BANA7051 Assignment 2

Ang Zhang

1. Suppose the population mean of the variable "density" is , do the following inferences:

a. Provide an estimate of based on the sample

```
wine <- read.csv("data/winequality-red.csv", sep = ";")
density <- wine$density
mu.hat <- mean(density)
print(paste('estimate of mu based on sample is', format(mu.hat, digits = 3)))
```

```
## [1] "estimate of mu based on sample is 0.997"
```

b. Use the Central Limit Theorem (CLT) to quantify the variability of your estimate ### c. Use the CLT to give a 95% confidence interval for .

```
sd.mu.hat <- sd(density) / sqrt(length(density))
print(paste('variability of the estimate is', format(sd.mu.hat, digits = 3)))
```

```
## [1] "variability of the estimate is 4.72e-05"
```

```
upper <- mu.hat - 2*sd.mu.hat
lower <- mu.hat + 2*sd.mu.hat
print(paste('confidence interval of 95% is 2 times the standard deviation',
            'i.e., [', format(upper, digits = 3), ',', format(lower, digits = 3),']'))
```

```
## [1] "confidence interval of 95% is 2 times the standard deviation i.e., [ 0.997 , 0.997 ]"
```

d. Use the bootstrap method to do parts b and c, and compare the results with those obtained from the CLT. State your findings.

```
mu.hat.set <- NULL
n = length(density)
for (i in 1:2000){
  sample.bootstrap <- sample(density, size = n, replace = T)
  mu.hat.set[i] <- mean(sample.bootstrap)
}
sd.mu.hat <- sd(mu.hat.set)
upper <- quantile(mu.hat.set, 0.975)
lower <- quantile(mu.hat.set, 0.025)

print(paste('variability of the estimate is', format(sd.mu.hat,digits =3 )))
```

```
## [1] "variability of the estimate is 4.7e-05"
```

```
print(paste('confidence interval by getting the 97.5% and 2.5% quantile of the bootstrap result',
            'i.e., [', format(upper, digits = 3), ',', format(lower, digits = 3),']'))
```

```
## [1] "confidence interval by getting the 97.5% and 2.5% quantile of the bootstrap result i.e., [ 0.99
```

findings: resutls from CLT and bootstrap are quite close, but not exactly the same.

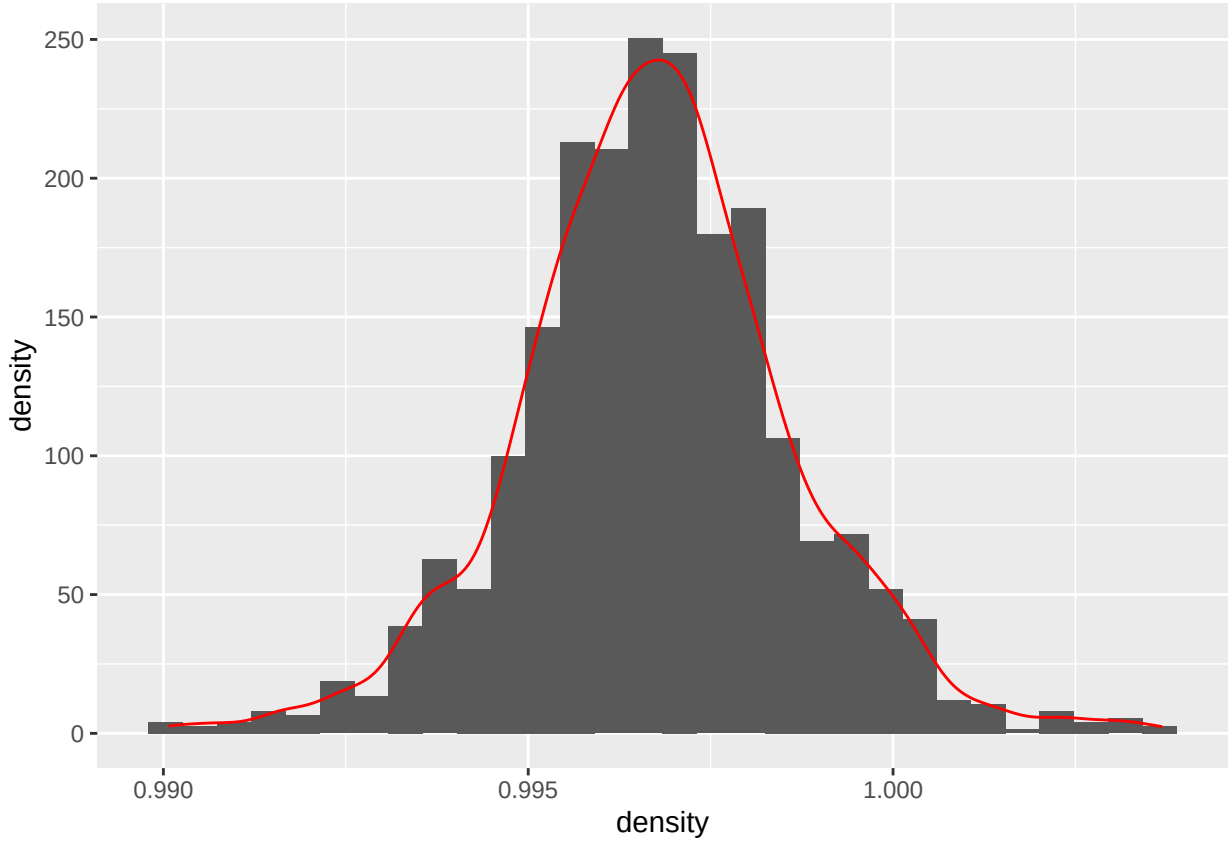
e. Can we use a normal distribution to model "density"? If yes, what are the maximum likelihood estimates of the mean and standard deviation? Please provide their standard errors as well.

