Final Assignment-In All Likelihood

Tao He

5/11/2022

First Part

Answers to exercises 4.25, 4.27, and 4.39.

Exercise 4.25

```
#' Gets order statistics from a 0-1 uniform distribution
#' @import stats
#' @param draw_size - The size of the output sample
#' @param k - The Kth smallest value from a sample
#' @param n - The size the sample to compute the order statistic from
#' @return A vector of random order statistic variables from a 0-1 uniform distribution
#' @references
#'  Gentle, James E. (2009), Computational Statistics, Springer, p. 63, ISBN 9780387981444
#' @export
order_probs <- function(draw_size, k, n) {
  return(rbeta(draw_size, k, n + 1 - k))
}

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
#distribution of the order statistics
integrand <- function(x,r,n) {
  x * (1 - F(x))^(r-1) * F(x)^(n-r) * f(x)
}
#get expectation
E <- function(r,n) {
  (1/beta(r,n-r+1)) * integrate(integrand,-Inf,Inf, r, n)$value
}
# approx function
medianprrox<-function(k,n){
  m<-(k-1/3)/(n+1/3)
  return(m)
}
E(2.5,5)
```

```
## [1] 0.4166667
```

```
medianprrox(2.5,5)
```

```
## [1] 0.40625
```

```
E(5,10)
```

```
## [1] 0.4545455
```

```
medianprrox(5,10)
```

```
## [1] 0.4516129
```

The two results are the same.

Exercise 4.27

```
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) Compare the summary statistics for the two months.

```
summary(Jan)
```

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

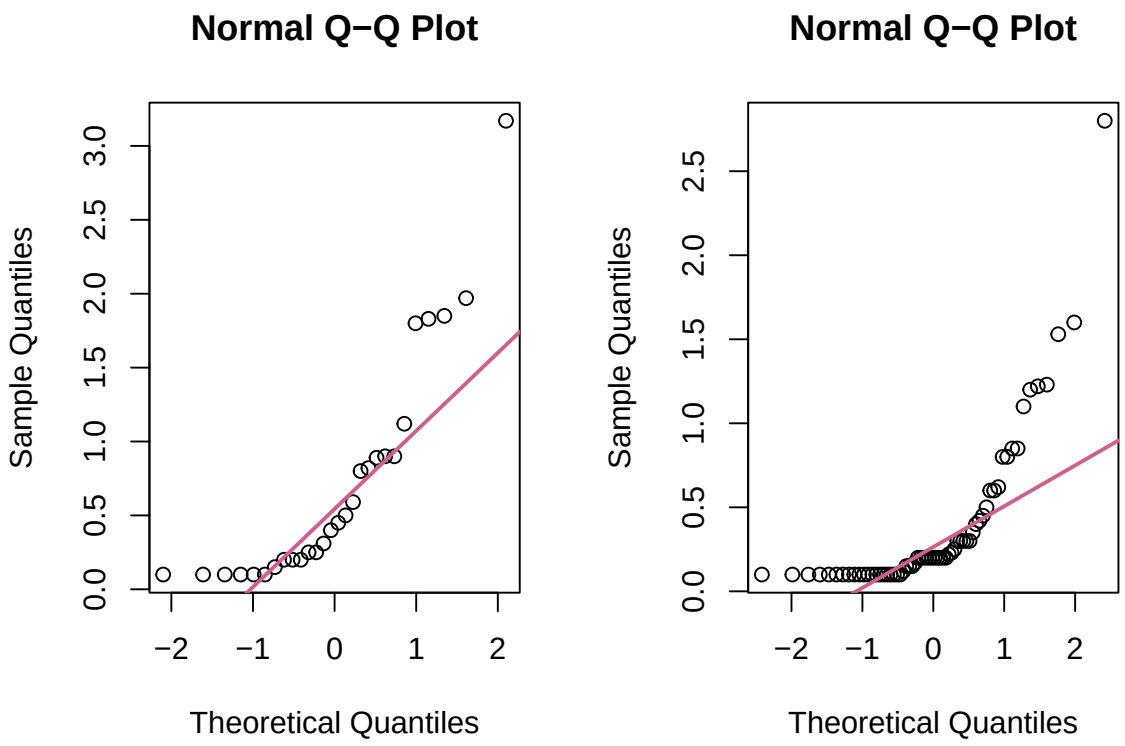
```
summary(Jul)
```

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

Except the minimum, the 1st Qu., median, mean, 3rd Qu., and maximum of January are higher than July.

(b) Look at the QQ-plot of the data and, based on the shape, suggest what model is reasonable.

