Exercise4

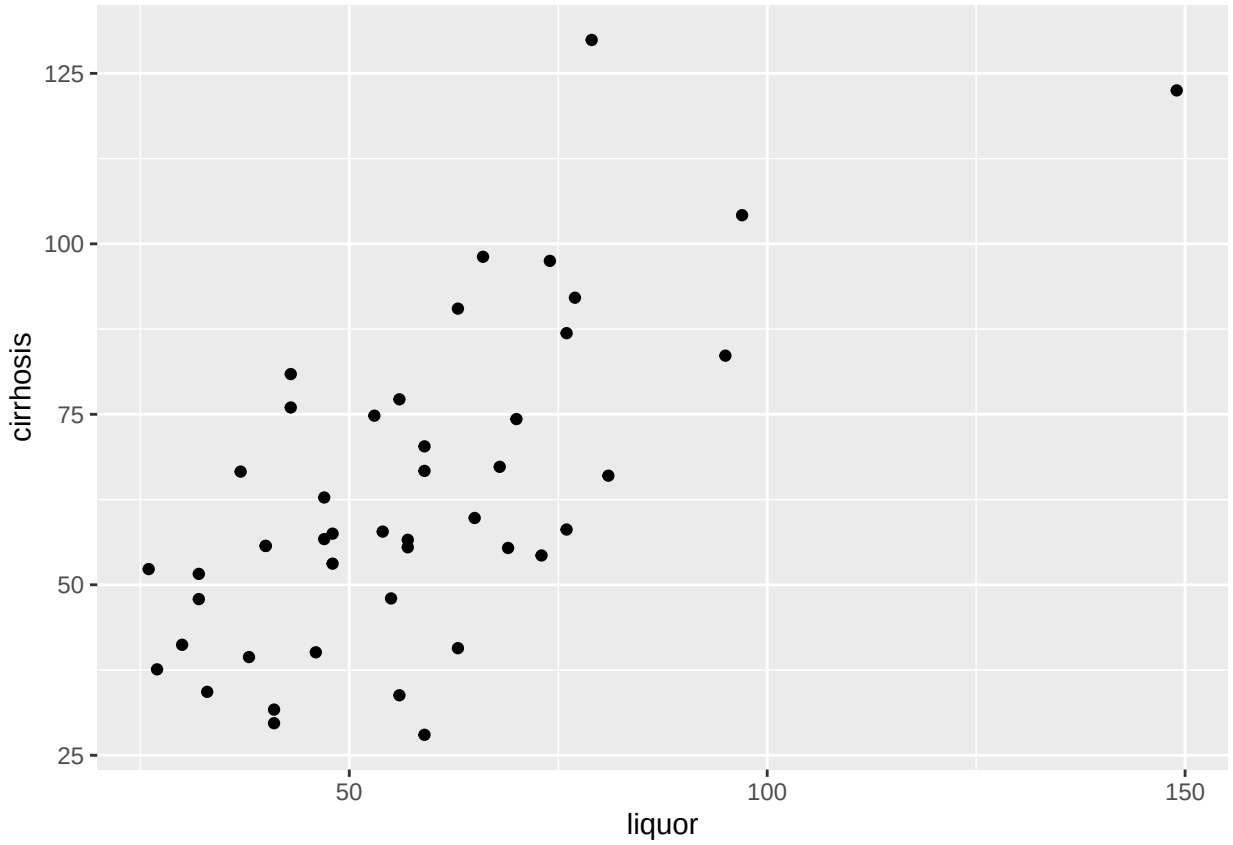
Chutian Chen cc4515; Congcheng Yan cy2550; Mingrui Liu ml4404

```
library(ggplot2)
```

```
data <- read.csv('liver.csv')
```

(1)

```
ggplot(data) + geom_point(aes(x = liquor, y = cirrhosis))
```



From the graph, we can see positive linear correlation between two variables. So it's reasonable to fit a straight line to this data set.

(2)

The response variable is cirrhosis mortality rate.

(3)

I expect the slope and intercept of the line and to be positive. Because more liquor consumption leads to worse health. The mortality rate would rise. And if people don't drink alcohol, people would still die because of cirrhosis. So the intercept should be positive.

(4)

(a) α is the cirrhosis mortality rate in region where people don't drink alcohol. β represents the influence caused by liquor consumption per capita on the cirrhosis mortality rate.

(b) α is the cirrhosis mortality rate in region where liquor consumption is equal to the mean consumption. β represents the influence caused by liquor consumption per capita on the cirrhosis mortality rate.

