PSTAT 105 - Assignment 2

Roshan Mehta

1 - Basketball B-Day Data

A well-known analysis in Malcolm Gladwell's book Outlier argues that the best hockey players are more likely to be born earlier in the year presumably because this gives them advantages in the youth hockey leagues. We are interested in checking whether there is a similar effect in basketball.

The data set Basketball_Ref_BDays.txt contains information for a large sample of professional basketball players listed on the http://www.basketball-reference.com website. Use the `table` or `dplyr::count` function to calculate how many players were born in each month. Draw an appropriate plot.

```
library(readr)
library(tidyr)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
```

a)

```
## Reading in the entire data file with some special
## formatting to interpret the birthday dates correctly
BasketBDays <- read_csv("https://github.com/roshanmehta12/pstat105-nonparametric-methods/raw/main/data_
                        col_types = cols(`Birth Date`= col_date(format = "%B %d %Y")))

## There are some players without birthdays, we should drop them
sum(is.na(BasketBDays$`Birth Date`))
```

```
## [1] 18
```

```r
BasketBDays <- BasketBDays %>% drop_na(`Birth Date`)

## format allows us to extract the month of their birth
BasketBDays$Month <- format(BasketBDays$`Birth Date`,format="%m")
BasketBDays$Year <- format(BasketBDays$`Birth Date`,format="%Y")
BasketBDays$Day <- format(BasketBDays$`Birth Date`,format="%d")

## count is a tidyverse function which tabulates the outcomes
BBall.table <- BasketBDays %>% count(Month)
```

Here is the data tabulated by month.

Month	Jan	Feb	Mar	Apr	May	Jun
Count	430	418	445	366	402	420
Month	Jul	Aug	Sep	Oct	Nov	Dec
Count	462	417	443	431	397	387

We can represent this data in a bar graph as in figure 1. The horizontal line represents the level if all the months had an equal number of births.

```r
ggplot(BBall.table, aes(x=Month, y=n)) +
  geom_col(fill="violet") +
  labs(y="Count") +
  geom_hline(yintercept = mean(BBall.table$n), size=1.4, color="red")
```

```
## Warning: Using ‘size‘ aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use ‘linewidth‘ instead.
```