```
par(mfrow = c(1, 2))
qqnorm(Jan, pch = 1)
qqline(Jan, col = "hotpink3", lwd = 2)
qqnorm(Jul, pch = 1)
qqline(Jul, col = "hotpink3", lwd = 2)
```



Since the observations are far from the normal distribution line, the sample does not follow the normal distribution. In addition, the observations are continuously distributed, and the gamma distribution can be considered.

(c) Fit a gamma model to the data from each month. Report the MLEs and standard errors, and draw the profile likelihoods for the mean parameters. Compare the parameters from the two months.

```
# gamma model
fit_Jan <- fitdist(Jan, distr = "gamma", method = "mle")
fit_Jul <- fitdist(Jul, distr = "gamma", method = "mle")
summary(fit_Jan)
```

```
## Fitting of the distribution ' gamma ' by maximum likelihood
## Parameters :
##       estimate Std. Error
## shape 1.056222  0.2497495
## rate  1.467650  0.4396202
## Loglikelihood:  -18.7616   AIC:  41.5232   BIC:  44.18761
## Correlation matrix:
##           shape      rate
## shape 1.0000000 0.7893943
## rate  0.7893943 1.0000000
```

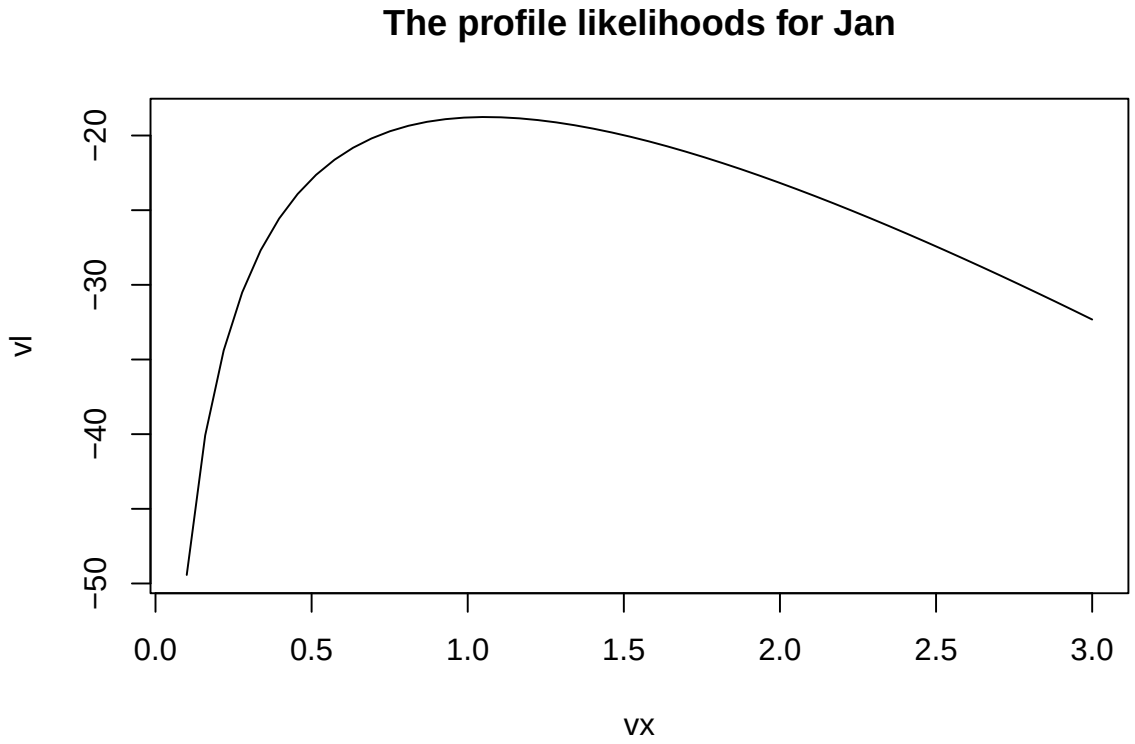
```
summary(fit_Jul)
```