First plot density using both histogram and quantile-quantile plot:

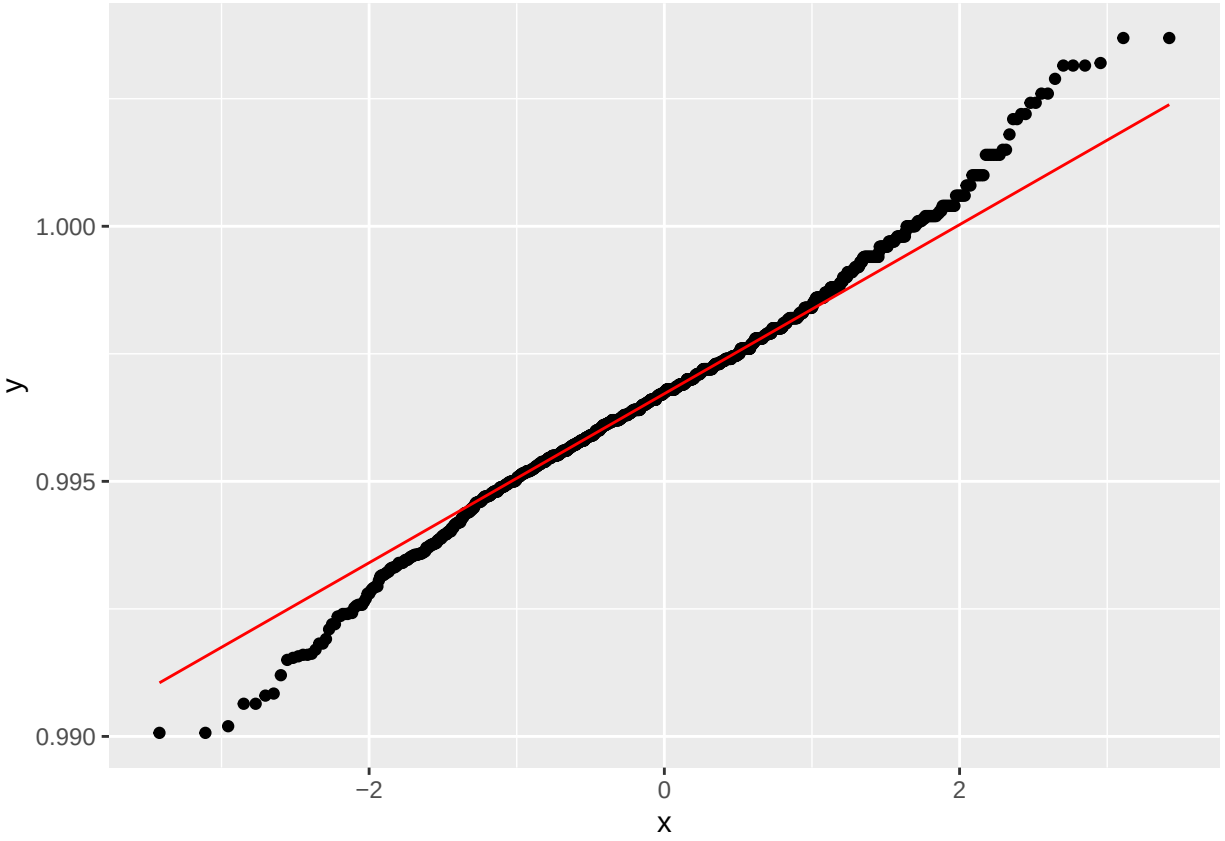
```
ggplot(data = NULL, aes(x = density)) +
  geom_histogram(aes(y = ..density..)) + geom_density(color = "red")
```

```
## Warning: The dot-dot notation (`..density..`) was deprecated in ggplot2 3.4.0.
## i Please use `after_stat(density)` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
ggplot(data= NULL, aes(sample = density)) +
  geom_qq() + geom_qq_line(color = "red")
```



so looks like the middle part of the data follows normal distribution quite well while the outter part diviates from normal distribution.

assume normal distribution, derive the log likelihood function for μ and σ, then use the mle() function from stats4 package to find the MLE estimate:

```
library(stats4)
density_10 <- density*10
likelihood.log <- function(mu, sigma){
  likelihood <- 0
  for(i in 1:length(density_10)){
    likelihood <- likelihood + log(dnorm(density_10[i], mean = mu, sd = sigma))
  }
  return(likelihood)
}
minuslog <- function(mu, sigma){
  return(-likelihood.log(mu, sigma))
}
est <- mle(minuslog = minuslog, start = list(mu = mean(density_10), sigma = sd(density_10)))
summary(est)
```

```
## Maximum likelihood estimation
##
## Call:
## mle(minuslogl = minuslog, start = list(mu = mean(density_10),
##     sigma = sd(density_10)))
##
## Coefficients:
##         Estimate   Std. Error
```

```
## mu    9.96746679 0.0004719810
## sigma 0.01887334 0.0003296881
##
## -2 log L: -8159.31
```

so the MLE estimates of μ and σ are 0.9967 and 0.001887 respectively, and the standard errors are 0.00004720 and 0.00003297 respectively.

