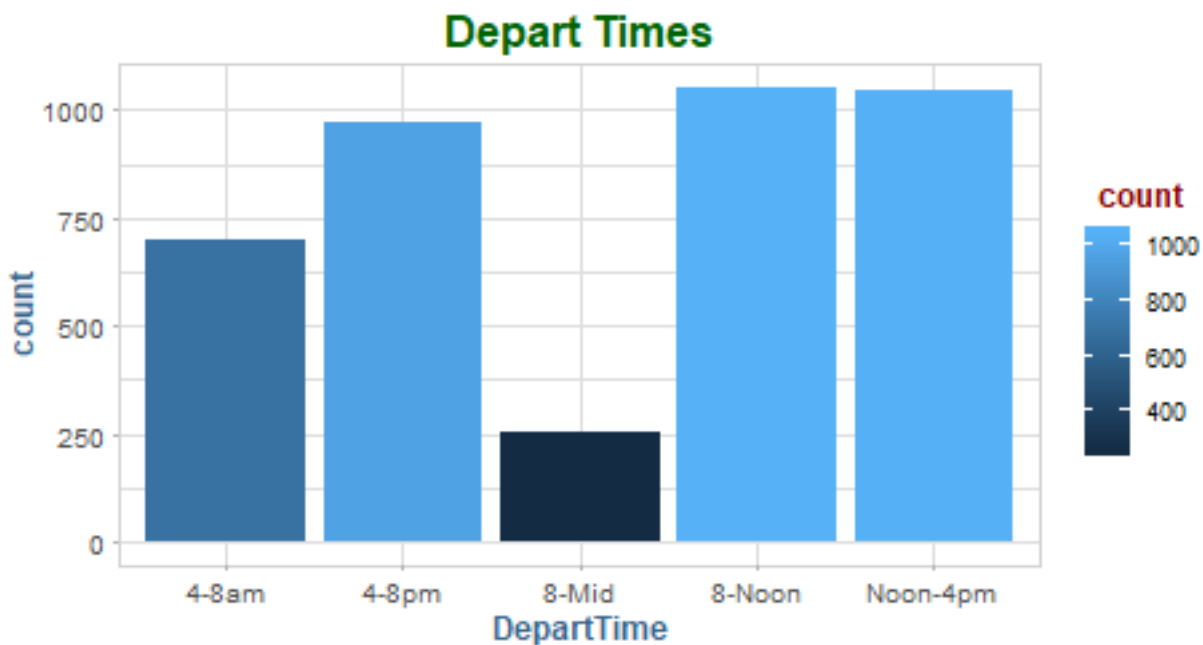
```
flights <- data.table(read.csv(paste0(data.dir, "FlightDelays.csv"),  
                               header = T))
```

a.) Create a table and bar chart of the departure times (*DepartTimes*)

```
table(flights$DepartTime)
```

4-8am	4-8pm	8-Mid	8-Noon	Noon-4pm
699	972	257	1053	1048

```
ggplot(flights, aes(DepartTime)) +  
  geom_bar(aes(fill = ..count..)) +  
  labs(title = "Depart Times")
```



b.) Create a contingency table of the variables *Day* and *Delay30*.

```
delay <- table(flights$Day, flights$Delayed30)

pretty_kable(delay, "Flight Delays")
```

Table 1: Flight Delays

	No	Yes
Fri	493	144
Mon	569	61
Sat	406	47
Sun	507	44
Thu	434	132
Tue	535	93
Wed	488	76

Show the proportions of delayed flights, by day:

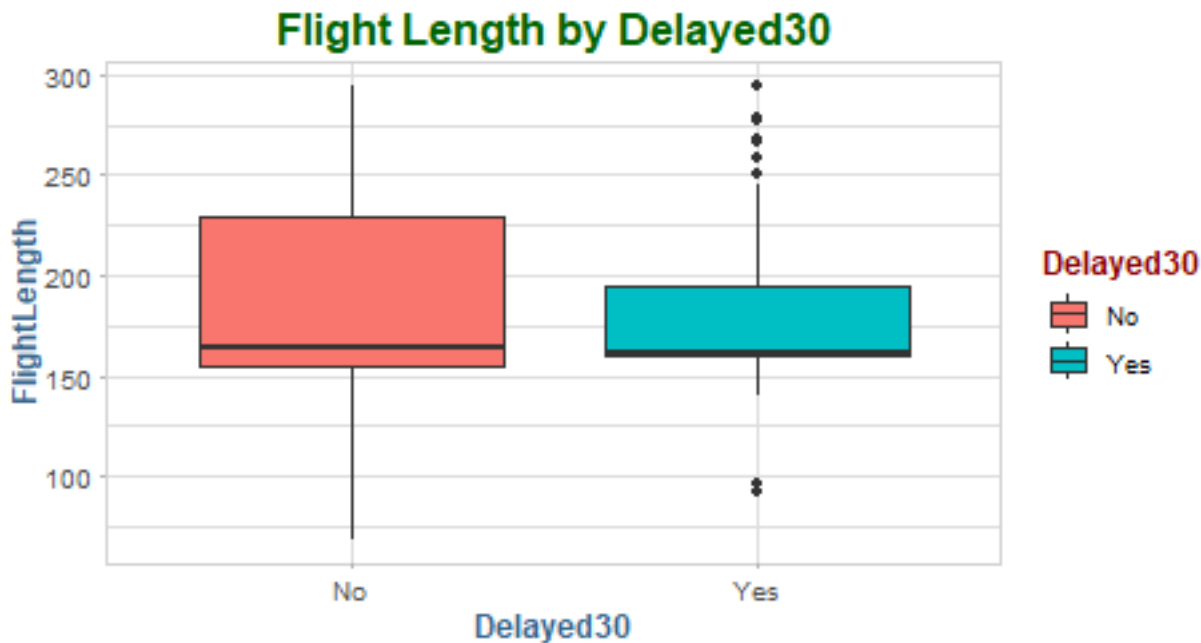
```
pretty_kable(round(prop.table(delay), 4) * 100, "Flight Delays Proportions")
```