```
## Fitting of the distribution ' gamma ' by maximum likelihood
## Parameters :
##       estimate Std. Error
## shape 1.196419  0.1891196
## rate  3.043403  0.5936302
## Loglikelihood:  -3.634886   AIC:  11.26977   BIC:  15.58754
## Correlation matrix:
```

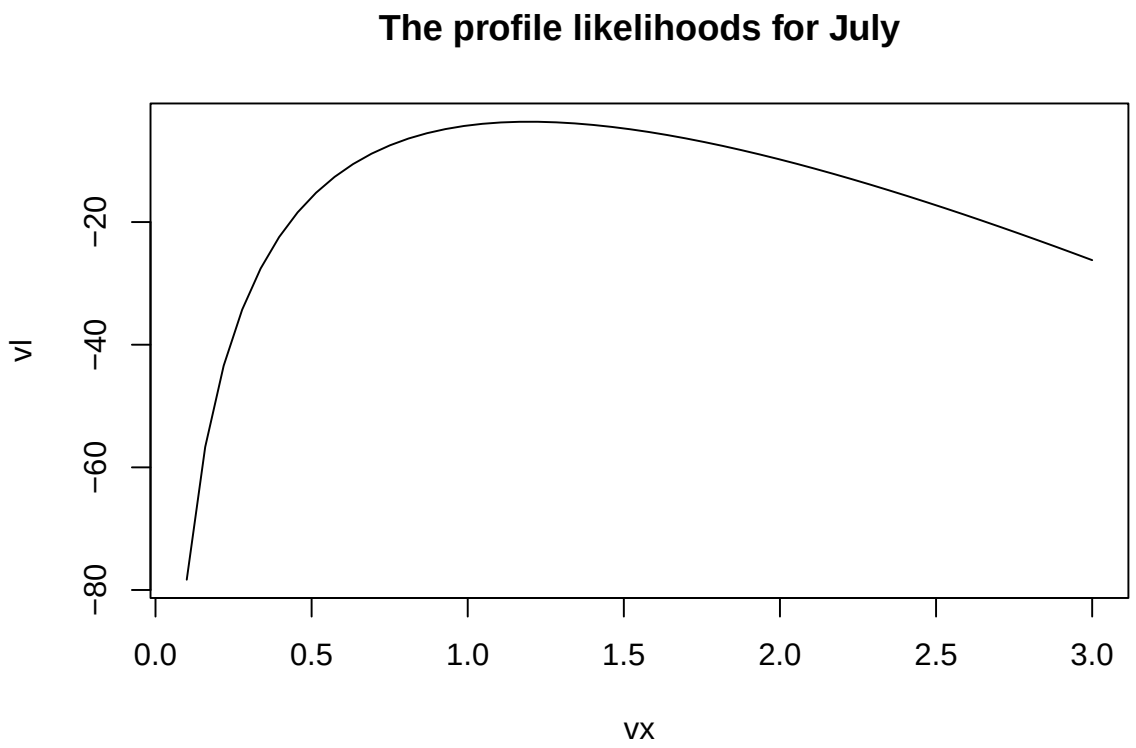
```
##           shape      rate
## shape 1.0000000 0.8103948
## rate  0.8103948 1.0000000
```

July's MLE is higher than Jan's. July's model is better than Jan's.
Jan's alpha is lower than July's alpha. Jan's beta is lower than July's beta.

```
# draw the profile likelihoods for the mean parameters
x <- Jan
prof_log_lik = function(a){
  b=(optim(1,function(z)-
             sum(log(dgamma(x,a,z)))))$par
  return(-sum(log(dgamma(x,a,b))))
}
vx=seq(.1,3,length=50)
vl=-Vectorize(prof_log_lik)(vx)
plot(vx,vl,type="l", main = "The profile likelihoods for Jan")
```

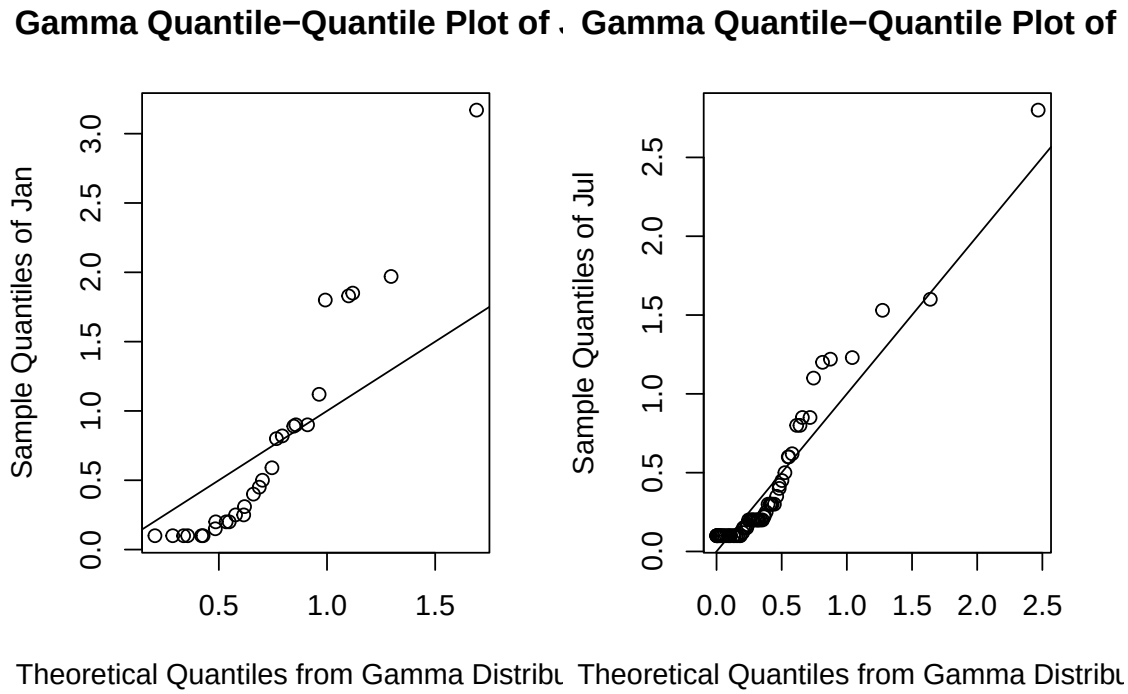