2. Suppose the population mean of the variable "residual sugar" is , answer the following questions.

a.Provide an estimate of based on the sample

```
residual_sugar <- wine$residual.sugar
mu.hat <- mean(residual_sugar)
print(paste('estimate of mu based on sample is', mu.hat))
```

```
## [1] "estimate of mu based on sample is 2.53880550343965"
```

b. Noting that the sample distribution of "residual sugar" is highly skewed, can we use the CLT to quantify the variability of your estimate? Can we use the CLT to give a 95% confidence interval for ? If yes, please give your solution. If no, explain why.

According to CLT, regardless of the type or skewness of the data, the sample mean will roughly follow the same distribution that is normal. so we can still use CLT to give the confidence interval:

```
sd.mu.hat <- sd(density) / sqrt(length(density))
print(paste('variability of the estimate is', format(sd.mu.hat, digits = 3)))
```

```
## [1] "variability of the estimate is 4.72e-05"
```

```
upper <- mu.hat - 2*sd.mu.hat
lower <- mu.hat + 2*sd.mu.hat
print(paste('confidence interval of 95% is',
            '[', format(upper, digits = 3), ',', format(lower, digits = 3),']'))
```

```
## [1] "confidence interval of 95% is [ 2.54 , 2.54 ]"
```

c. Use the bootstrap method to do part b. Is the bootstrap confidence interval symmetric? (Hint: check the bootstrap distribution; see p. 43 in Lecture 3).

```
mu.hat.set <- NULL
n = length(residual_sugar)
for (i in 1:2000){
  sample.bootstrap <- sample(residual_sugar, size = n, replace = T)
  mu.hat.set[i] <- mean(sample.bootstrap)
}
sd.mu.hat <- sd(mu.hat.set)
upper <- quantile(mu.hat.set, 0.975)
lower <- quantile(mu.hat.set, 0.025)

print(paste('variability of the estimate is', sd.mu.hat))
```

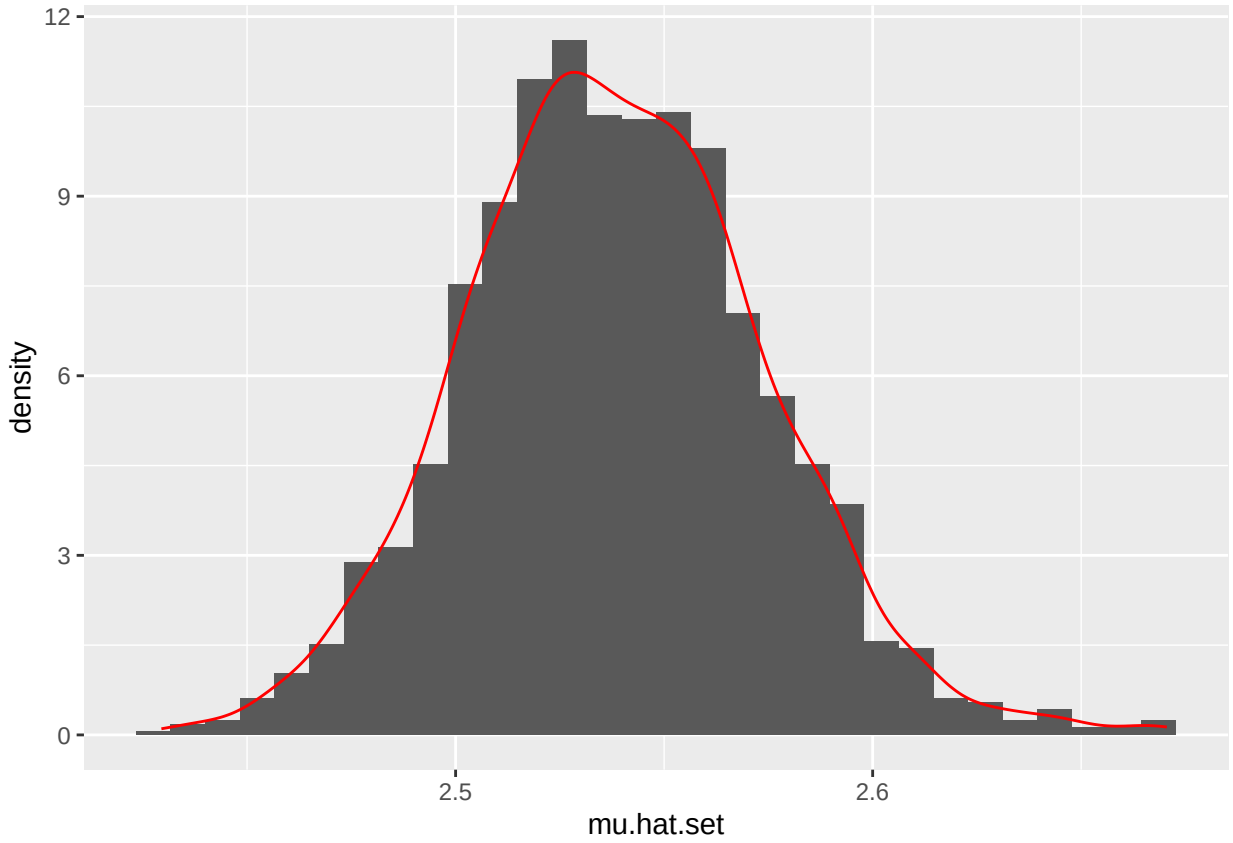
```
## [1] "variability of the estimate is 0.0356795921652999"
```

```
print(paste('confidence interval by getting the 97.5% and 2.5% quantile of the bootstrap result',
            'i.e., [', format(upper, digits = 3), ',', format(lower, digits = 3),']'))
```

```
## [1] "confidence interval by getting the 97.5% and 2.5% quantile of the bootstrap result i.e., [ 2.61
```

```
ggplot(data = NULL, aes(mu.hat.set)) +
  geom_histogram(aes(y = ..density.. )) +
  geom_density(color = "red")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



from the histogram we can tell that the the distribution of the set of mus is not symmetric.

d. Use the sample median as an estimate of the population mean and provide the 95% confidence interval using the appropriate method.