Table 2: Flight Delays Proportions

	No	Yes
Fri	12.24	3.57
Mon	14.12	1.51
Sat	10.08	1.17
Sun	12.58	1.09
Thu	10.77	3.28
Tue	13.28	2.31
Wed	12.11	1.89

c.) Create side-by-side boxplots of the lengths of flight times, grouped by whether or not the flight was delayed at least 30 minutes:

```
ggplot(flights) +
  geom_boxplot(aes(Delayed30, FlightLength, fill = Delayed30)) +
  labs(title = "Flight Length by Delayed30")
```



d.) Do you think there is a relationship between the length of the flight and whether or not the departure time is delayed by at least 30 minutes?

The average flight time is the same, however, the flights that are delayed 30 minutes or more seem to be shorter overall.

2.5

Import the General Social Survey data.

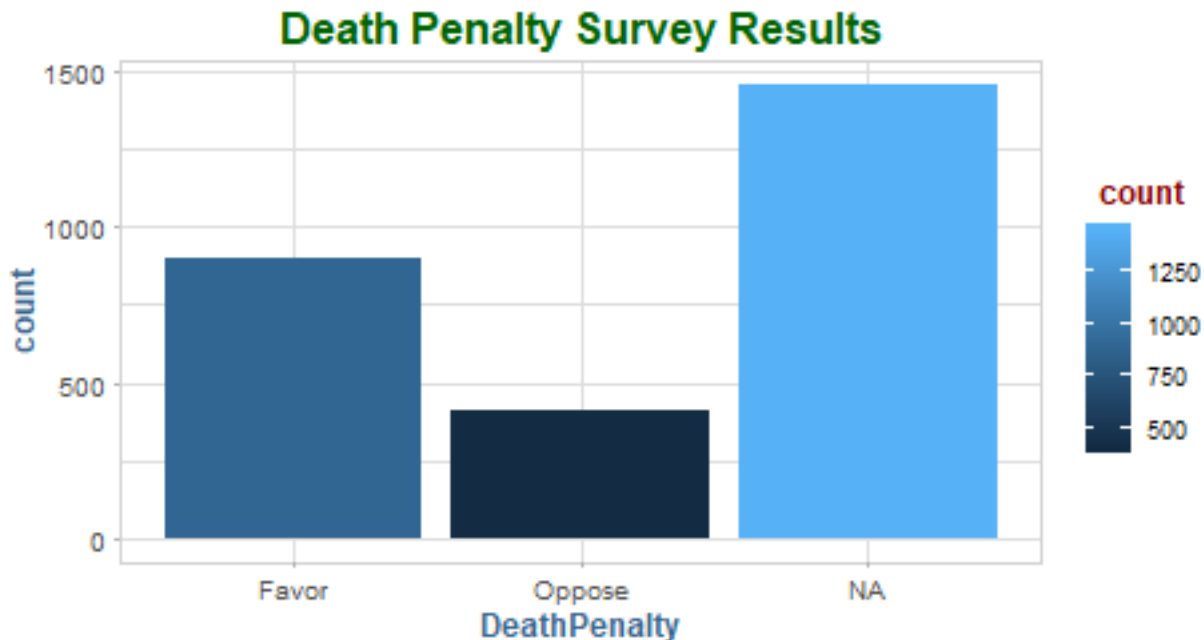
```
gss <- data.table(read.csv(paste0(data.dir, "GSS2002.csv"),
  header = T))
```

a.) Create a table and a bar chart of the response to the question about the death penalty.

```
table(gss$DeathPenalty)
```

Favor	Oppose
899	409

```
ggplot(gss, aes(DeathPenalty)) +
  geom_bar(aes(fill = ..count..)) +
  labs(title = "Death Penalty Survey Results")
```



b.) Use the `table` command and the summary command in R on the gun ownership variable. What additional information does the summary command give that the table does not?

```
table(gss$OwnGun)
```

No	Refused	Yes
605	9	310

```
summary(gss$OwnGun)
```

No	Refused	Yes	NA's
605	9	310	1841

The summary tells us how many people didn't respond at all to the question.

c.) Create a contingency table displaying the relationship between opinions about the death penalty to that about gun ownership.

```
with(gss, {
  table(OwnGun, DeathPenalty)
})
```

OwnGun	DeathPenalty	
	Favor	Oppose
No	375	199
Refused	7	2
Yes	243	59

d.) What proportion of gun owners favor the death penalty? Does it appear to be different from the proportion among those who do not own guns?

```
round(prop.table(with(gss, {
  table(OwnGun, DeathPenalty)
}))) * 100, 2)
```

OwnGun	DeathPenalty	
	Favor	Oppose
No	42.37	22.49
Refused	0.79	0.23
Yes	27.46	6.67

It does seem that gun owners are overwhelmingly in favor of the death penalty compared to those who do not own a gun.

