Final project 677_Illinois_precipitations

Jinyu

5/13/2022

#Introduction

To start with, I will solve the exercises mentioned in the book In All Likelihood

Answer the exercise 4.25

To solve this problem, we need to build a function of the order statistics

```
# reference: https://stackoverflow.com/questions/24211595/order-statistics-in-r?msclkid=fd6683dac56711e

# reference: https://www.statisticshowto.com/box-cox-transformation/#:~:text=What%20is%20a%20Box%20Cox%

f <- function(x, a=0, b=1) dunif(x, a,b) #pdf function
F <- function(x, a=0, b=1) punif(x, a,b, lower.tail=FALSE) #cdf function

# a function of distribution of the order statistics
integrand <- function(x,r,n) {
  x * (1 - F(x))^(r-1) * F(x)^(n-r) * f(x)
}
```

Then, we get the approximation function

```
#obtain the expectation
E <- function(r,n) {
  (1/beta(r,n-r+1)) * integrate(integrand,-Inf,Inf, r, n)$value
}

# approx function
median_aprrox<-function(k,n){
  m<-(k-1/3)/(n+1/3)
  return(m)
}
```

And I calculate the median when n=5 and when n=10 as well as their approximation

```
# get the value when n=5, i=3
print("Get the median value When n=5")
```

```
## [1] "Get the median value When n=5"
```

```
E(3,5)
```

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

```
median_aprrox(3,5)
```

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

```
# get the value when n=10, i=5.5
print("Get the median value When n=10")
```

```
## [1] "Get the median value When n=10"
```

```
(E(5,10) + E(6,10))/2
```

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

```
(median_aprrox(5,10) + median_aprrox(6,10))/2
```

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

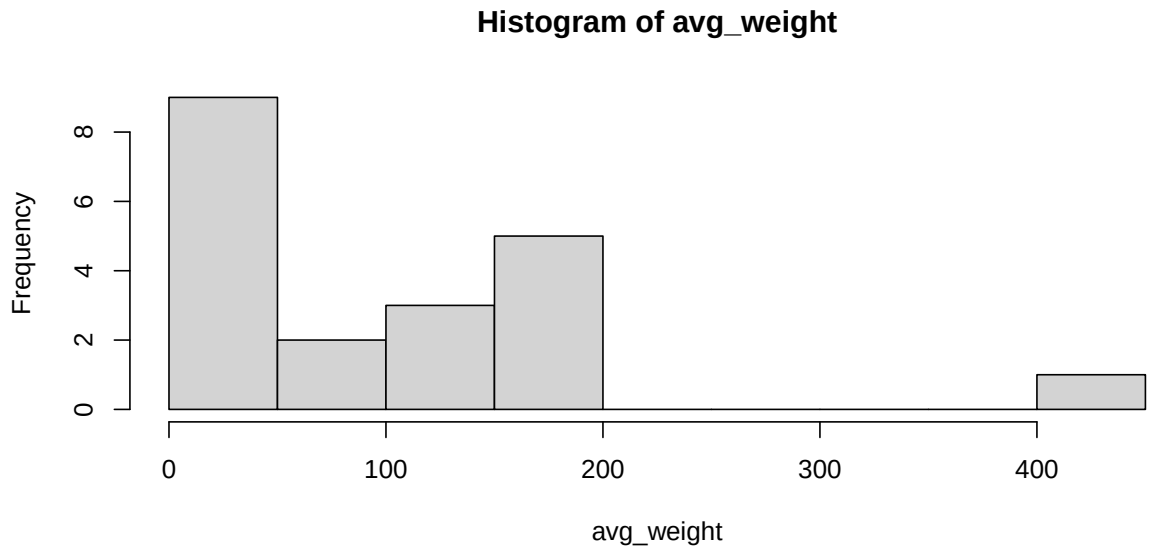
As we can see, the approximation for n=5 and n=10 are exactly the same.