```
mu.hat.set <- NULL
n = length(residual_sugar)
for (i in 1:2000){
  sample.bootstrap <- sample(residual_sugar, size = n, replace = T)
  mu.hat.set[i] <- median(sample.bootstrap)
}
sd.mu.hat <- sd(mu.hat.set)
upper <- quantile(mu.hat.set, 0.975)
lower <- quantile(mu.hat.set, 0.025)

print(paste('variability of the estimate is', sd.mu.hat))
```

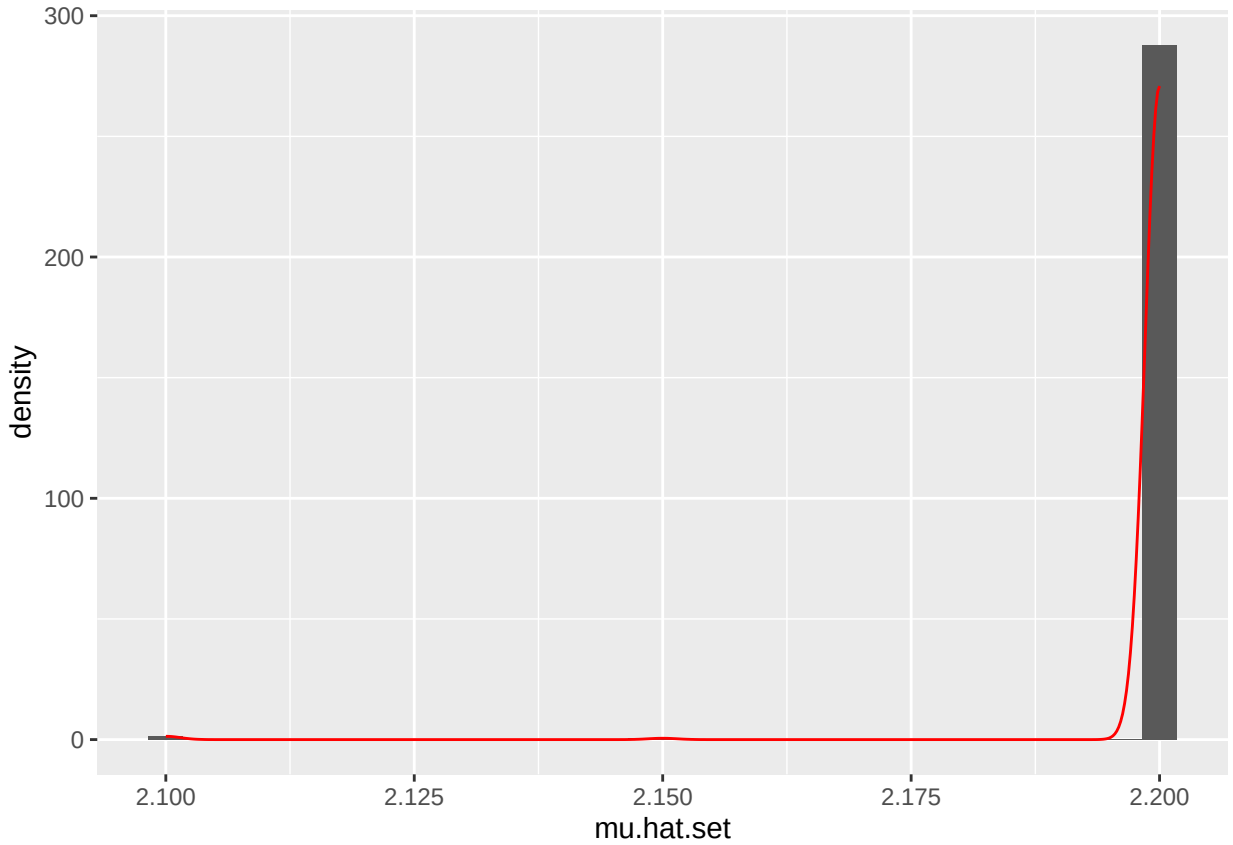
```
## [1] "variability of the estimate is 0.00739373611017031"
```

```
print(paste('confidence interval by getting the 97.5% and 2.5% quantile of the bootstrap result',
            'i.e., [', format(upper, digits = 3), ',', format(lower, digits = 3),']'))
```

```
## [1] "confidence interval by getting the 97.5% and 2.5% quantile of the bootstrap result i.e., [ 2.2
```