2.6

Import the data from the recidivism case study in section 1.4.

Import the General Social Survey data.

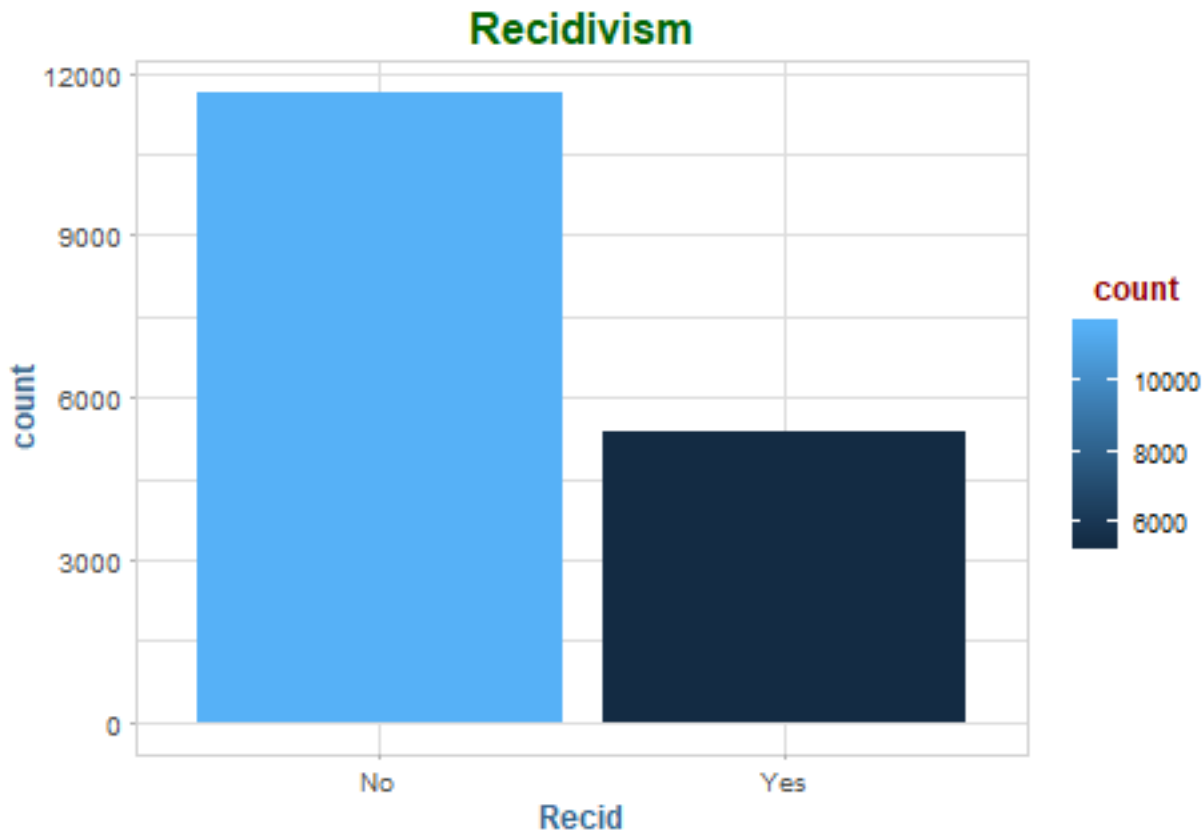
```
recid <- data.table(read.csv(paste0(data.dir, "Recidivism.csv"),
  header = T))
```

a.) Create a table and bar chart of the **Recid** variable.

```
table(recid$Recid)
```

No	Yes
11636	5386

```
ggplot(recid, aes(Recid)) +
  geom_bar(aes(fill = ..count..)) +
  labs(title = "Recidivism")
```



b.) Create a contingency table summarizing the relationship between recidivism (**Recid**) by age (**Age25**).

```
with(recid, {
  table(Recid, Age25)
})
```