```
y <- Jul
prof_log_lik = function(a){
  b=(optim(1, function(z)-
             sum(log(dgamma(y, a, z)))))$par
  return(-sum(log(dgamma(y, a, b))))
}
vx=seq(.1, 3, length=50)
vl=-Vectorize(prof_log_lik)(vx)
plot(vx, vl, type="l", main = "The profile likelihoods for July")
```



(d) Check the adequacy of the gamma model using a gamma QQ-plot.

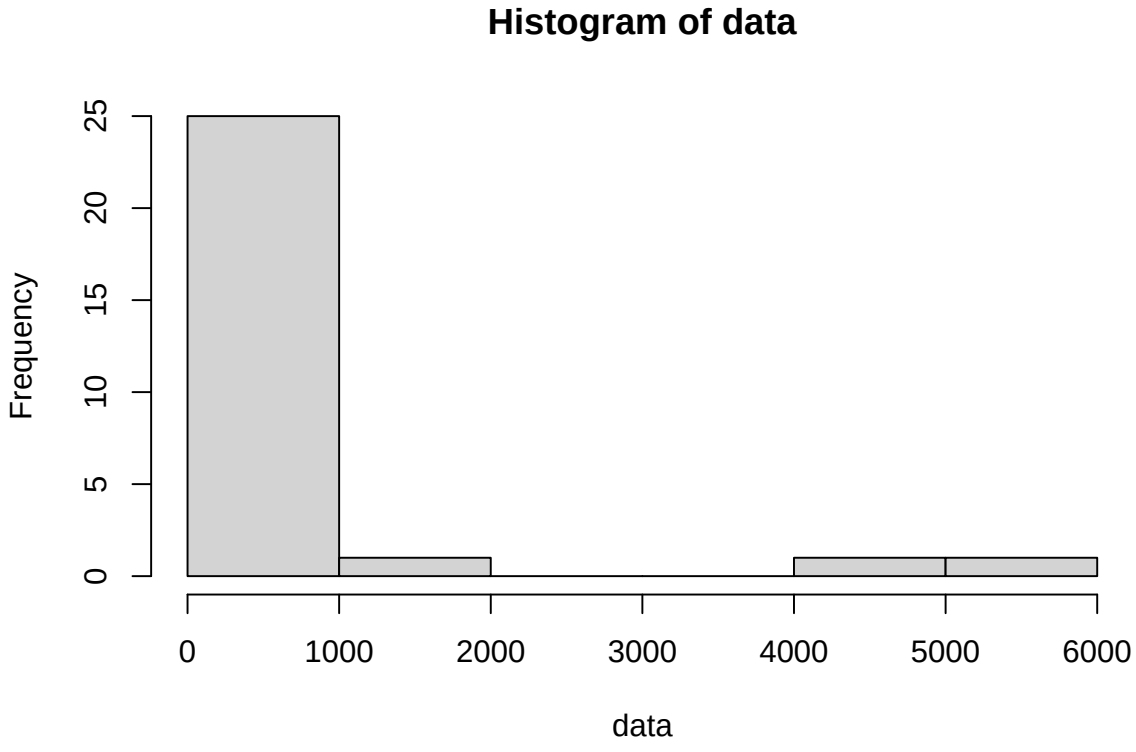
```
par(mfrow = c(1, 2))
# calculate gamma quantiles using mean and standard deviation from "ozone" to calculate shape and scale
mean_Jan <- fit_Jan$estimate[1]/fit_Jan$estimate[2]
var_Jan <- (fit_Jan$sd)^2
probabilities = (1:length(Jan))/(length(Jan)+1)
gamma.quantiles = qgamma(probabilities, shape = mean_Jan^2/var_Jan, scale = var_Jan/mean_Jan)
# gamma quantile-quantile plot for "ozone"
plot(sort(gamma.quantiles), sort(Jan), xlab = 'Theoretical Quantiles from Gamma Distribution', ylab = '
abline(0,1)