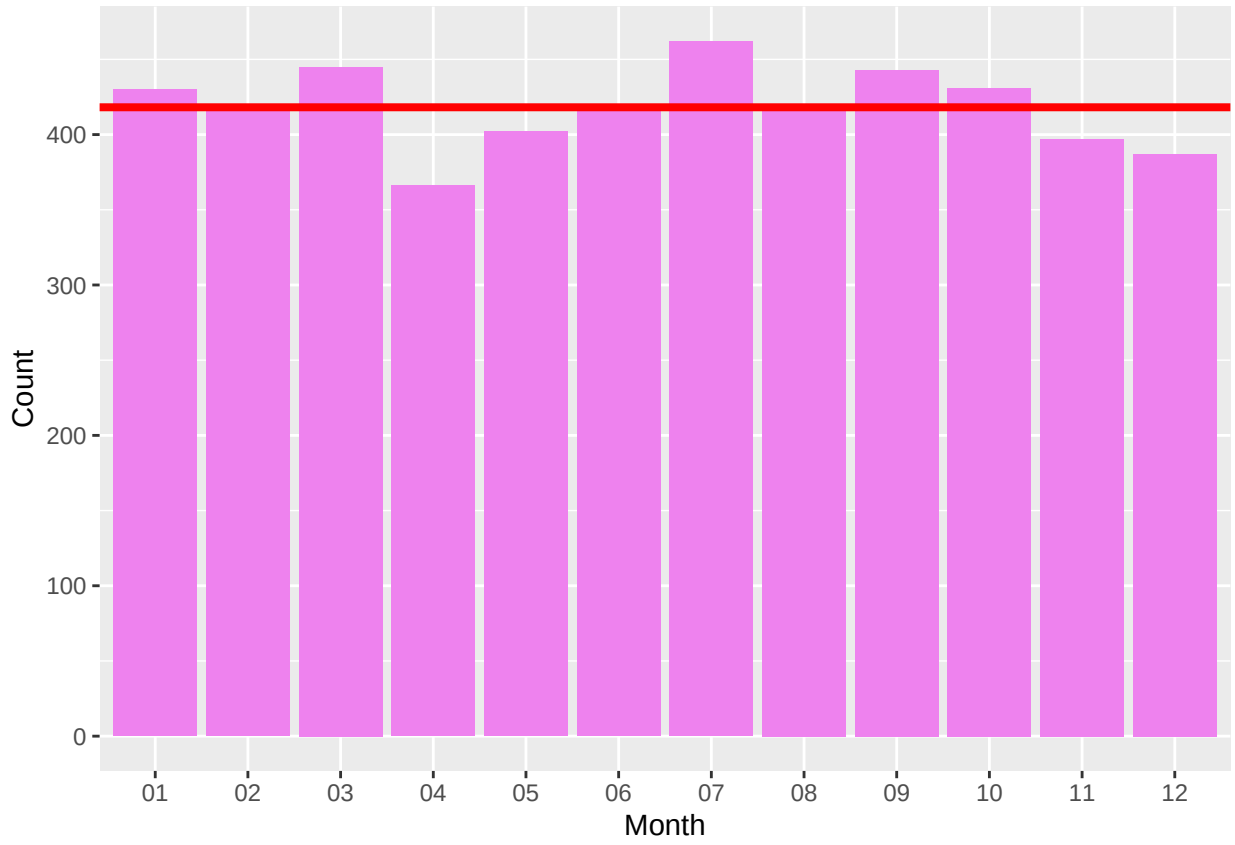

Figure 1: The number of basketball players born in each month. The red line indicates the average over all months.

b) Perform a χ^2 test to see if the players are equally likely to be born in any month.

```
expected <- sum(BBall.table$n[1:12])/12
X2 <- sum((BBall.table$n[1:12]-expected)^2/expected)
X2
```

```
## [1] 19.05859
```

```
1 - pchisq(X2,df=11)
```

```
## [1] 0.06004989
```

```
# or
chisq.test(BBall.table$n[1:12])
```

```
##
##  Chi-squared test for given probabilities
##
## data:  BBall.table$n[1:12]
## X-squared = 19.059, df = 11, p-value = 0.06005
```

Therefore, there is not a significant difference between the observed birthdays and just an even distribution across the months.

c) In order to focus our attention on modern players, repeat this analysis with only those players that were born after 1/1/1955. (also use this smaller data set for the following questions.)

```
BasketBDays$Year <- format(BasketBDays$`Birth Date`, format="%Y")
BBTableMnths.modern  <- filter(BasketBDays, Year >= 1955) %>% count(Month)
BBTableMnths.modern
```

```
## # A tibble: 12 x 2
##    Month     n
##    <chr> <int>
##  1 01      276
##  2 02      290
##  3 03      289
##  4 04      241
##  5 05      273
##  6 06      284
##  7 07      280
##  8 08      269
##  9 09      287
## 10 10      291
## 11 11      253
## 12 12      247
```

Then we can use this tabulation to produce the barplot in figure 2. The χ^2 test is produced just as it was before.

```
expct.n <- mean(BBTableMnths.modern$n)
X2 <- sum((BBTableMnths.modern$n-expct.n)^2/expct.n)
X2
```

```
## [1] 12.2878
```

```
1 - pchisq(X2, df=11)
```

```
## [1] 0.3424033
```

This statistic shows that there is still no evidence that players birthdays are clustered in any particular months.

```
ggplot(BBTableMnths.modern, aes(x=Month, y=n)) +
  geom_col(fill="violet") +
  labs(y="Count") +
  geom_hline(yintercept = mean(BBTableMnths.modern$n),
             size=1.4,
             color="red")
```