	Age25	
Recid	Over 25	Under 25
No	9679	1954
Yes	4263	1123

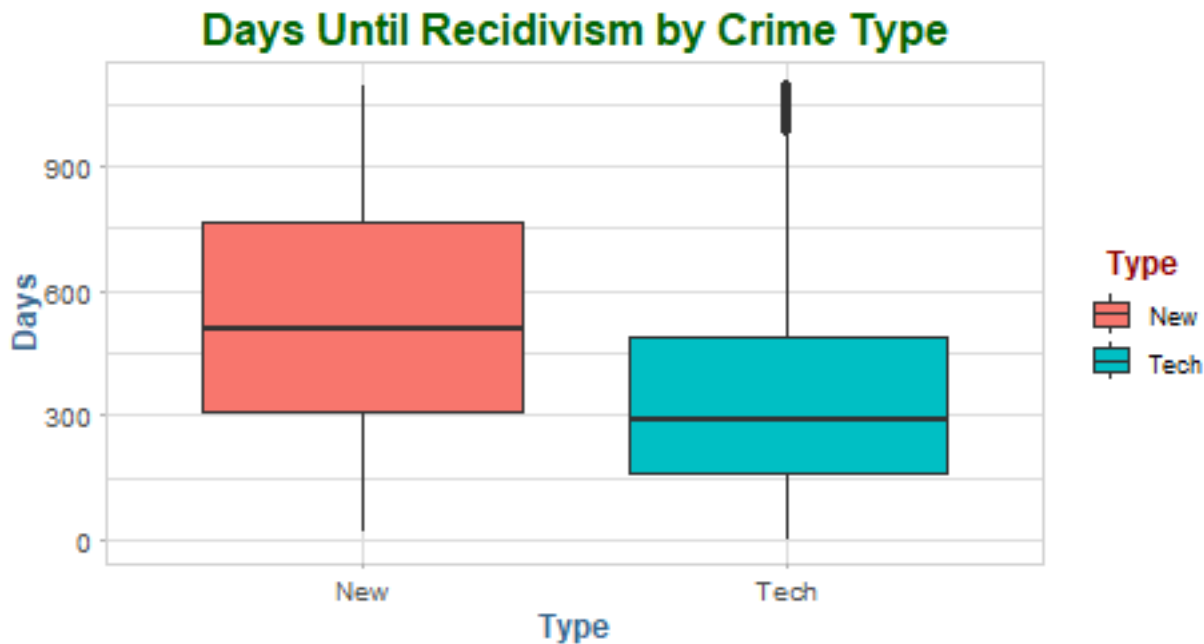
Of those over 25 years of age, what proportion were sent back to prison?

```
round(prop.table(with(recid, {
  table(Recid, Age25)
}))) * 100, 2)
```

	Age25	
Recid	Over 25	Under 25
No	56.87	11.48
Yes	25.05	6.60

c.) Create side-by-side boxplots of the number of days to recidivism grouped by type of violation, and give three comparative statements about the distributions.

```
ggplot(recid[!is.na(Days)], aes(Type, Days)) +
  geom_boxplot(aes(fill = Type)) +
  labs(title = "Days Until Recidivism by Crime Type")
```



- 1.) Technical violations seem to happen quicker after initial release than new crimes.
- 2.) There are more outliers present in the technical violations, with a cluster of them occurring after 900 days.
- 3.) The variance in days until recidivism is larger for new crimes. The chances of a technical violation occurring after 450 days drop dramatically.

```
recid[!is.na(Days), .(Variance = comma(var(Days))), by = Type]
```

```

Type  Variance
1: Tech 63,200.01
2: New 76,898.53
```

d.) Use the quantile command to obtain the quartiles of the number of days to recidivism. Since there are missing values (**NA**) for those released offenders who had not recidivated, you will need to add the argument **na.rm = T** to the **quantile** command to exclude those observations.

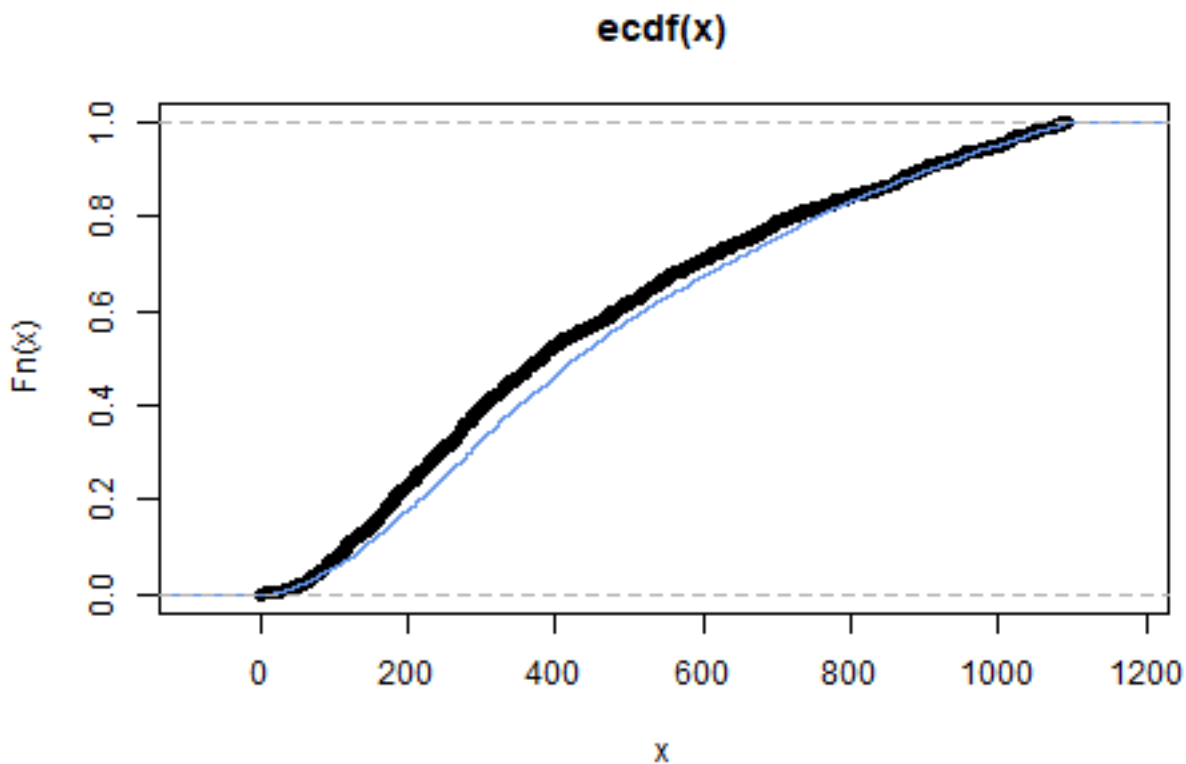

```
quantile(recid$Days, na.rm = T)
```

```
0%  25%  50%  75% 100%
0   241  418  687 1095
```