mean_Jul <- fit_Jul$estimate[1]/fit_Jul$estimate[2]
var_Jul <- (fit_Jul$sd)^2
probabilities = (1:length(Jul))/(length(Jul)+1)
gamma.quantiles = qgamma(probabilities, shape = mean_Jul^2/var_Jul, scale = var_Jul/mean_Jul)
# gamma quantile-quantile plot for "ozone"
plot(sort(gamma.quantiles), sort(Jul), xlab = 'Theoretical Quantiles from Gamma Distribution', ylab = '
abline(0,1)
```



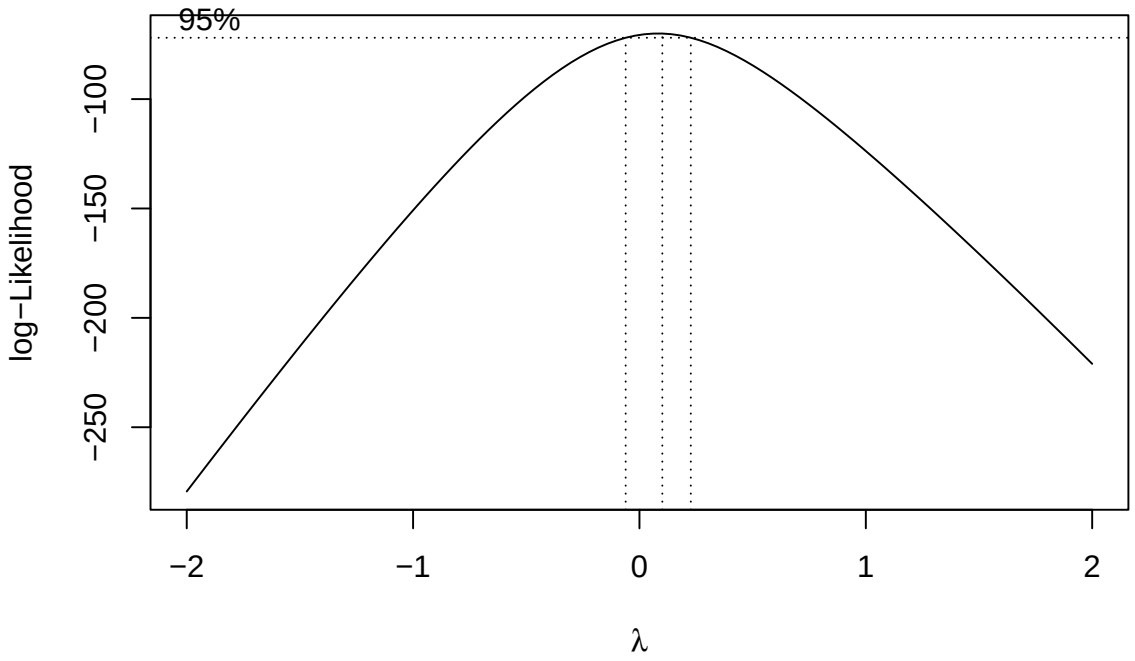
The adequacy of gamma model is greater than the normal distribution model.

Exercise 4.39