Answer the exercise 4.39

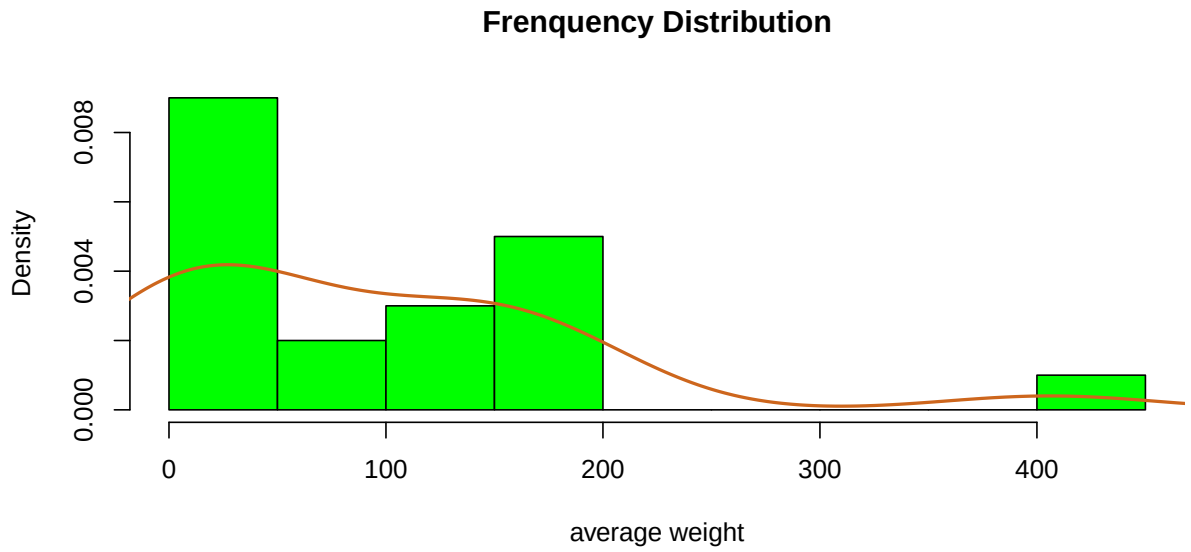
Now I try to answer the exercise 4.39

we first take a look at our data and describe them by histgram and density plot

```
avg_weight <-c(0.4,1.0,1.9,3.0,5.5,8.1,12.1,25.6,50.0,56.0,70.0,115.0,115.0,119.5,154.5,157.0,175.0,179
hist(avg_weight)
```



```r
hist(avg_weight,col="green",
     border="black",
     prob = TRUE,
     xlab = "average weight",
     main = "Frenquency Distribution")

lines(density(avg_weight),
      lwd = 2,
      col = "chocolate3")
```




```r
# density plot
# plot(density(avg_weight), frame = FALSE, col = "blue",main = "Density plot")
```

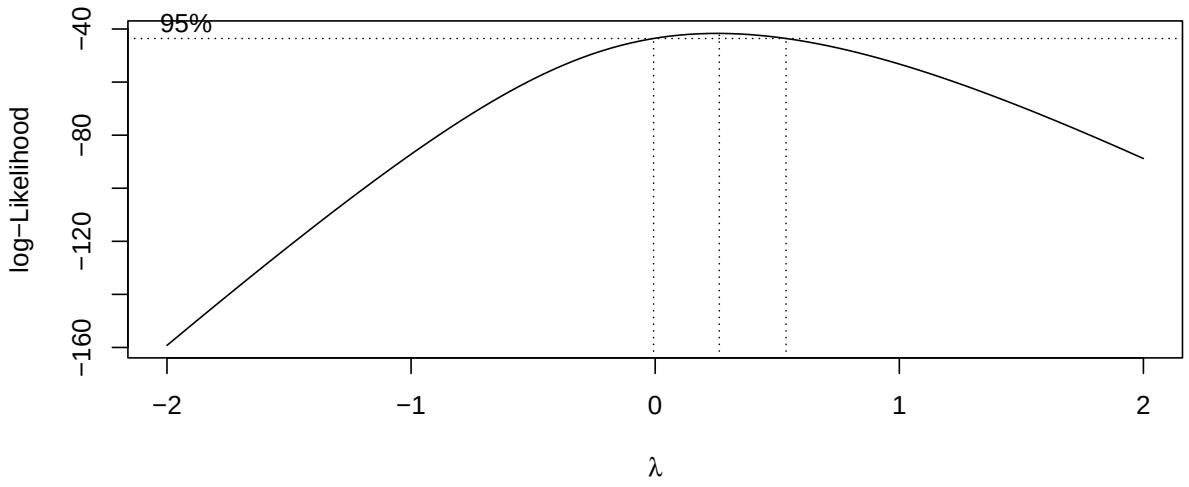
Now I applied the box-cox transformation to the data, which is to transform my non-normal dependent variables into normal shape.

Here is a box-cox formula for positive dataset, when given an input of Y:

$$y(\lambda) = \begin{cases} \frac{y^{\lambda}-1}{\lambda}, if \lambda \neq 0 \\ log(y), if \lambda = 0 \end{cases}$$

The λ is a parameter of the exponent, which varies from -5 to 5. All values of λ are considered and the optimal value for my data is selected

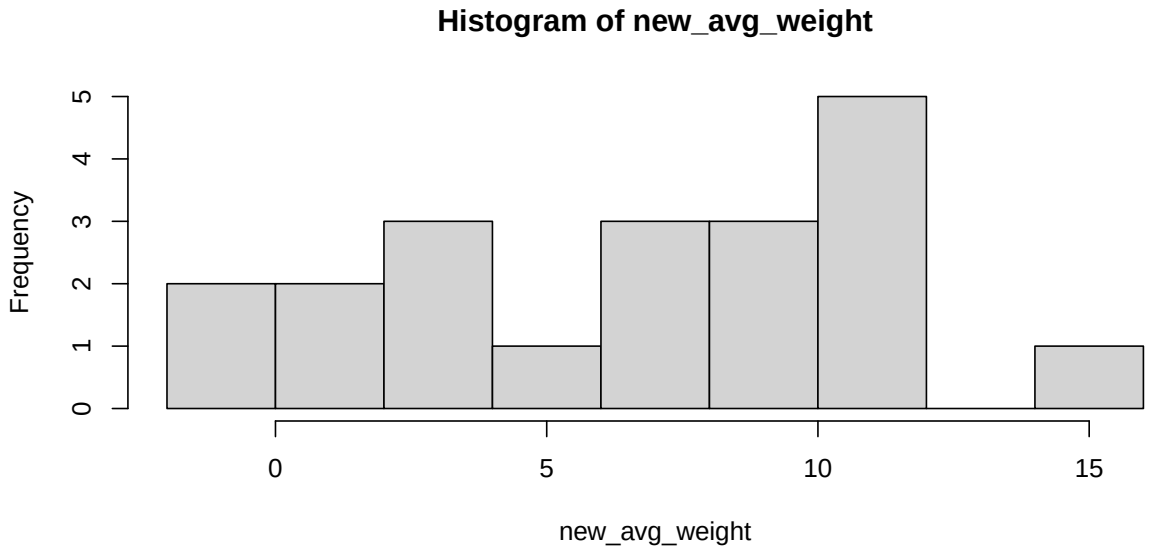
```r
# reference: https://r-coder.com/box-cox-transformation-r/

# Conduct boxcox transformation
box_weight <- boxcox(lm(avg_weight ~ 1))
```




```
# the optimal value of the lambda
lambda <- box_weight$x[which.max(box_weight$y)]
lambda #lambda=0.2626263
```

```
## [1] 0.2626263
```

```
new_avg_weight <- (avg_weight ^ lambda - 1) / lambda
hist(new_avg_weight)
```