e.) Create ecdf's of days to recidivism for those under 25 years of age and those 25 years of age or older.

```
with(recid, {
  recid$U25 <- Age25 == "Under 25"

  plot.ecdf(recid[U25 == T]$Days)
  plot.ecdf(recid[U25 == F]$Days, col="cornflowerblue", add = T)
})
```



Approximately what proportion in each age group were sent back to prison 400 days after release?

Of those who were sent back to prison,

```
recid[!is.na(Days),
      .(Days, 0400 = Days >= 400), by = Age25][,
      .(Proportion = sum(0400) / .N), by = Age25]
```

```

      Age25 Proportion
1: Under 25  0.4746215
2: Over 25   0.5378841

```

went back after being release 400 days or more.

2.7

Import the data from the black spruce case study in 1.10.

```

spruce <- data.table(read.csv(paste0(data.dir, "Spruce.csv"),
                               header = T))

```

a.) Compute the numeric summaries for the hight changes of the seedlings.

```
summary(spruce$Ht.change)
```

```

Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 8.30  23.20   30.10   30.93  38.17   51.50

```

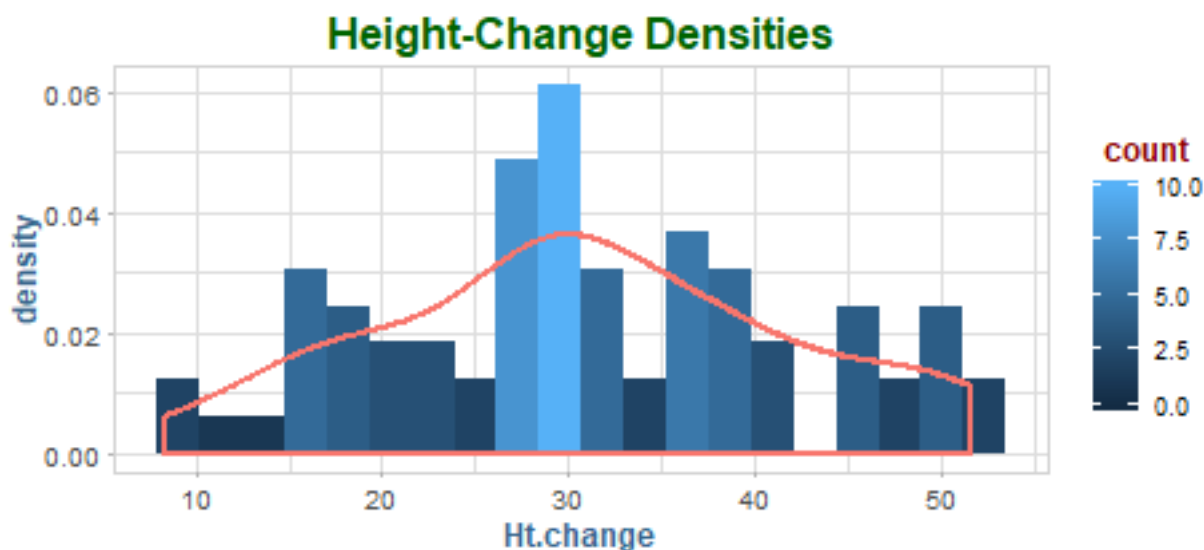
b.) Create a histogram and normal quantile plot for the height changes of the seedlings.

Is the distribution approximately normal?