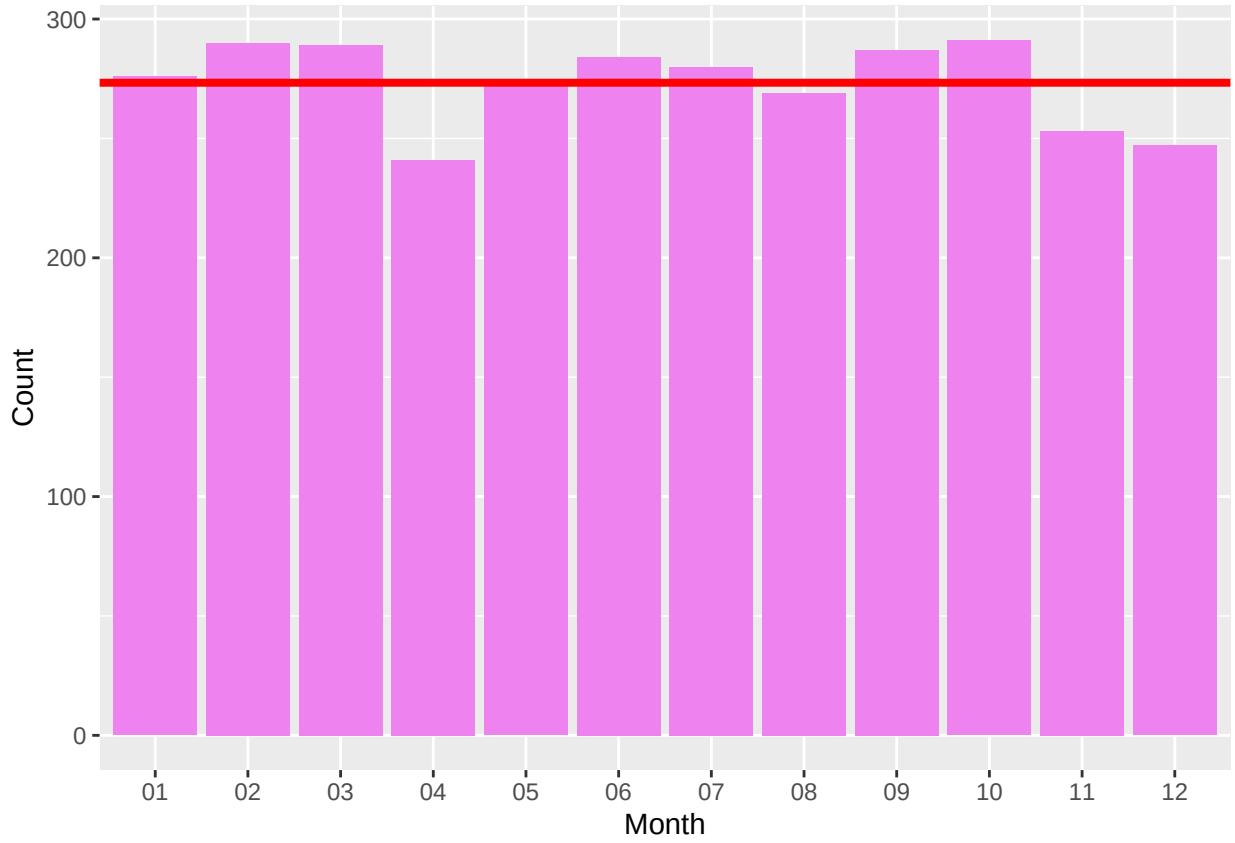

Figure 2: The number of players born in each month for players born after 1954. There does appear to be fewer than average birthdays in April, November and December.

d) To be more careful, we should realize that more people are probably born in January than February just because there are more days in January. Perform a χ^2 test where the null hypothesis is that the probability of each month is proportional to the average number of days in that month.

To be more careful, we should realize that more people are probably born in January than February just because there are more days in January. As a result, we should be testing the null hypothesis

Month	Jan	Feb	Mar	Apr	May	Jun
Days	31	28.25	31	30	31	30
Prob.	0.085	0.077	0.085	0.082	0.085	0.082
Month	Jul	Aug	Sep	Oct	Nov	Dec
Days	31	31	30	31	30	31
Prob.	0.085	0.085	0.082	0.085	0.082	0.085

Note how we have dealt with the issue of Leap Years (the players were born in months over many years so we expect about 1 in 4 years to include leap years. There were only 3 players in the modern data set, Chucky Brown, Vonteego Cummings, and Tyrese Haliburton born on Feb. 29.)

```
LeapYears <-  filter(BasketBDays, Month == "02" & Day == "29")
LeapYears$Player
```

```
## [1] "Chucky Brown\\brownch01"      "John Chaney\\chanejo01"
## [3] "Vonteego Cummings\\cummivo01" "Tyrese Haliburton\\halibty01"
```