So the transformation should be $y(0.2626263) = \frac{y^{0.2626263}-1}{0.2626263}$, where y is the data we input (avg_weight).

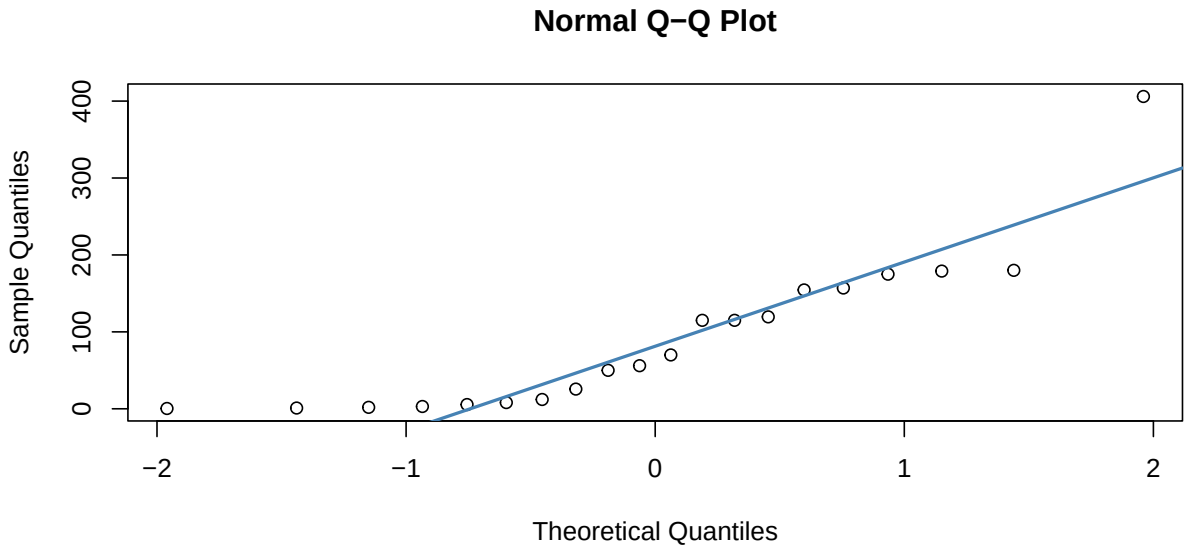
We can have a simple compare the original data and the data after the transformation based on the shapiro-wilk normality test or qqplot. Here is the test on the original data

```r
#Shapiro-Wilk normality test on original data
shapiro.test(avg_weight)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  avg_weight
## W = 0.81551, p-value = 0.001486
```

Here is the qqplot on the original data

```r
qqnorm(avg_weight)
qqline(avg_weight, col = "steelblue", lwd = 2)
```



Now the transformed data looks more like following a normal distribution, but we can also perform, for instance, a statistical test to check it, as the Shapiro-Wilk test:

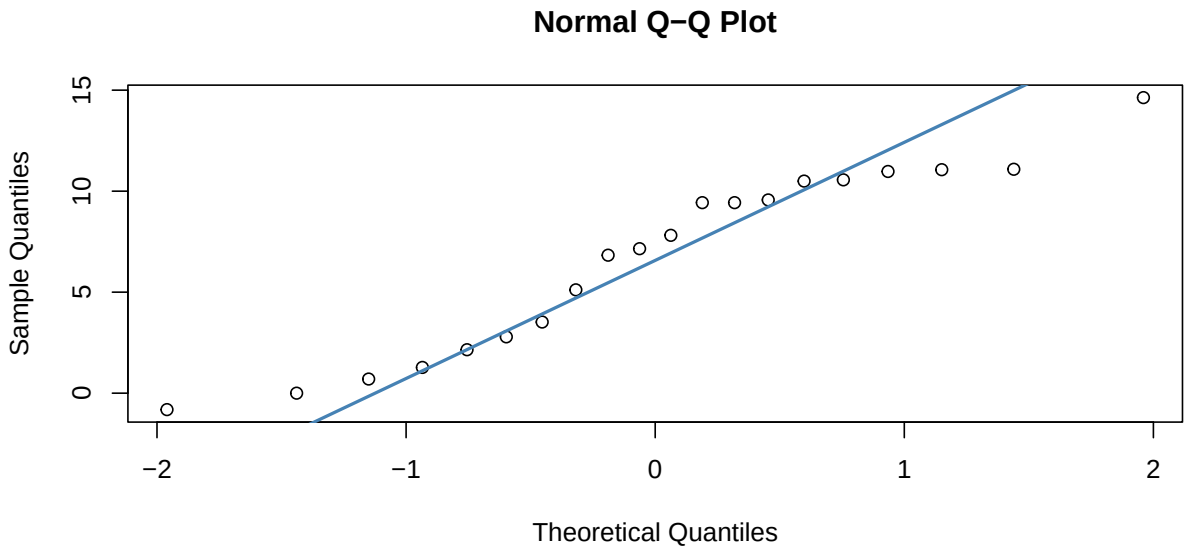
```r
#Shapiro-Wilk normality test on data after transformation
shapiro.test(new_avg_weight)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  new_avg_weight
## W = 0.93213, p-value = 0.1697
```

As the p-value is greater than the usual levels of significance (1%, 5% and 10%) we have no evidence to reject the null hypothesis of normality.

Here is the qqplot on the data after transformation:

```
qqnorm(new_avg_weight)
qqline(new_avg_weight, col = "steelblue", lwd = 2)
```



We can tell the data after the transformation

Answer the exercise 4.27

```
# save the rainfall data
Jan<-c(0.15,0.25,0.10,0.20,1.85,1.97,0.80,0.20,0.10,0.50,0.82,0.40,1.80,0.20,1.12,1.83,
       0.45,3.17,0.89,0.31,0.59,0.10,0.10,0.90,0.10,0.25,0.10,0.90)
Jul<-c(0.30,0.22,0.10,0.12,0.20,0.10,0.10,0.10,0.10,0.10,0.10,0.17,0.20,2.80,0.85,0.10,
       0.10,1.23,0.45,0.30,0.20,1.20,0.10,0.15,0.10,0.20,0.10,0.20,0.35,0.62,0.20,1.22,
       0.30,0.80,0.15,1.53,0.10,0.20,0.30,0.40,0.23,0.20,0.10,0.10,0.60,0.20,0.50,0.15,
      0.60,0.30,0.80,1.10,
      0.2,0.1,0.1,0.1,0.42,0.85,1.6,0.1,0.25,0.1,0.2,0.1)
```

(a)

```
summary(Jan)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.1000  0.1875  0.4250  0.7196  0.9000  3.1700
```

```
summary(Jul)
```

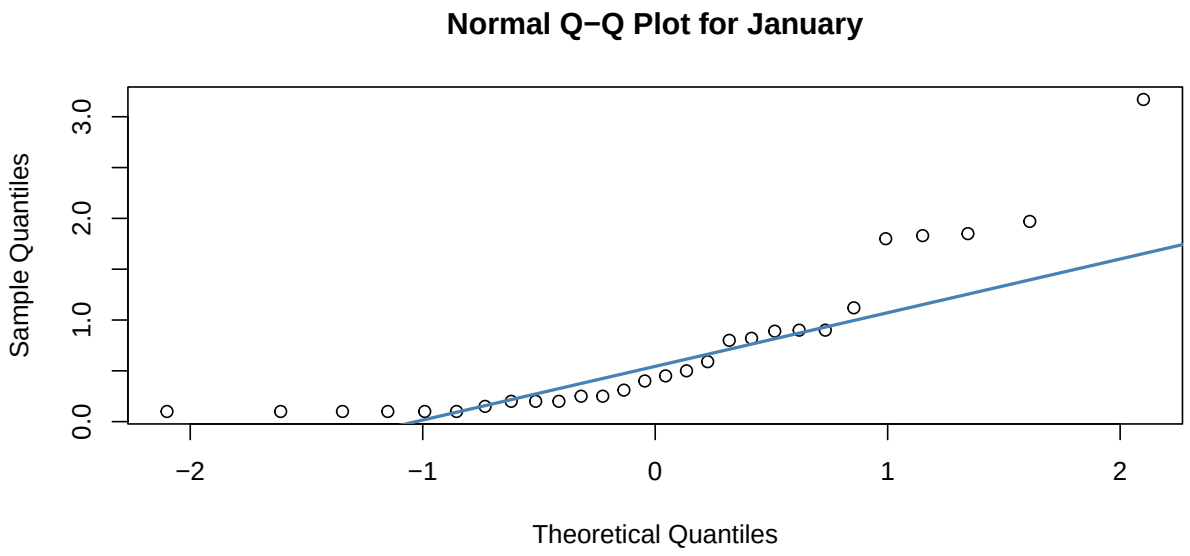
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.1000  0.1000  0.2000  0.3931  0.4275  2.8000
```

For the rainfall in January, the 1st, Median, Mean 3rd Max are larger than the ones in July. We can also tell that the IQR of the rainfall in January is larger than that in July.

(b)

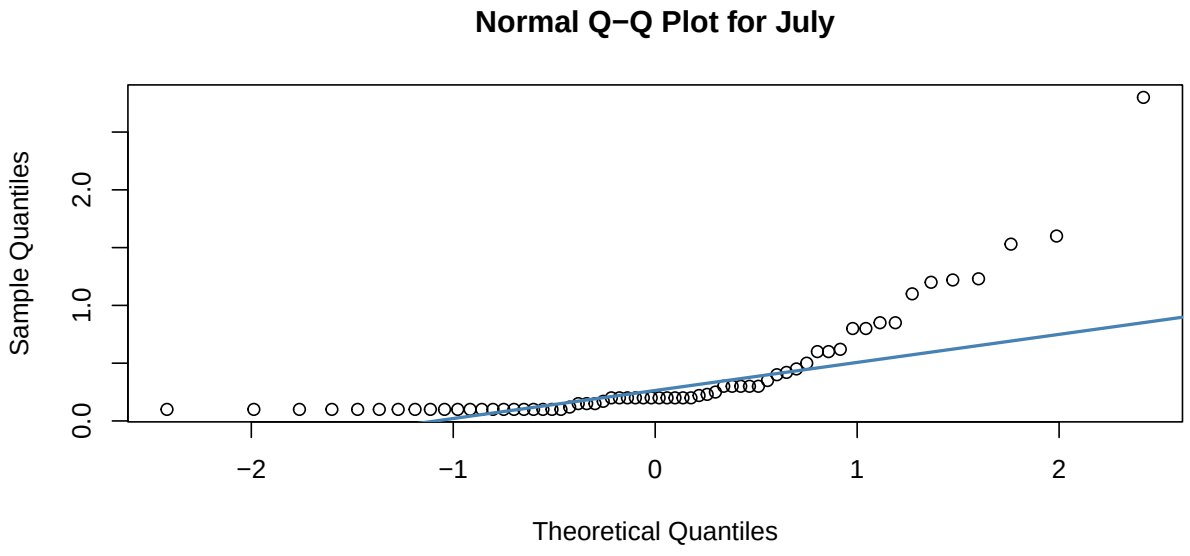
We looked at the QQ-plot of the data In January,