```
ggplot(data = NULL, aes(mu.hat.set)) +
  geom_histogram(aes(y = ..density.. )) +
  geom_density(color = "red")
```

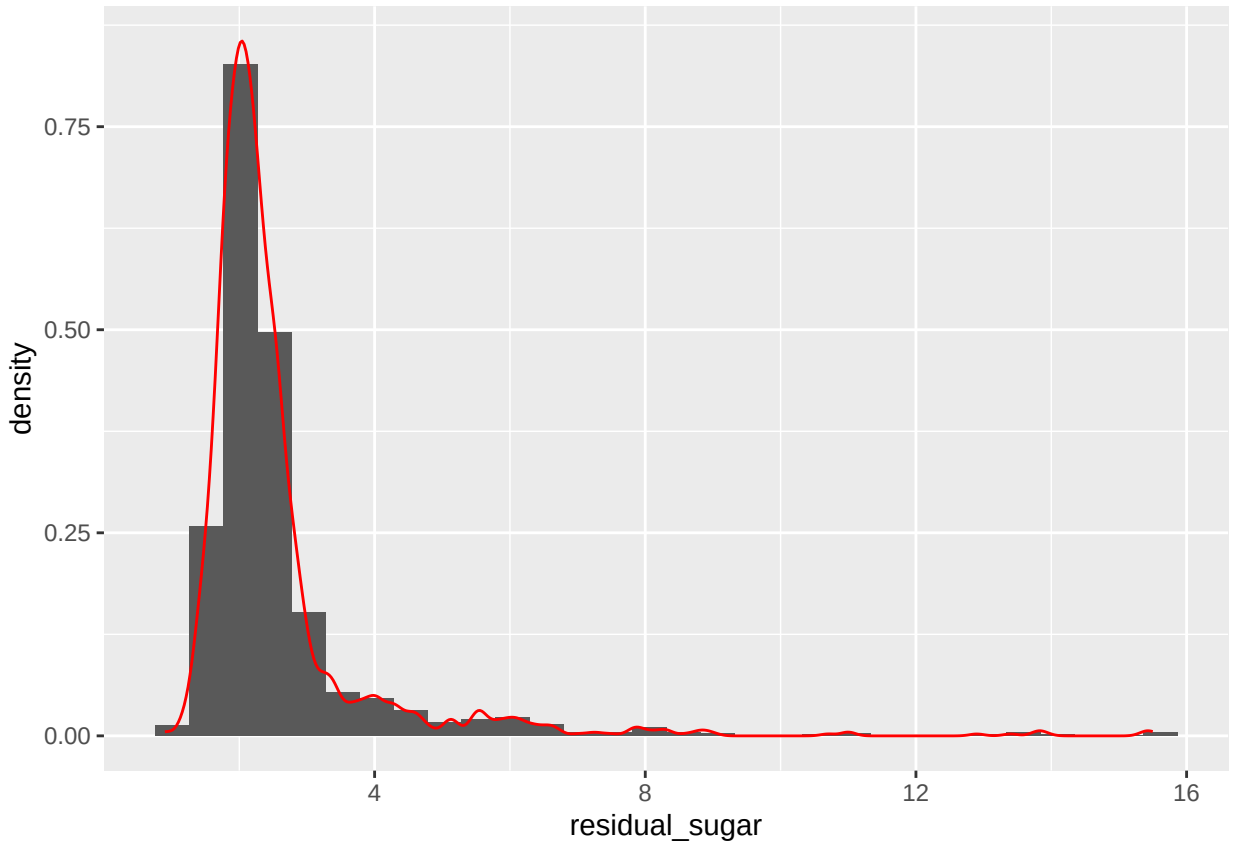
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



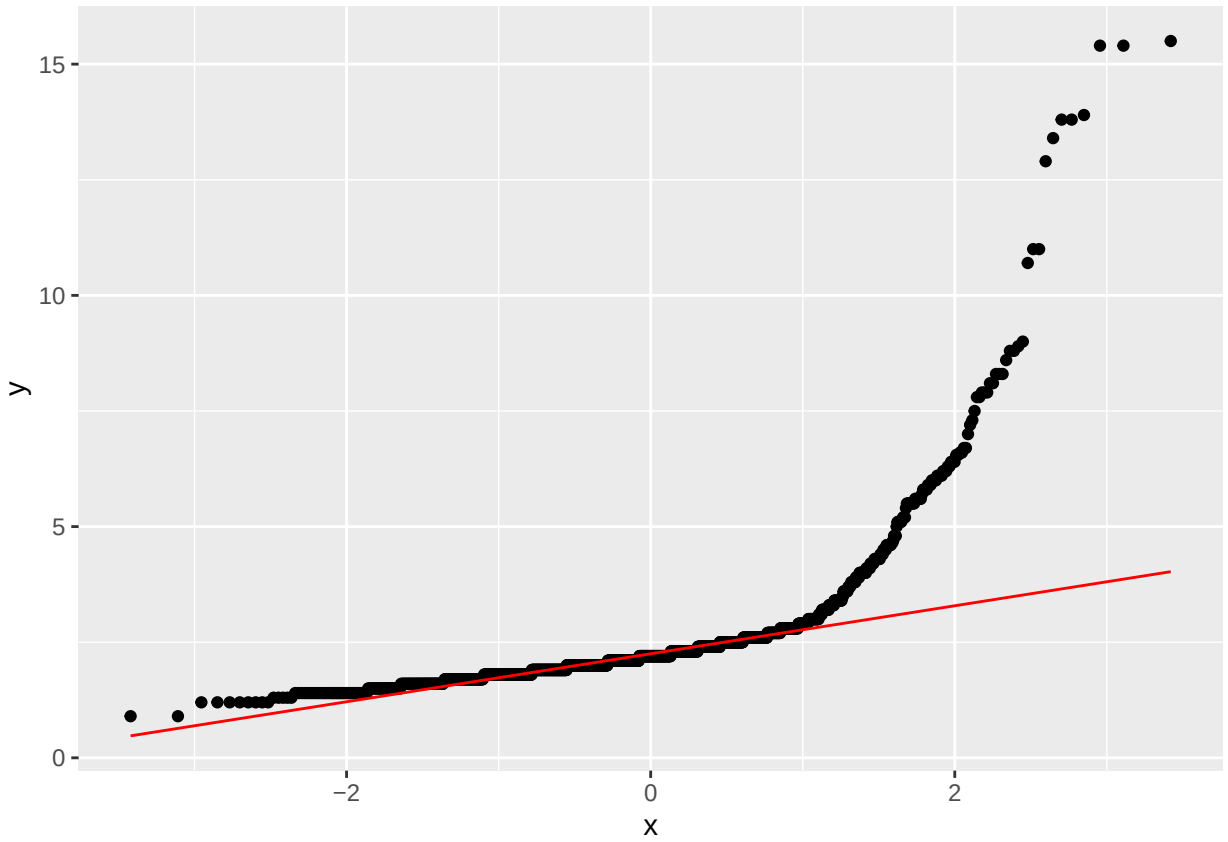
seems that sample median is not so good an estimator due to the lack of variability. ### e. Can we use a normal distribution to model "residual sugar"? If no, what distribution do you think can approximate its empirical distribution? What parameters are needed to characterize such a distribution? what are their maximum likelihood estimates? Please provide their standard errors as well.