Here is the calculation of the χ^2 statistic

```
p <- c(31,28.25,31,30,31,30,31,31,30,31,30,31)/365.25
chisq.test(BBTableMnths.modern$n, p=p)
```

```
##
##  Chi-squared test for given probabilities
##
## data:  BBTableMnths.modern$n
## X-squared = 16.096, df = 11, p-value = 0.1376
```

The data fits this new distribution worse than before, but there is not a statistically significant difference from the proportion of days. We should accept the null hypothesis in this case as well.

e) Going even further, it seems that some months generally are favored over others for having babies (summer births are more likely). We should probably compare our basketball player data to the following probabilities from the CDC.

Month	Jan	Feb	Mar	Apr	May	Jun
Prob.	0.0815	0.0752	0.0837	0.0816	0.0860	0.0813
Month	Jul	Aug	Sep	Oct	Nov	Dec
Prob.	0.0883	0.0892	0.0866	0.0849	0.0787	0.0830

This leads us to another version of the χ^2 test.

```
n <- sum(BBTableMnths.modern$n)
pc = c(0.0815, 0.0752, 0.0837, 0.0816, 0.0859, 0.0813,
       0.0883, 0.0892, 0.0866, 0.0849, 0.0787, 0.0831)
chisq.test(BBTableMnths.modern$n,p=pc)
```

```
##
##  Chi-squared test for given probabilities
##
## data:  BBTableMnths.modern$n
## X-squared = 18.027, df = 11, p-value = 0.08095
```

This test is once again further from our data, but not to a statistically significant degree.

f) Interpret your results. Is there significant evidence at an $\alpha = 0.05$ level that professional basketball players are born earlier in the year than the normal population?

This data does not show a statistically significant tendency for professional basketball players to be born earlier in the year.

```
expectc <- n*pc
ggplot(BBTableMnths.modern, aes(x=Month, y=n)) +
  geom_col(fill="violet") +
  labs(y="Count") +
  annotate(geom="point", x=1:12, y = expectc)
```