```
data <- 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, 179.0, 180.0, 406.0,
          419.0, 423.0, 440.0, 655.0, 680.0, 1320.0, 4603.0, 5712.0)
hist(data)
```



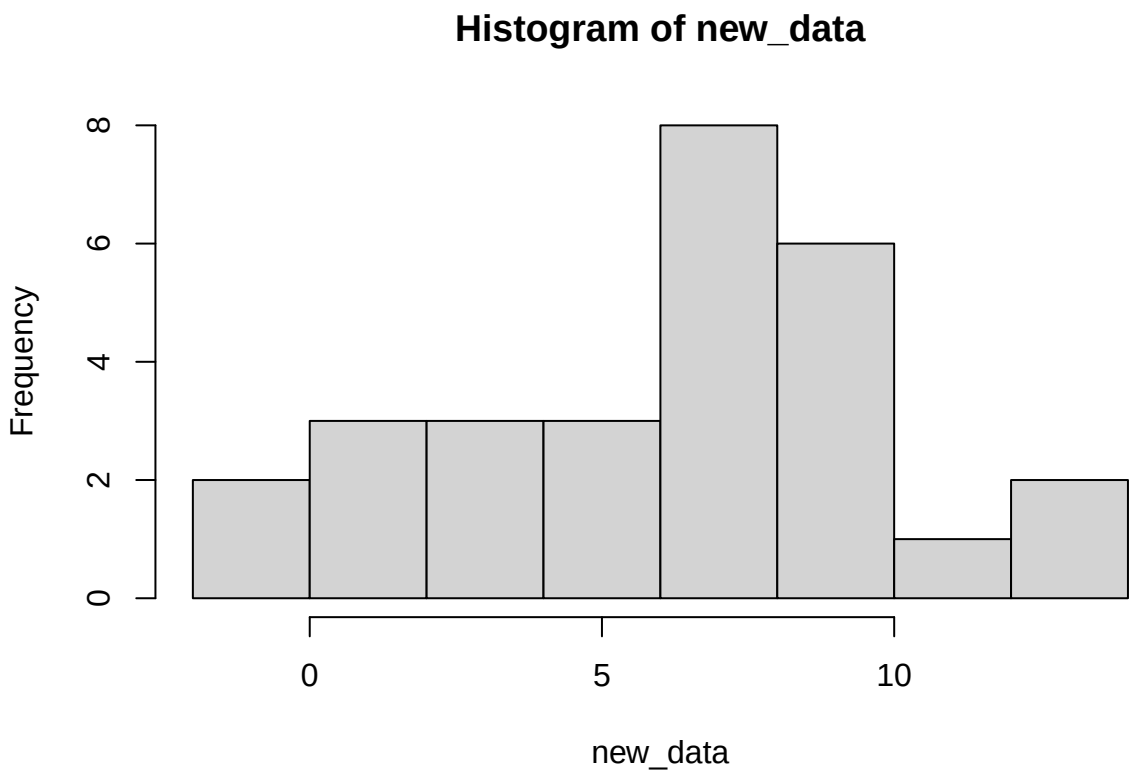
```r
# Use the Box-Cox transformation family to find which transform would be sensible to analyse or present
# fit linear regression model
model <- lm(data ~ 1)
# find optimal lambda for Box-Cox transformation
bc <- MASS::boxcox(model)
```




```r
(lambda <- bc$x[which.max(bc$y)])
```

```
## [1] 0.1010101
```

```r
# The best transformation lambda = 0.1010101
# fit new linear regression model using the Box-Cox transformation
new_model <- lm(((data ^ lambda - 1) / lambda) ~ 1)
new_data <- (data ^ lambda - 1) / lambda
hist(new_data)
```



Second Part

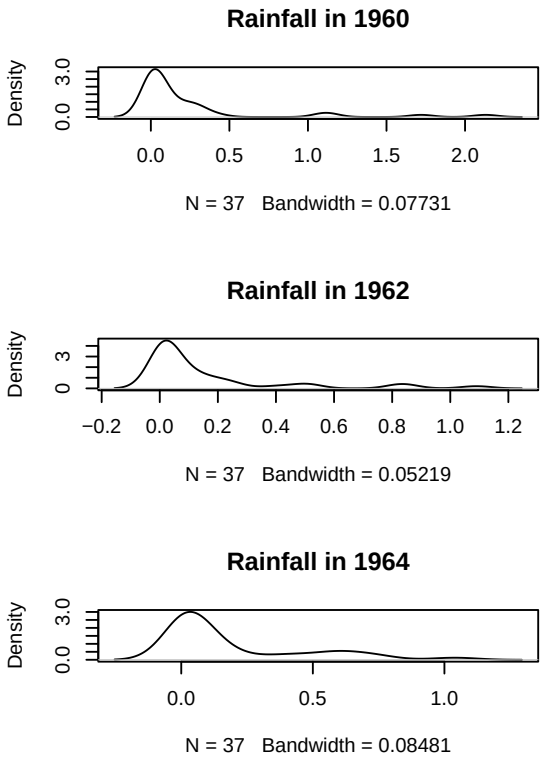
Consider the data in the spreadsheet file ***Illinois rain 1960-1964.xlxs***, which reports amounts of precipitation during storms in Illinois from 1960 to 1964. These data were gathered in a study of the natural variability of rainfall. The rainfall from summer storms was measured by a network of rain gauges in southern Illinois for the years 1960-1964 (Changnon and Huff, 1967). The average amount of rainfall (in inches) from each storm, by year, is contained in the spreadsheet.