```
ggplot(data = NULL, aes(residual_sugar)) +
  geom_histogram(aes(y = ..density.. )) + geom_density(color = "red")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



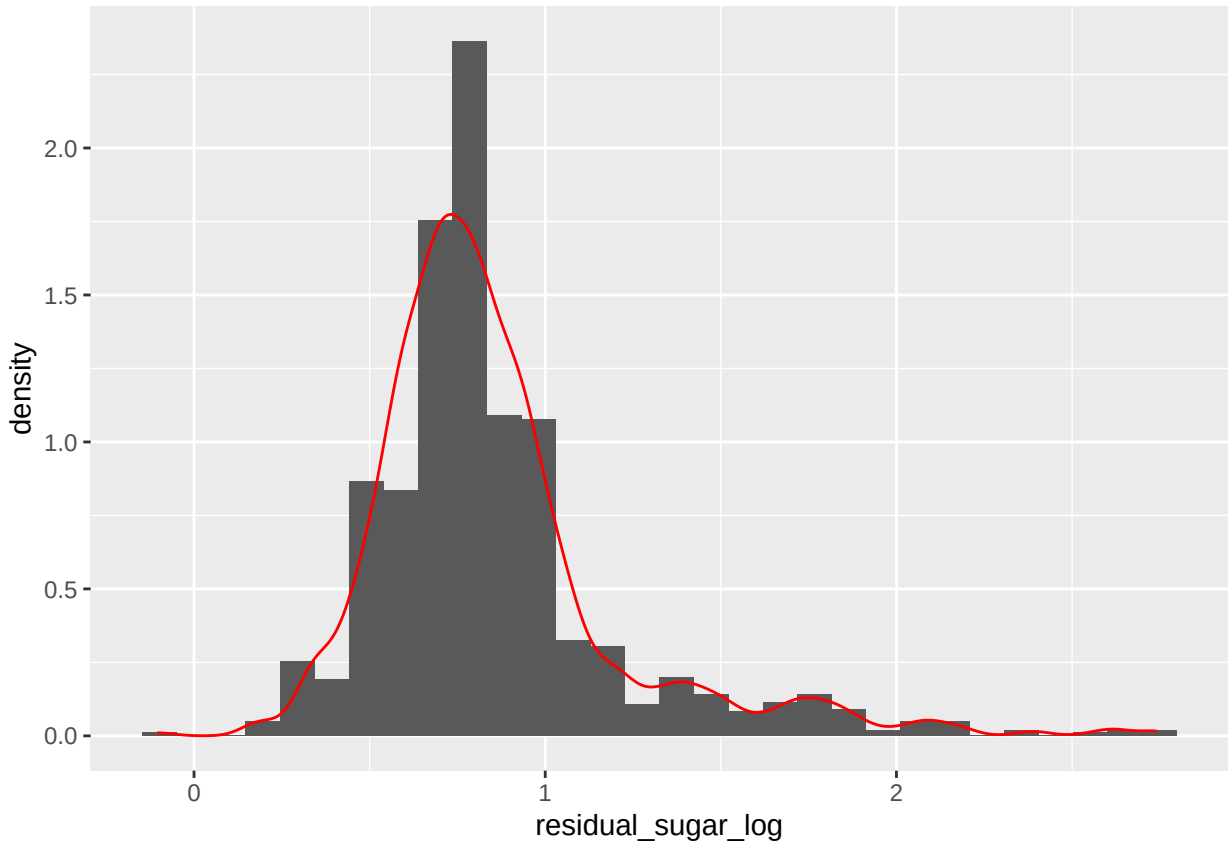

```
ggplot(data = NULL, aes(sample = residual_sugar)) +
  geom_qq() + geom_qq_line(color = "red")
```