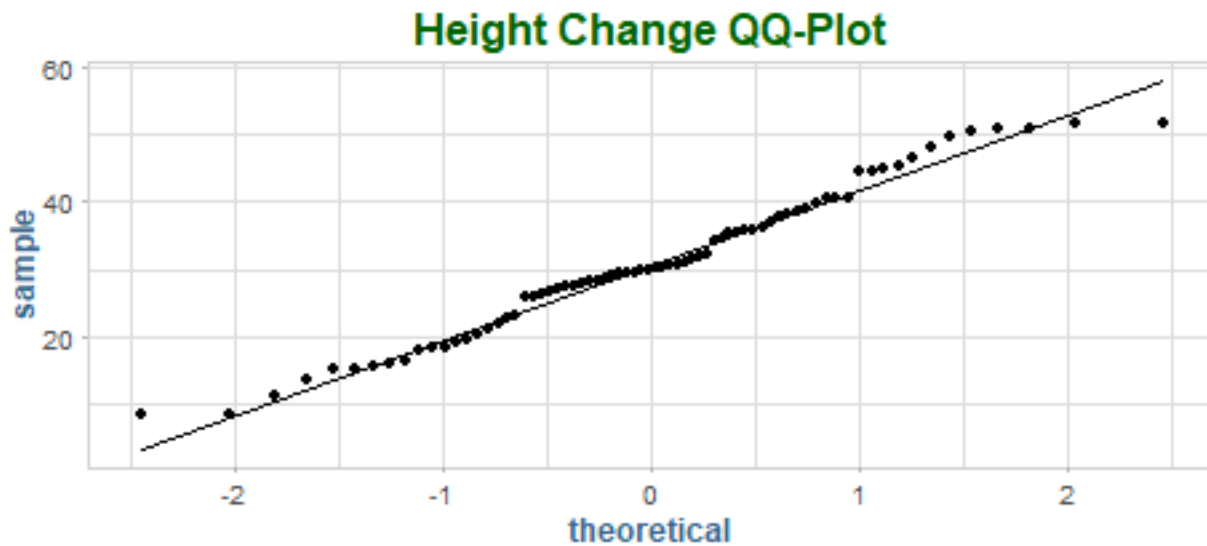
```

ggplot(spruce) +
  geom_histogram(aes(x = Ht.change, y = ..density.., fill = ..count..), bins = 20) +
  geom_density(aes(x = Ht.change, y = ..density.., col = "darkred"), lwd = 1) +
  guides(col = "none") +
  labs(title = "Height-Change Densities")

```



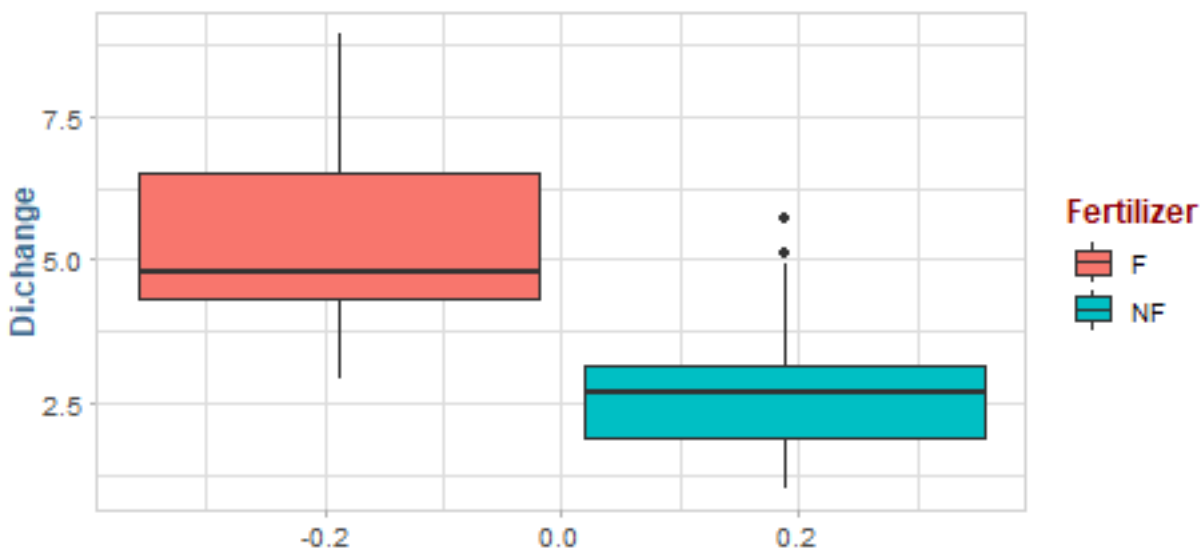
```
ggplot(spruce) +
  geom_qq(aes(sample = Ht.change)) +
  geom_qq_line(aes(sample = Ht.change)) +
  labs(title = "Height Change QQ-Plot")
```



For the height change variable, we see an uneven density plot, qq-plot with outliers in the tails. The data appears to be “approximately” normal.

c.) Create a boxplot to compare the distribution of the change in diameters of the seedlings (**Di.change**), grouped by whether or not they were in fertilized plots.

```
ggplot(spruce) +
  geom_boxplot(aes( y = Di.change, fill = Fertilizer ))
```



d.) Use the **tapply** command to find the numeric summaries of the diameter changes for the two levels of fertilization.

```
tapply(spruce$Di.change, spruce$Fertilizer, summary)
```