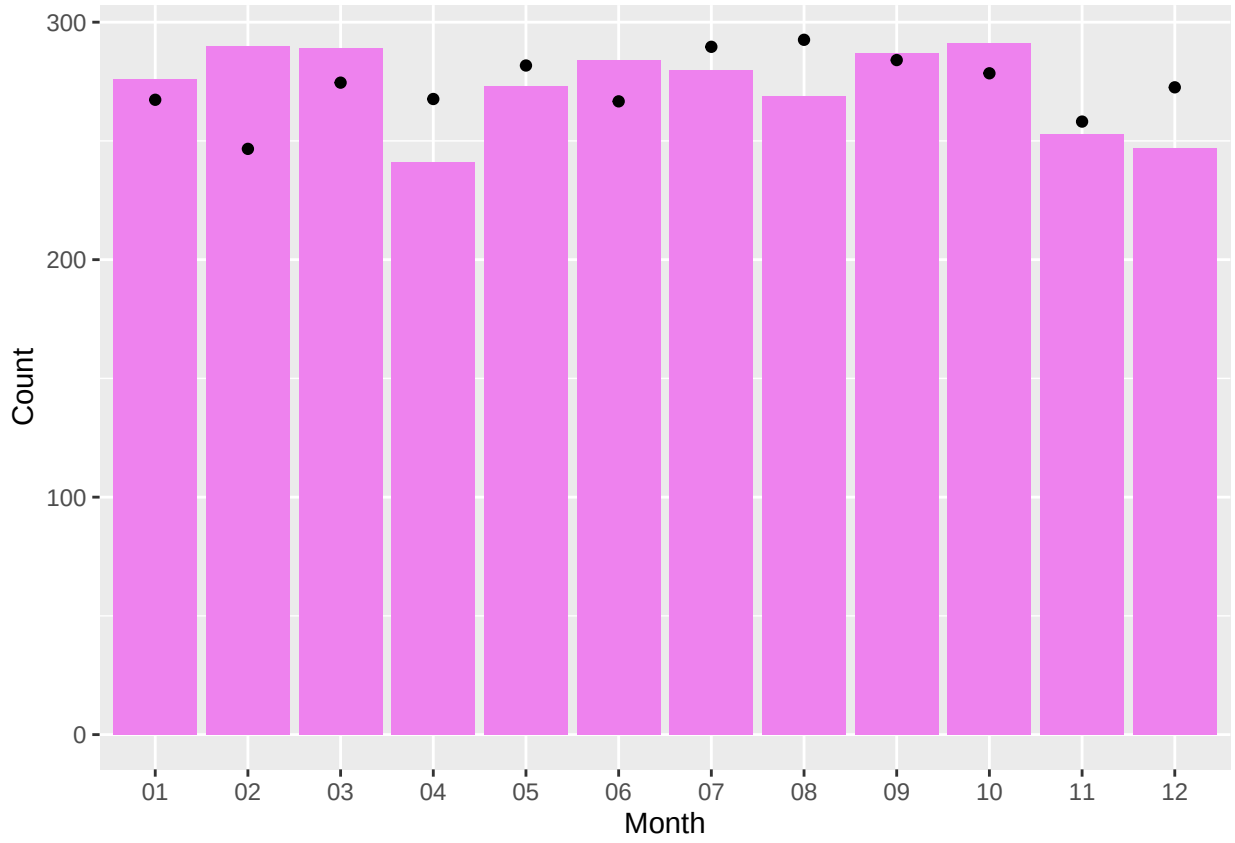

Figure 3: A comparison of the counts for each month versus the expected values from the CDC numbers represented by the dots.

Figure 3 shows a comparison from this last test between which months the players were born and which months the CDC reported as the general population proportions.From this plot, it seems that the main difference is that a lot more players are born in February and fewer are born in August and December. This lends some evidence to our theory that children born earlier in the year are more likely to become professional athletes. Though, the difference is not large enough to be significant.The first two tests are less significant probably because the effect of an advantage for players born earlier in the year is somewhat counteracted by families having fewer children in those winter months.

2 - Selling times data

The data set `Selltimes.txt` consists of the time that elapsed between when sell orders for CISCO stock were placed during April 5, 2010. My hypothesis is that these times have an exponential distribution with CDF $F(t) = 1 - e^{-\lambda t}$ for some unknown rate λ. (`R` has a function `pexp` that calculates these exponential probabilities.)

```
setwd("/Users/roshanmehta/Downloads/PSTAT/PSTAT 105/pstat105-nonparametric-methods/data_files")
selltimes <- scan("selltimes.txt")
```

a) Histogram of the data with appropriate number of bins.

```
hist(selltimes, breaks=300, xlab="Time",
     main="Histogram of Times between Sales", col="green")
```