```
qqnorm(Jan, pch = 1, main = "Normal Q-Q Plot for January")
qqline(Jan, col = "steelblue", lwd = 2)
```

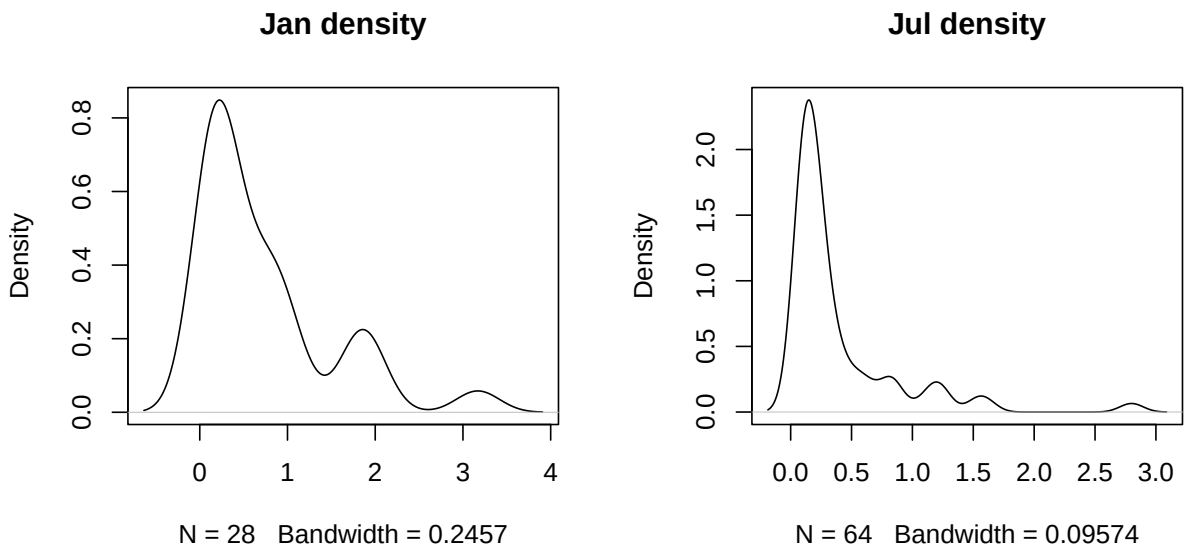


In July,

```
qqnorm(Jul, pch = 1, main = "Normal Q-Q Plot for July")
qqline(Jul, col = "steelblue", lwd = 2)
```



```r
# get the density plot
par(mfrow = c(1, 2))
plot(density(Jan),main='Jan density')
plot(density(Jul),main='Jul density')
```



As we can see in the qqplots, these 2 datasets don't look like following the normal distribution. According to the density plot, we can tell that they might follow gamma distributions, so we may use gamma distribution to fit the data.

(c)

There are many ways to solve the problem. I use the fitdist here, which is to fit a distribution based on the data, the distribution family given by ourselves, and the estimation given by ourselves:

```r
Jan.fit1=fitdist(Jan,'gamma','mle')
Jan.fit1
```

```
## Fitting of the distribution ' gamma ' by maximum likelihood
## Parameters:
##       estimate Std. Error
## shape 1.056222  0.2497495
## rate  1.467650  0.4396202
```

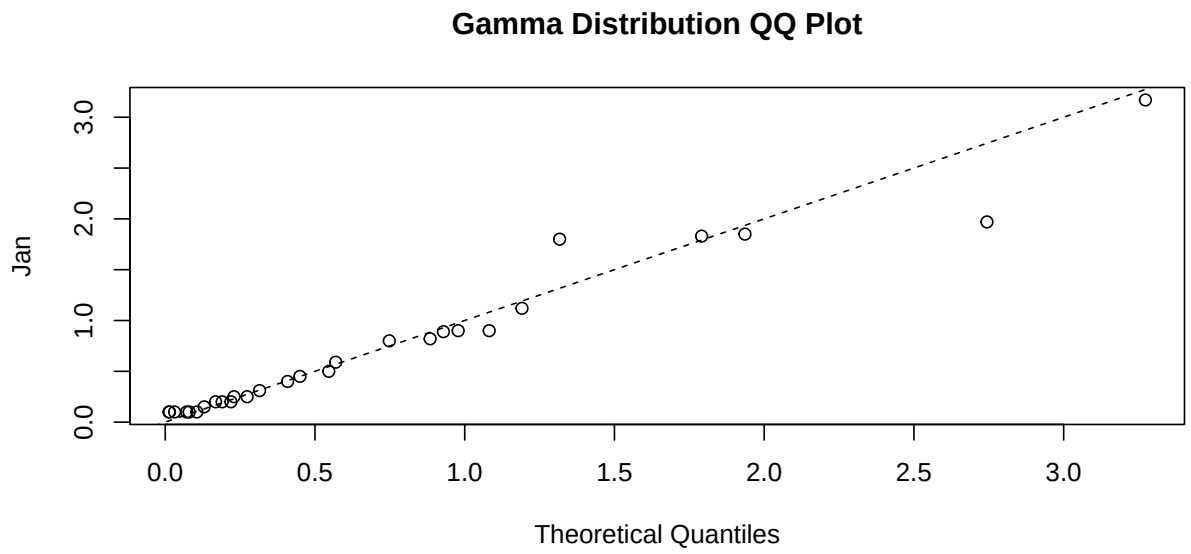
```r
Jul.fit1=fitdist(Jul,'gamma','mle')
Jul.fit1
```

```
## Fitting of the distribution ' gamma ' by maximum likelihood
## Parameters:
##       estimate Std. Error
## shape 1.196419  0.1891196
## rate  3.043403  0.5936302
```

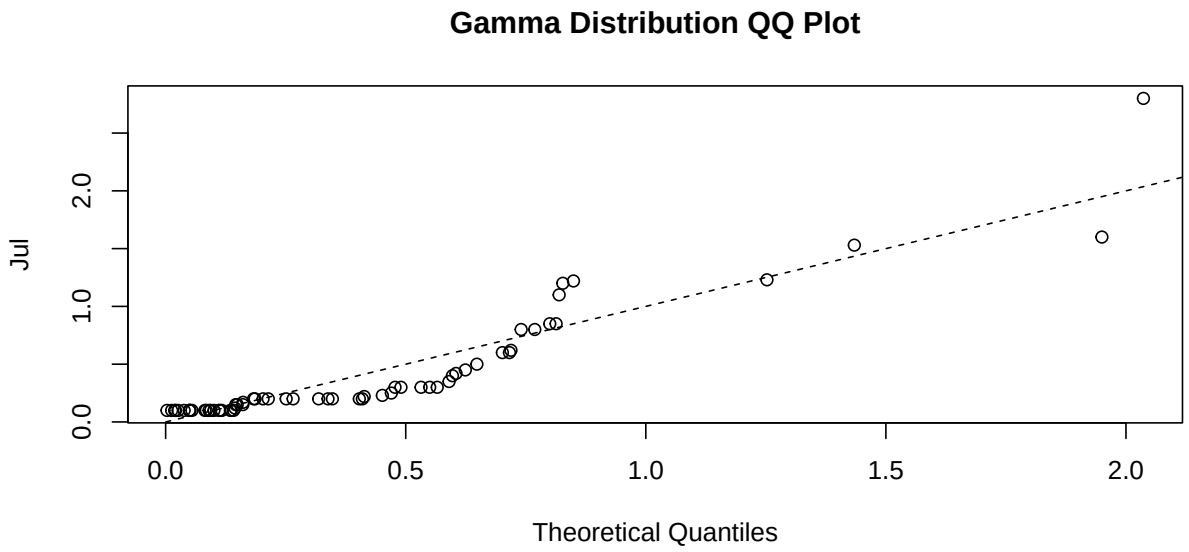
(d)

Now I conduct a qqgamma function to check if the data relatively follow gamma distribution.