```
rain <- read.xlsx('Illinois_rain_1960-1964.xlsx')
```

Question1:

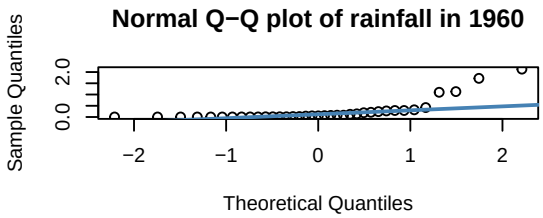
Use the data to identify the distribution of rainfall produced by the storms in southern Illinois. Estimate the parameters of the distribution using MLE. Prepare a discussion of your estimation, including how confident you are about your identification of the distribution and the accuracy of your parameter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

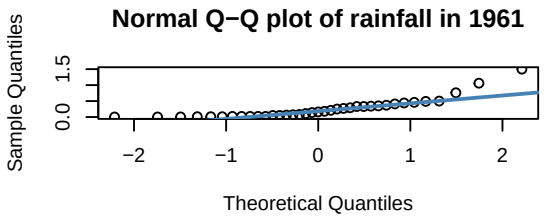
```
# identify the distribution
rain <- rain %>% na.omit()
years <- c(1960, 1961, 1962, 1963, 1964)
par(mfrow = c(3, 2))
for (i in 1:5){
  density(rain[,i]) %>% plot(main=paste("Rainfall in", years[i]))
}
```

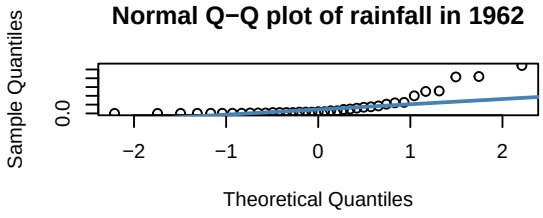


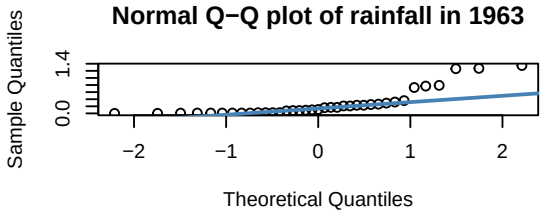
Firstly, I am looking into the density plots. I also plot the normal Q-Q plots.

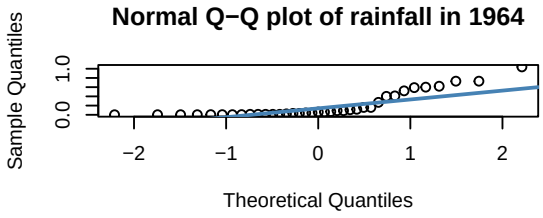
```
years <- c(1960, 1961, 1962, 1963, 1964)
par(mfrow = c(3, 2))
for (i in 1:5){
  qqnorm(rain[,i], pch = 1, main = paste("Normal Q-Q plot of rainfall in", years[i]))
  qqline(rain[,i], col = "steelblue", lwd = 2)
}
```









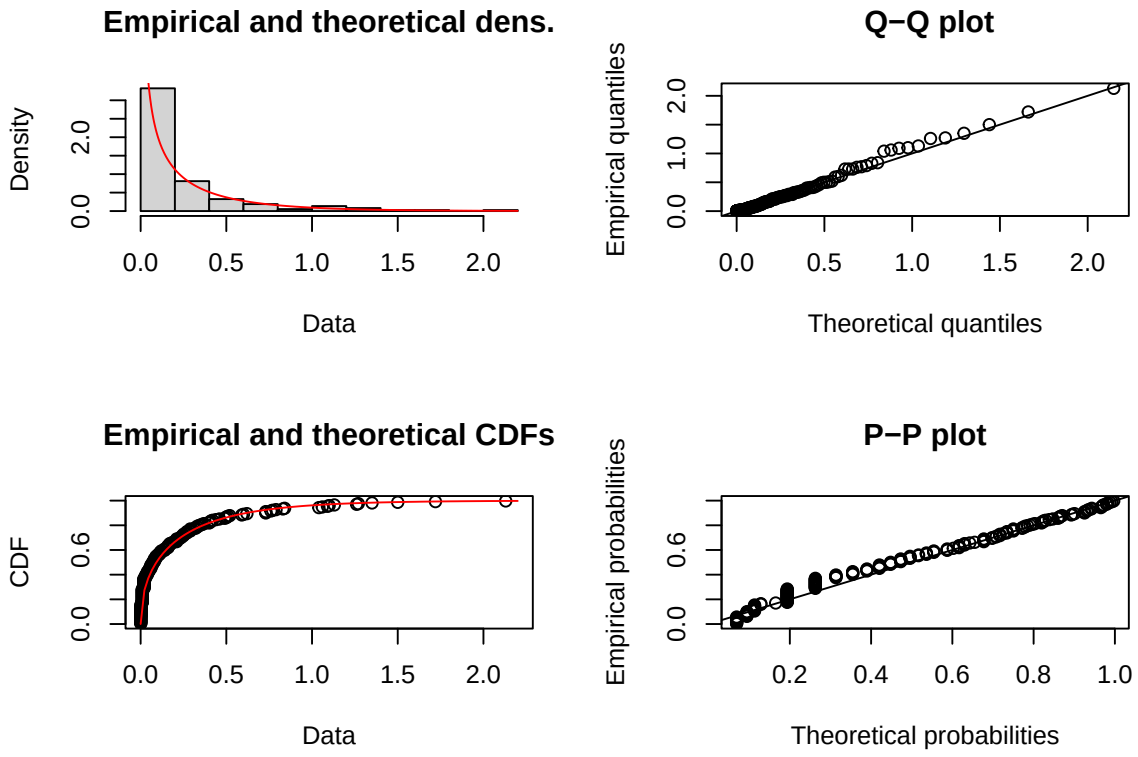


Since the observations are far from the normal distribution line, the sample does not follow the normal distribution. In addition, the observations are continuously distributed, and the gamma distribution can be considered.