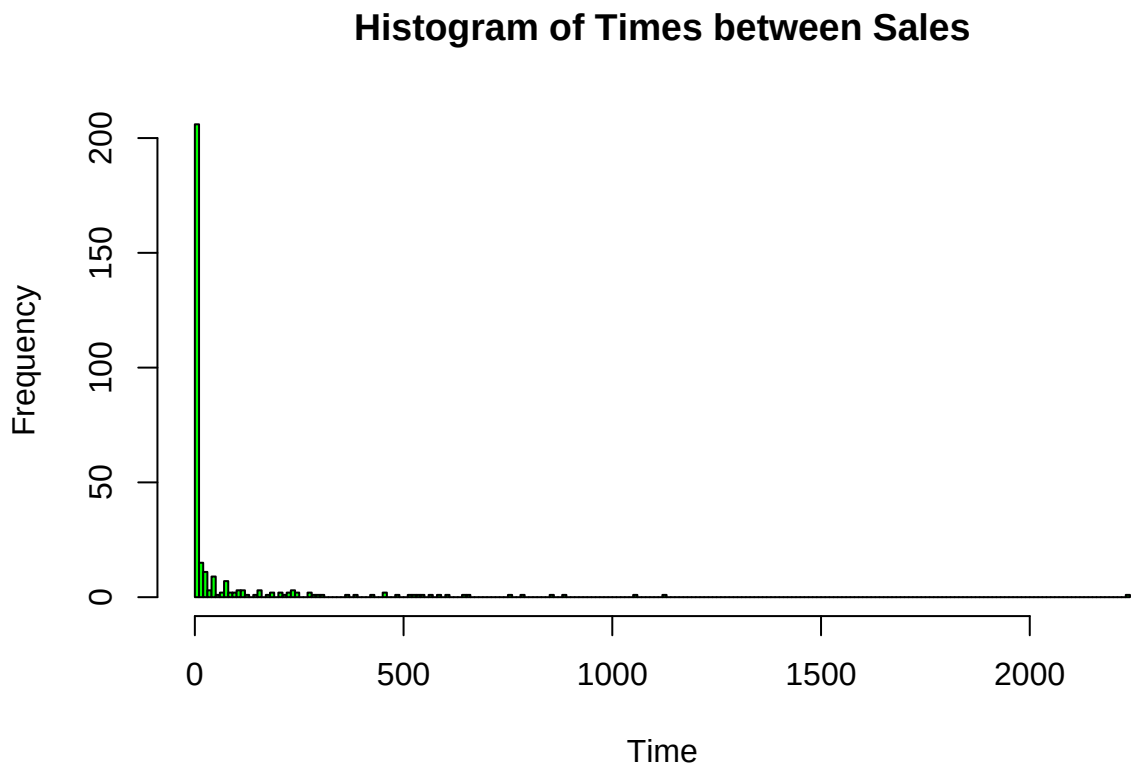

Figure 4: A histogram of all the selling times that shows a high degree of skewness in the data.

```
hist(selltimes[selltimes < 200], breaks=100, xlab="Time",
     main="Histogram of Times Less than 200", col="green")
```

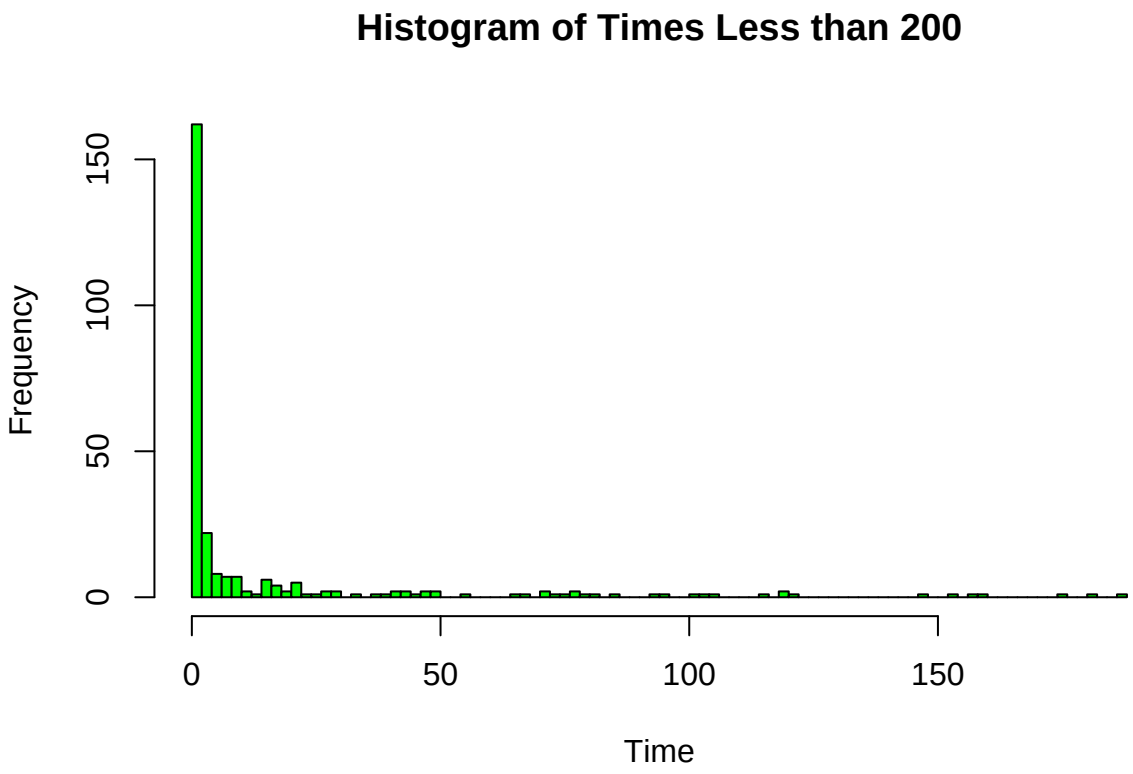

Figure 5: The time between sales for the 272 sales that happened within 200 seconds.

```
hist(selltimes[selltimes < 1], breaks=50, xlab="Time",
     main="Histogram of Times Less than 1 sec.", col="green")
```