\$F

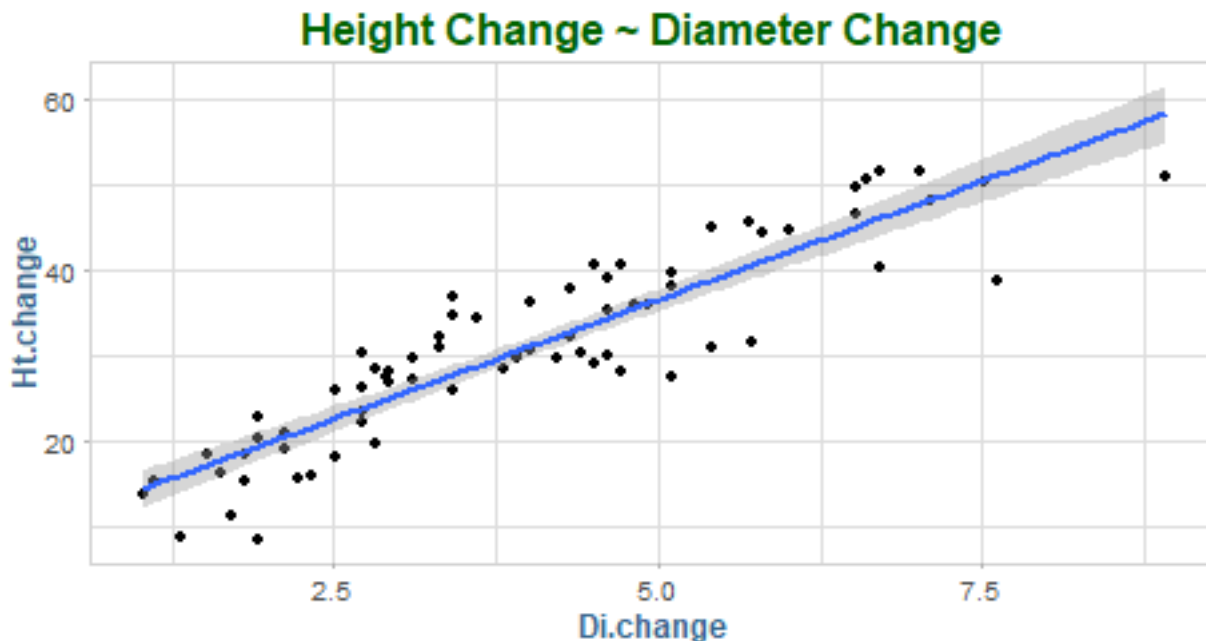
Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
2.913	4.318	4.763	5.274	6.518	8.919

\$NF

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
1.019	1.915	2.712	2.718	3.165	5.713

e.) Create a scatter plot of the hight change against the diameter changes, and describe the relationship.

```
ggplot(spruce, aes(Di.change, Ht.change)) +  
  geom_point() +  
  geom_smooth(method = "lm") +  
  labs(title = "Height Change ~ Diameter Change")
```



2.8

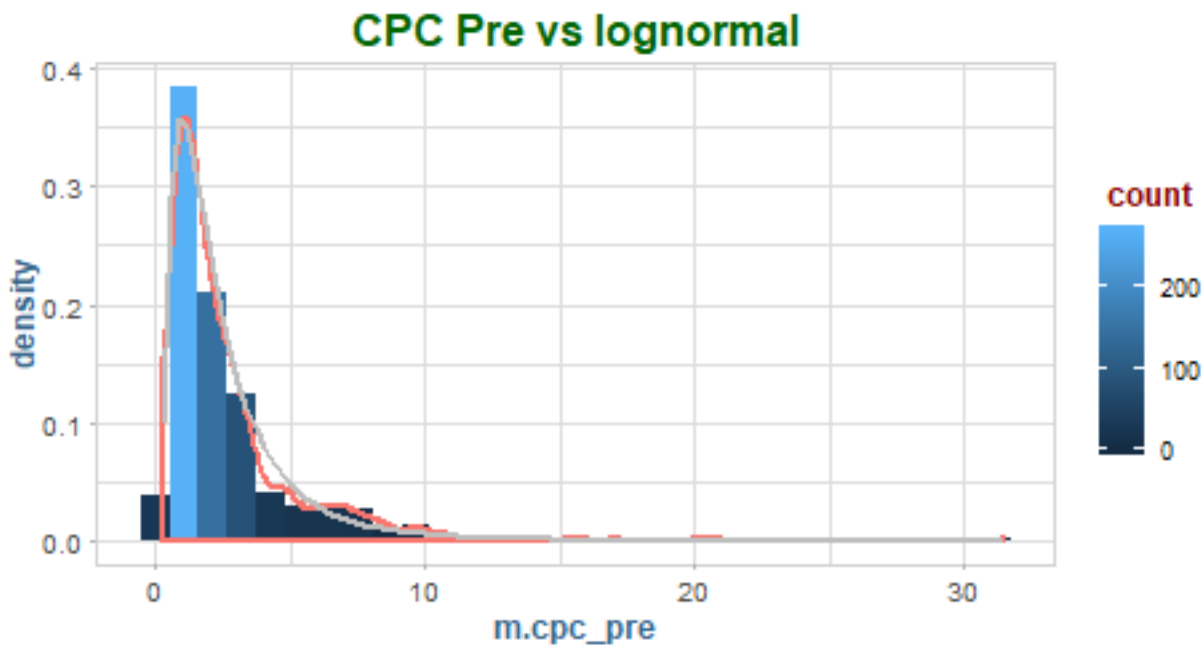
Import the mobile ads data from section 1.12.

```
mobile <- data.table(read.csv(paste0(data.dir, "MobileAds.csv"),
                               header = T))
```