```
fit<-fitdist(unlist(rain) %>% c(),'gamma',method='mle') #MLE estimation
summary(bootdist(fit)) #boot get confidence interval
```

```
## Parametric bootstrap medians and 95% percentile CI
##          Median      2.5%     97.5%
## shape 0.4524049 0.3837939 0.5474009
## rate  2.0313554 1.5730761 2.7736134
```

```
plot(fit)
```




```
percip_param <- matrix(NA, ncol = 4, nrow = 5)
colnames(percip_param) <- c("shape", "rate", "shape_sd", "rate_sd")
for (i in 1:5) {
  fit <- fitdist(rain[,i], distr = "gamma", method = "mle")
  percip_param[i, 1] <- as.numeric(fit$estimate)[1]
  percip_param[i, 2] <- as.numeric(fit$estimate)[2]
  percip_param[i, 3] <- as.numeric(fit$sd)[1]
  percip_param[i, 4] <- as.numeric(fit$sd)[2]
}
rownames(percip_param) <- years
percip_param <- data.frame(percip_param)
percip_param <- percip_param %>% mutate(mean = shape/rate,
                                        per_shape_sd = shape_sd/shape,
                                        per_rate_sd  = rate_sd/rate)
percip_param$year <- years
percip_param <- percip_param %>% dplyr::select(year, shape, rate, shape_sd, rate_sd, mean, per_shape_sd
percip_param$year <- as.character(percip_param$year)
flextable(percip_param) %>% theme_booktabs() %>% autofit()
```

```
## Warning: Warning: fonts used in `flextable` are ignored because the `pdflatex`
## engine is used and not `xelatex` or `lualatex`. You can avoid this warning
## by using the `set_flextable_defaults(fonts_ignore=TRUE)` command or use a
## compatible engine by defining `latex_engine: xelatex` in the YAML header of the
## R Markdown document.
```

year	shape	rate	shape_sd	rate_sd	mean	per_shape_sd
1960	0.3455613	1.405946	0.06417400	0.4719872	0.2457856	0.1857095
1961	0.6133472	2.414985	0.11999756	0.6929839	0.2539756	0.1956438

year	shape	rate	shape_sd	rate_sd	mean	per_shape_sd
1962	0.4407661	2.692241	0.08362145	0.8398382	0.1637172	0.1897184
1963	0.5283565	2.013200	0.10192874	0.5984650	0.2624461	0.1929166
1964	0.4395231	2.303906	0.08336429	0.7192702	0.1907730	0.1896699

The mean of rainfall in 1962 and 1964 are more lower than the other rainfall.

Question2:

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? ***sol.n:*** If the mean of rainfall is higher, we can say it is a wet year. And if the mean of rainfall is lower, we can say it is a dry year. The highest mean value occurred in 1961 with about 0.27, which can be considered as a wet year. The lowest mean value occurred in 1962 with about 0.18, so this was a relatively dry year.

Question3:

To what extent do you believe the results of your analysis are generalization? 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. ***sol.n:*** Since we only have few data with rainfall in five different years, there is not enough evidence to improve the results of your analysis are generalization. For the next step, we still need to explore the relationship of storm moving and rainfall perception, read more paper, and get more data.

Reference

[1] https://stackoverflow.com/questions/24211595/order-statistics-in-r?msclkid=fd6683dac56711ecbfcea9bd8a172395

[2] https://www.r-bloggers.com/2015/11/profile-likelihood/

[3] https://r-coder.com/box-cox-transformation-r/

[4] https://github.com/MA615-Yuli/MA677_final/blob/main/final.Rmd

[5] https://github.com/BU-Franky/677_final_proj/blob/main/Final_Assignment_Yifan_Zhang.Rmd