```
# library(qpToolkit)
# qqGamma(resid(Jan.fit))
# reference:qpToolkit
# https://github.com/qPharmetra/qpToolkit/blob/master/R/qqGamma.r

qqGamma <- function(x
                    , ylab = deparse(substitute(x))
                    , xlab = "Theoretical Quantiles"
                    , main = "Gamma Distribution QQ Plot",...)
{
    # Plot qq-plot for gamma distributed variable

    xx = x[!is.na(x)]
    aa = (mean(xx))^2 / var(xx)
    ss = var(xx) / mean(xx)
    test = rgamma(length(xx), shape = aa, scale = ss)

    qqplot(test, xx, xlab = xlab, ylab = ylab, main = main,...)
    abline(0,1, lty = 2)
}


qqGamma(Jan)
```




```
qqGamma(Jul)
```



According to the plot, we assume that the data in July is better.

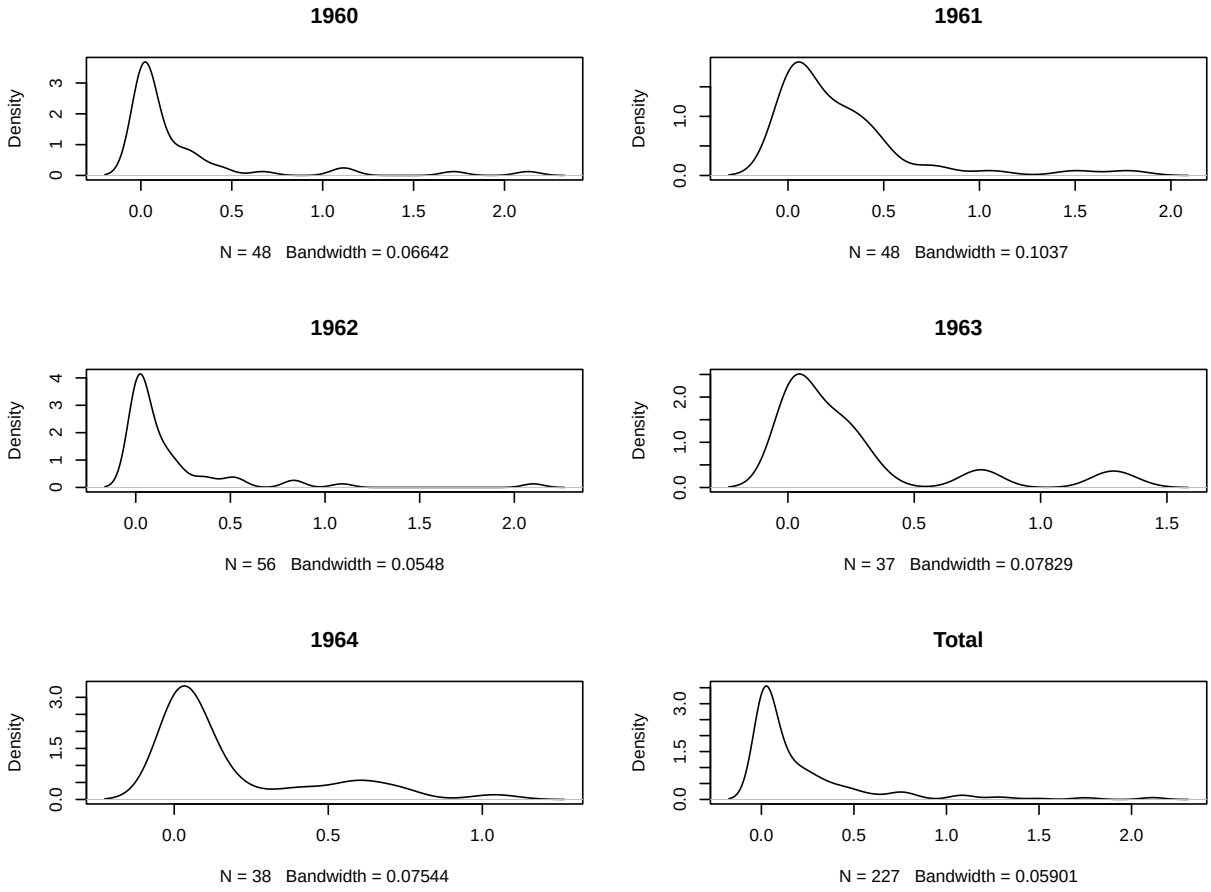
Illinois rain

Question 3

Use the data to identify the distribution of rainfall produced by the storms in southern Illinois. Estimate the parameters of the distribution using MLE. Prepare a discussion of your estimation, including how confident you are about your identification of the distribution and the accuracy of your parameter estimates.

I first investigate the density distribution of our data in these 5 year.

```
rain=read.xlsx('Illinois_rain_1960-1964.xlsx')
par(mfrow = c(3, 2))
density(rain$`1960` %>% na.omit()) %>% plot(main='1960')
density(rain$`1961` %>% na.omit()) %>% plot(main='1961')
density(rain$`1962` %>% na.omit()) %>% plot(main='1962')
density(rain$`1963` %>% na.omit()) %>% plot(main='1963')
density(rain$`1964` %>% na.omit()) %>% plot(main='1964')
density(unlist(rain) %>%  na.omit()) %>% plot(main='Total')
```



As we can see, it looks like all the data follow gamma distribution. So I decide to use gamma distribution fit the data.

To build a gamma distribution, I also need to estimate the parameters based on our sample data.

I conduct the fitdist based on the data of the whole 5 year. The MLS and MSE will be selected to decide which method is better.