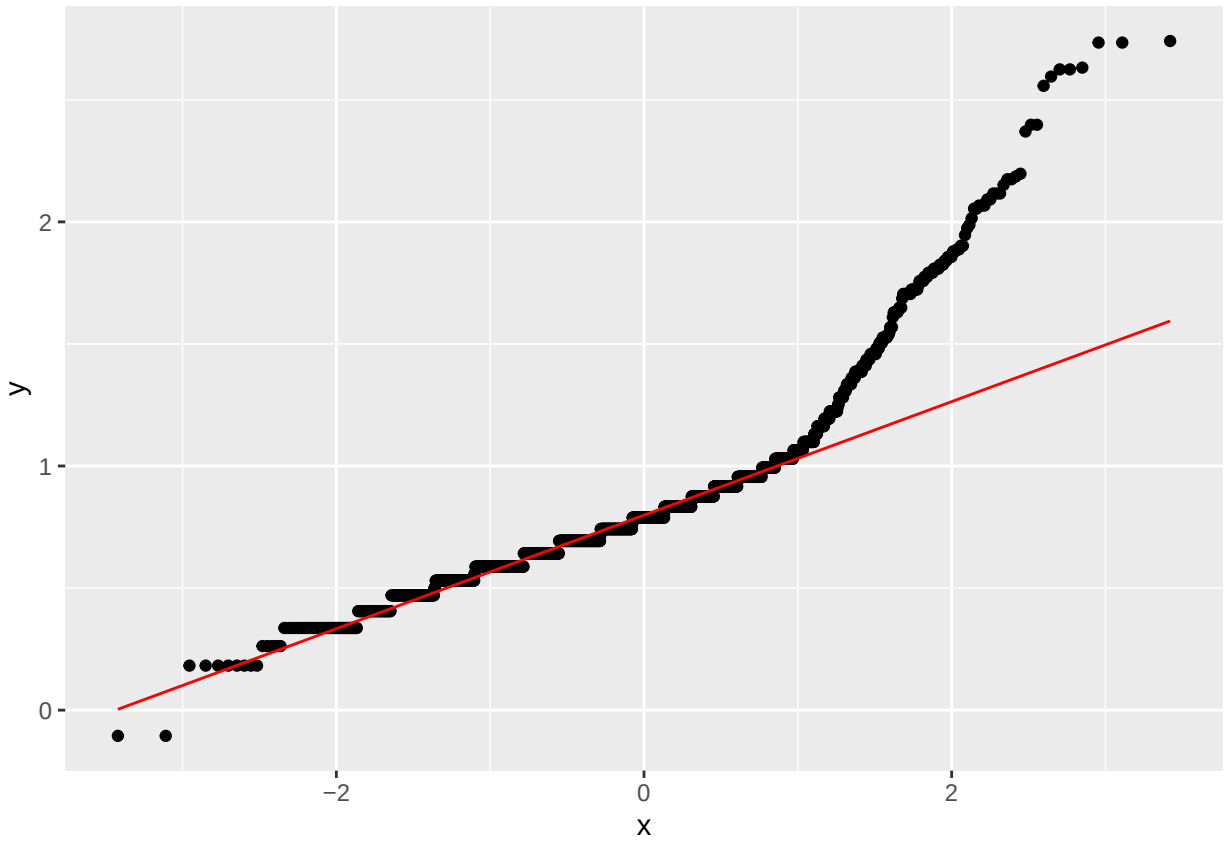
From the histogram and pp plot we can see it's no where like a normal distribution because it's highlly skewed, so using normal distribution to model it would not be a good idea. Instead we can use a log-normal distribution to model it, because the log-normal distribution accounts for skewness. First take the logarithm of residual_sugar and look at it's distribution:

```
residual_sugar_log = log(residual_sugar)
ggplot(data = NULL, aes(residual_sugar_log)) +
  geom_histogram(aes(y = ..density.. )) + geom_density(color = "red")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
ggplot(data = NULL, aes(sample = residual_sugar_log)) +
  geom_qq() + geom_qq_line(color = "red")
```



Looks much better. Then we need two parameter: α and β to characterize it. To avoid numerical problems, multiply the result of logarithm by 10 before plug into the normal distribution function:

```
library(stats4)
residual_sugar_log_10 = residual_sugar_log * 10
likelihood.log <- function(mu, sigma){
  likelihood <- 0
  for(i in 1:length(residual_sugar_log)){
    likelihood <- likelihood + log(dnorm(residual_sugar_log_10[i], mean = mu, sd = sigma))
  }
  return(likelihood)
}
minuslog <- function(mu, sigma){
  return(-likelihood.log(mu, sigma))
}

est <- mle(minuslog = minuslog, start = list(mu = mean(residual_sugar_log_10), sigma = sd(residual_sugar
summary(est)
```

```
## Maximum likelihood estimation
##
## Call:
## mle(minuslogl = minuslog, start = list(mu = mean(residual_sugar_log_10),
##     sigma = sd(residual_sugar_log_10)))
##
## Coefficients:
##       Estimate Std. Error
## mu    8.502318 0.08936081
```

```
## sigma 3.573316 0.06318761
##
## -2 log L: 8610.399
```

```
print("MLE of mu is 0.8502, standard error is 0.008936")
```

```
## [1] "MLE of mu is 0.8502, standard error is 0.008936"
```

```
print("MLE of sigma is 0.3573, standard error is 0.006319")
```

```
## [1] "MLE of sigma is 0.3573, standard error is 0.006319"
```

We classify those wines as "excellent" if their rating is at least 7. Suppose the population proportion of excellent wines is p. Do the following:

a. Use the CLT to derive a 95% confidence interval for p

```
excellence = if_else(wine$quality >= 7, 1, 0)
p <- mean(excellence)
sd.p <- sd(excellence) / sqrt(length(excellence))
print(paste('estimate of p based on sample is', format(p, digits = 3)))
```

```
## [1] "estimate of p based on sample is 0.136"
```

```
print(paste('variability of the estimate is', format(sd.p, digits = 3)))
```

```
## [1] "variability of the estimate is 0.00857"
```

```
upper <- p - 2*sd.p
lower <- p + 2*sd.p
print(paste('confidence interval of 95% is 2 times the standard deviation',
            'i.e., [', format(upper, digits = 3), ',', format(lower, digits = 3),']'))
```

```
## [1] "confidence interval of 95% is 2 times the standard deviation i.e., [ 0.119 , 0.153 ]"
```

b. Use the bootstrap method to derive a 95% confidence interval for p;

```
p.set <- NULL
n = length(excellence)
for (i in 1:2000){
  sample.bootstrap <- sample(excellence, size = n, replace = T)
  p.set[i] <- mean(sample.bootstrap)
}
sd.p <- sd(p.set)
upper <- quantile(p.set, 0.975)
lower <- quantile(p.set, 0.025)
print(paste('estimate is', format(mean(p.set), digits = 3)))
```

```
## [1] "estimate is 0.136"
```

```
print(paste('variability of the estimate is', format(sd.p, digits = 3)))
```