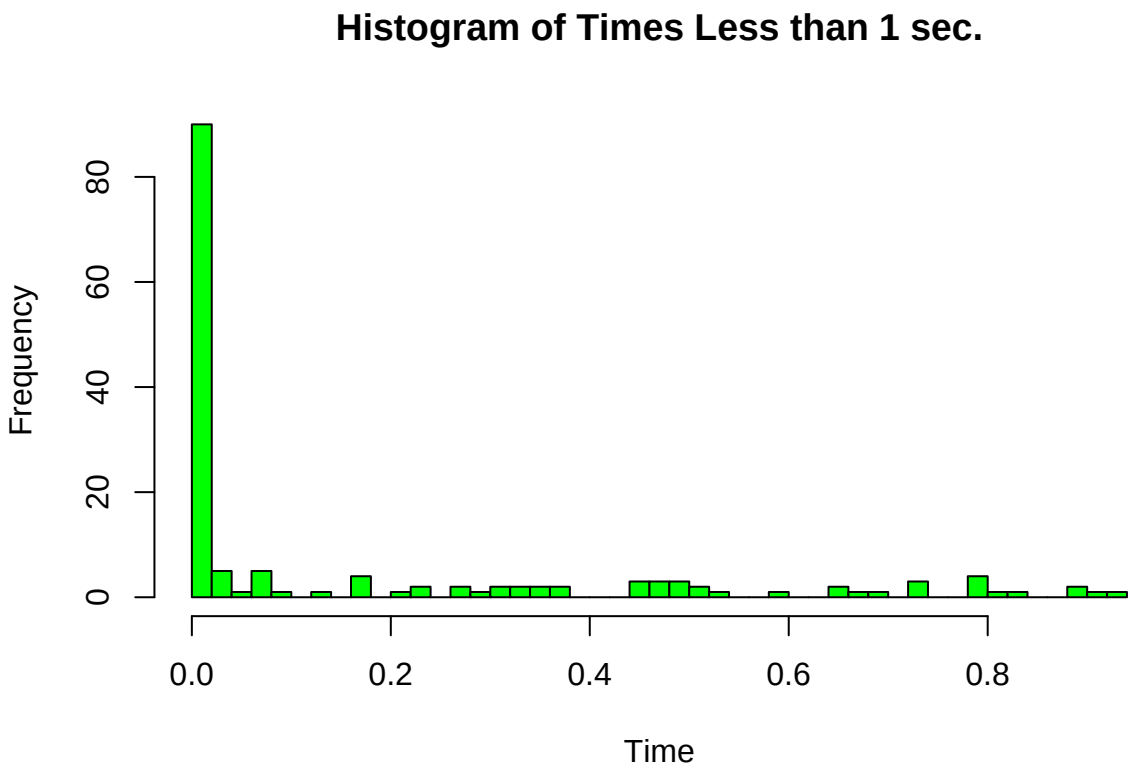

Figure 6: A histogram with 50 bins showing only the really short durations of less than 1 second.

We could produce a histogram of this data with 300 bins as in Figure 4, but what we notice is that there are a lot of observations near 0 and a few observations a long way out in the tails. We could try to get a better picture of the behavior near 0 by looking at only the shorter times. The histogram in Figure 5 shows only the times less than 200 seconds, and you can see that this is a little clearer. However, there are still a lot of observations near 0. This pattern persists if we look at really short times (less than 1 sec) in Figure 6 where by far the largest bin is still the very first.

b) Calculate the MLE, $\hat{\lambda} = \bar{x}^{-1}$ from the data.

```
hat.lambda <- 1/mean(selltimes)
round(hat.lambda,5)
```

```
## [1] 0.01321
```

c) Use this estimate of λ to divide the sample space into 10 intervals that will be big enough that the χ^2 approximation will be appropriate.