a.) Create histograms of the variables **m.cpc_pre** and **m.cpc_post**, and describe their distributions.

```
fit <- fitdistr(mobile$m.cpc_pre, "lognormal")

ggplot(mobile, aes(m.cpc_pre)) +
  geom_histogram(aes(y = ..density.., fill = ..count..), bins = 30) +
  geom_density(aes(y = ..density.., col = "darkred"), lwd = 1) +
  stat_function(fun = dlnorm, size = 1, color = 'gray',
               args = list(mean = fit$estimate[1], sd = fit$estimate[2])) +
  guides(col = "none") +
  labs(title = "CPC Pre vs lognormal")
```



```
pct_zero <- sum(mobile$m.cpc_post == 0) / nrow(mobile) # 7% of values zero

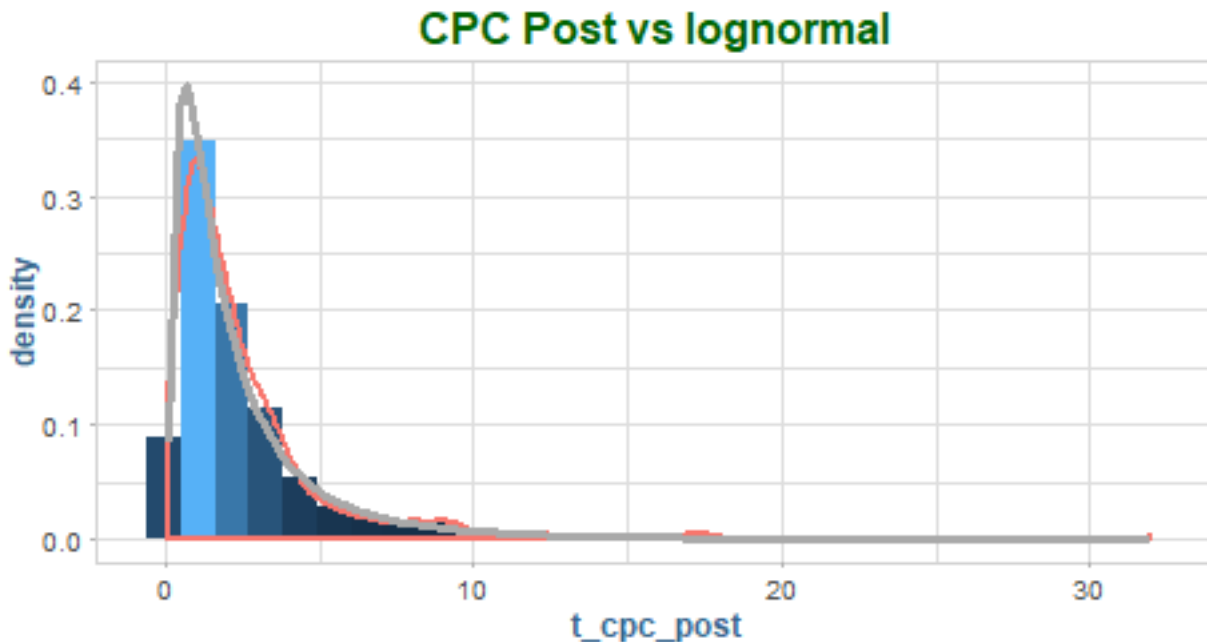
# we will replace the zeros with half the min value here to fit a lognormal.

t_cpc_post <- mobile$d.cpc_post
rep_val <- min( t_cpc_post[t_cpc_post > 0] ) / 2
```

```
t_cpc_post[t_cpc_post == 0] <- rep_val

fit <- fitdistr(t_cpc_post, "lognormal")

ggplot(mobile, aes(t_cpc_post)) +
  geom_histogram(aes(y = ..density.., fill = ..count..), bins = 30) +
  geom_density(aes(y = ..density.., col = "darkred"), lwd = 1) +
  stat_function(fun = dlnorm, size = 1.2, color = "darkgrey",
               args = list(mean = fit$estimate[1], sd = fit$estimate[2])) +
  guides(col = "none") +
  labs(title = "CPC Post vs lognormal") +
  theme(legend.position = "none")
```



The distribution of cpc pre and post seem to follow an approximately lognormal distribution.

b.) Compute the difference between these two variables, create a histogram, and describe the distribution.