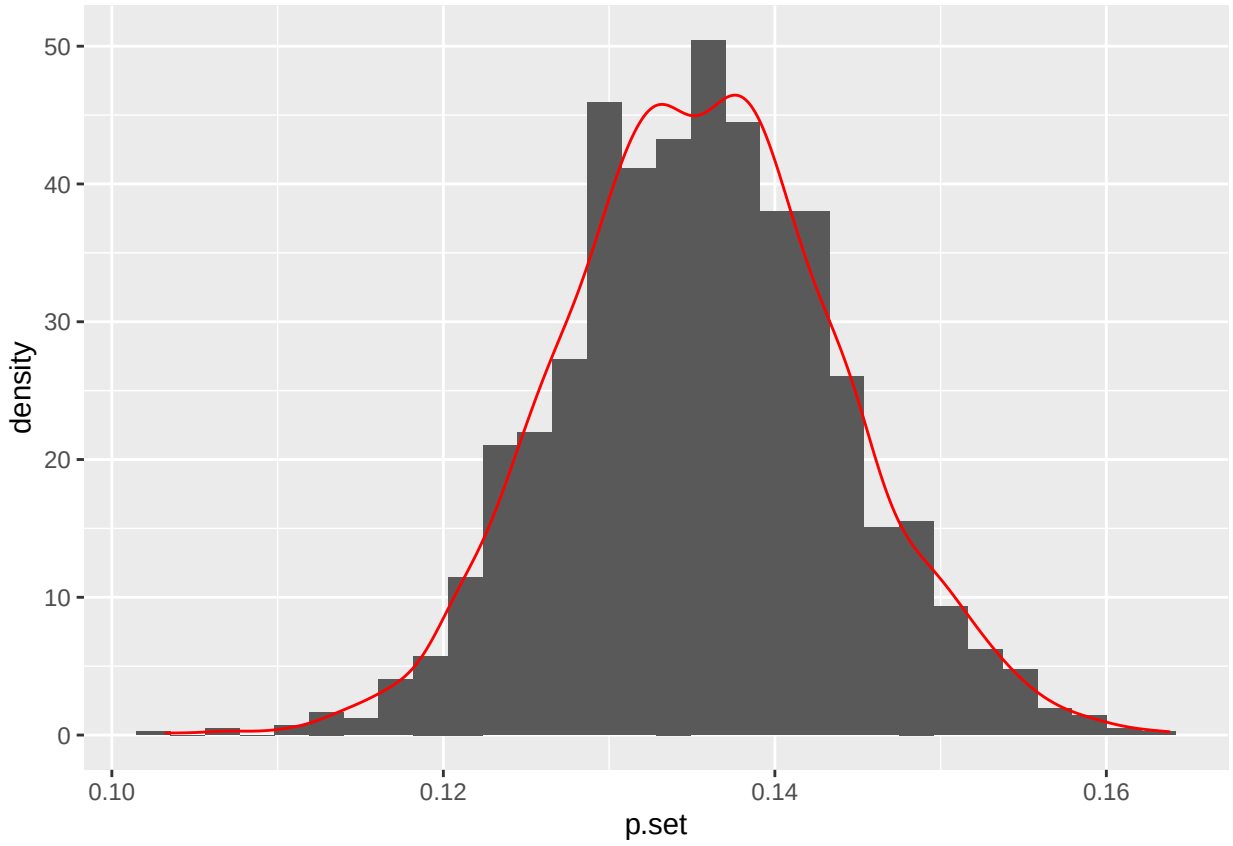
```
## [1] "variability of the estimate is 0.00839"
```

```
print(paste('confidence interval by getting the 97.5% and 2.5% quantile of the bootstrap result',
            'i.e., [', format(lower, digits = 3), ',', format(upper, digits = 3),']'))
```

```
## [1] "confidence interval by getting the 97.5% and 2.5% quantile of the bootstrap result i.e., [ 0.12
```

```
ggplot(data = NULL, aes(p.set)) +
  geom_histogram(aes(y = ..density.. )) +
  geom_density(color = "red")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



c. Compare the two intervals. Is there any difference worth our attention?

Bootstrap and CLT arrived at very close results.

d. What is the maximum likelihood estimate of p and its standard error?

```
minus.log <- function(p){
  -log(dbinom(x = sum(excellence), size = length(excellence), prob = p))
}
est <- mle(minuslog = minus.log, start = list(p = 0.136))
```

```
## Warning in dbinom(x = sum(excellence), size = length(excellence), prob = p):
## NaNs produced
## Warning in dbinom(x = sum(excellence), size = length(excellence), prob = p):
## NaNs produced
## Warning in dbinom(x = sum(excellence), size = length(excellence), prob = p):
## NaNs produced
```

```
summary(est)
```

```
## Maximum likelihood estimation
##
```

```
## Call:
## mle(minuslogl = minus.log, start = list(p = 0.136))
##
## Coefficients:
##    Estimate  Std. Error
## p 0.1357118 0.008564384
##
## -2 log L: 7.072712
```

so MLE of p is 0.1357, standard error is 0.008564. Again this is very close to the CLT and bootstrap results.