To choose the 10 intervals, we need to make sure that they have expectations greater than 5 under the null hypothesis that the distribution is exponential with parameter $\hat{\lambda}$= 0.01321. Bins with equal probability are often a good choice.

```
brks <- c(qexp(seq(0,.9,by=0.1),hat.lambda),2400)
cdf <- pexp(brks[c(-1,-11)],hat.lambda)
expected <- 309*( c(cdf,1) - c(0,cdf))
expected
```

```
##  [1] 30.9 30.9 30.9 30.9 30.9 30.9 30.9 30.9 30.9 30.9
```

d) Count the number of observations in each of those intervals. You can use the `hist`, `tabulate`, or `count` function.

```
b <- hist(selltimes, breaks = brks, plot=F)
round(b$breaks[c(-1, -11)], 3)
```

```
## [1]   7.978  16.897  27.008  38.681  52.487  69.384  91.168 121.870 174.357
```

```
b$counts
```

```
##  [1] 199  20  10   6   9   3   9   9   5  39
```

e) Perform the Chi-Square test.

```
x2 <- sum((b$counts - expected)^2/expected)
round(x2,2)
```

```
## [1] 1048.12
```

```
1 - pchisq(x2, 8)
```

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

This is significant. This data doesn't look much like an exponential, as was suggested by our plots.

f) Inspect the counts and the expected values and give some description of how the data looks different from an exponential distribution.

In order to see where the big difference was between the expectations and the observed counts,we can look at the components of the χ^2 statistic.

```
round((b$counts - expected)^2/expected, 2)
```

```
##  [1] 914.49   3.84  14.14  20.07  15.52  25.19  15.52  15.52  21.71   2.12
```

The main contribution was from the first interval which was much larger than expected. This indicates (as we observed in the histograms) that the selling times are much more heavily weighted around 0 than an exponential distribution. This sort of behavior is characteristic of process where many events happen close together followed by extended waits.

g) What difference does it make if we used 25 or 100 intervals instead of 10? Experiment a little with different sets of intervals and report the results and whether they demonstrate anything different from the original 10-interval analysis.

The analysis is probably more accurate with more bins, but in this case it does not look very much different.

```r
# for 25 bins
brks25 <- c(qexp(seq(0,0.96,by=1/25), hat.lambda), 2400)
b25 <- hist(selltimes, breaks=brks25, plot=F)
round(b25$breaks[c(-1,-26)], 1)
```

```
##  [1]   3.1   6.3   9.7  13.2  16.9  20.8  24.9  29.2  33.8  38.7  43.9  49.5
## [13]  55.6  62.2  69.4  77.4  86.3  96.4 108.1 121.9 138.8 160.6 191.3 243.7
```

```r
cdf <- pexp(b25$breaks[c(-1,-26)], hat.lambda)
expected25 <- 309*(c(cdf,1) - c(0, cdf))
expected25
```

```
##  [1] 12.36 12.36 12.36 12.36 12.36 12.36 12.36 12.36 12.36 12.36 12.36 12.36
## [13] 12.36 12.36 12.36 12.36 12.36 12.36 12.36 12.36 12.36 12.36 12.36 12.36
## [25] 12.36
```

```r
x2b25 <- sum((b25$counts-expected25)^2/expected25)
x2b25
```

```
## [1] 2253.54
```

```r
1 - pchisq(x2b25, 23)
```

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

```r
# for 100 bins
brks100 <- c(qexp(seq(0,0.99,by=1/100),hat.lambda),2400)
b100 <- hist(selltimes,breaks=brks100,plot=F)
cdf <- pexp(b100$breaks[c(-1,-101)],hat.lambda)
expected100 <- 309*( c(cdf,1) - c(0,cdf))
x2b100 <- sum((b100$counts-expected100)^2/expected100)
x2b100
```

```
## [1] 6518.508
```

```r
1 - pchisq(x2b100,98)
```

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

Unfortunately, for 100 intervals, the expected counts are no longer greater than 5 so the χ^2 approximation is not appropriate. Most likely, for this, the 25 bins would be the best choice. It has more power because of the finer level of detail in the intervals while still having enough observations in each interval to justify the normal approximation.