```
fit_1<-fitdist(unlist(rain) %>%  na.omit() %>% c(),'gamma',method='mle') #MLE estimation
fit_2<-fitdist(unlist(rain) %>%  na.omit() %>% c(),'gamma',method='mse') #MSE estimation
```

```
summary(bootdist(fit_1)) #boot get confidence interval
summary(bootdist(fit_2)) #boot get confidence interval
```

I make a table of the summary of the MLE and MSE The median along with 95% confidence interval is shown in the table below. We can tell that the confidence interval of MLE has a smaller range, which I assume is more reliable.

Table 1: MLE fit of

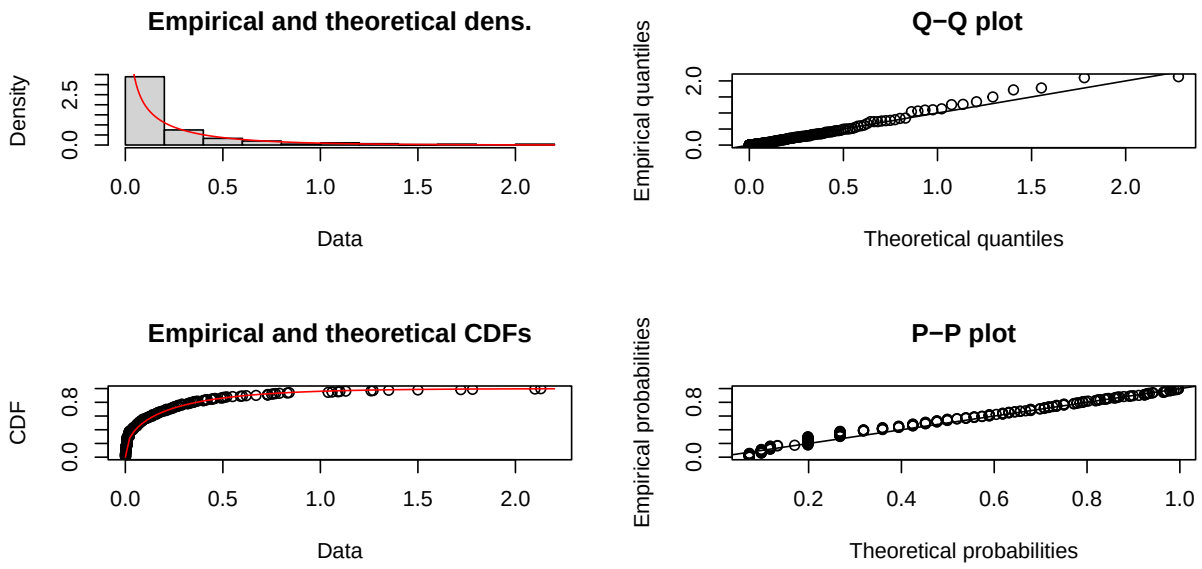
	Median	2.5%	97.5%
shape	0.4435628	0.3860337	0.5202142

	Median	2.5%	97.5%
rate	1.9995672	1.5649509	2.5637509

Table 2: MSE fit of Rain

	Median	2.5%	97.5%
shape	0.7172206	0.6158771	0.8358881
rate	1.3332668	1.0842435	1.6552038

```
plot(fit_1)
```



Question 2

Using this distribution, identify wet years and dry years. Are the wet years wet because there were more storms, because individual storms produced more rain, or for both of these reasons?

To answer this question, I made some descriptive statistics on the rainfall.

```
rain_mean=fit_1$estimate[1]/fit_1$estimate[2] #get mean for whole dataset
re=apply(rain,2,mean,na.rm =TRUE) # get mean for each year

out<-c(re,rain_mean %>% as.numeric() %>% round(4))
names(out)[6]='mean'
#out

num_storm<-c(nrow(rain)-apply(is.na(rain),2,sum),'/')

knitr::kable(rbind(out,num_storm)) # show the result
```

	1960	1961	1962	1963	1964	mean
out	0.220291666666667	0.2749375	0.18475	0.262432432432432	0.187105263157895	0.2244
num_storm	48	48	56	37	38	/

Based on the mean, 1962, 1964 are dryer years, 1961 and 1963 are wetter years. 1960 is the normal year. We can also conclude that more storms don't necessarily result in wet year and more rain in individual storm don't necessarily result in wet year. Another way to decide which year is wet year is to make a expert system, which is a threshold in this case. We can get the percentage of days beyond the threshold to rank those year and decide which year should be wet year.

As a result, there are many components influence our decisions.

	1960	1961	1962	1963	1964
mean	0.22032	0.27494	0.18475	0.26245	0.18713
num_storm	48	48	56	37	38

The result above shows the mean and number of storms, which is made by fitdist considering all the data in each year follow gamma distribution seperately.

Question 3

To what extent do you believe the results of your analysis are generalizable? What do you think the next steps would be after the analysis? An article by Floyd Huff, one of the authors of the 1967 report is included.

Data is not enough, which is to say that 5 years of observations are not enough for the distribution verification. We can try to enlarge the data based on some API online or applied bootstrap or MCMN. And Huff's article focused on description statistics. We still need a reliable data to make further analysis.