(5)

We use model (a) here.

```
model <- lm(cirrhosis~liquor, data = data)
summary(model)
```

```
##
## Call:
## lm(formula = cirrhosis ~ liquor, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -36.577 -11.127  -0.821  11.179  50.878
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  21.9649     7.1847   3.057  0.00379 **
## liquor        0.7222     0.1168   6.185  1.8e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 17.34 on 44 degrees of freedom
## Multiple R-squared:  0.4651, Adjusted R-squared:  0.4529
## F-statistic: 38.26 on 1 and 44 DF,  p-value: 1.803e-07
```

$\alpha = 21.96$, $\beta = 0.72$. They are statistically significant.

It shows that in region where people don't drink alcohol the cirrhosis mortality rate is 21.96 per 100,000 people. And for each capita increase of liquor consumption, the cirrhosis mortality rate would increase by 0.722.

(6)

The least square estimates are the best linear fit as we estimate a linear model by square function.

(7)

```
predict(model,data.frame(liquor=180))
```

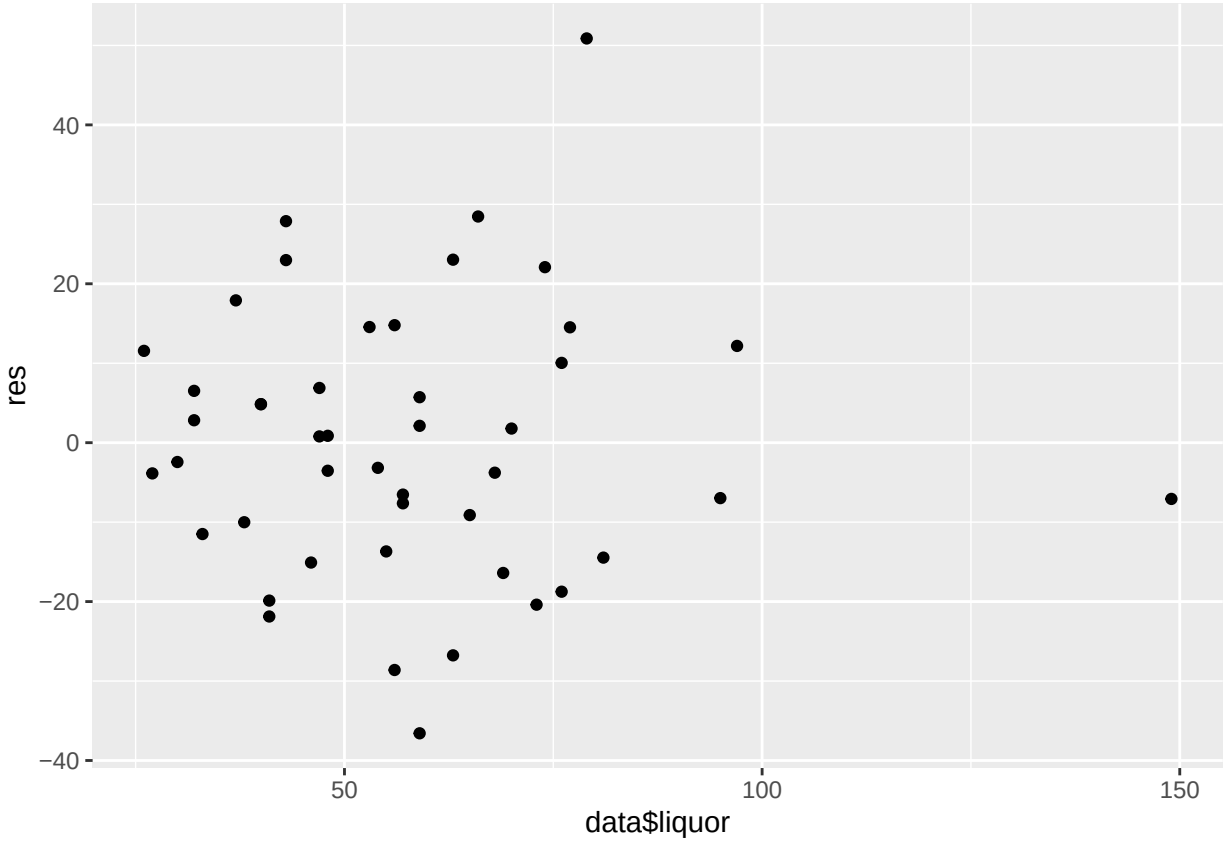
```
##        1
## 151.9673
```

Prediction: 151.97

It's not a good predictor. Because in the dataset there is no liquor that larger than 150. And there is only one data of liquor larger than 100. So it's not appropriate to use the model to predict region with liquor consumption of 180 ounces.

(8)

```
res <- resid(model)
ggplot() + geom_point(aes(x = data$liquor, y = res))
```



Yes, it does. We can see that the residuals are randomly distributed around 0. It's reasonable to assume iid.

(9)

As we know: $\sqrt{\frac{(n-2)S_{xx}}{SSR}}(\hat{\beta}-\beta) \sim t_{n-2}$

$S_{xx} = \sum_{i=1}^{n}(x_i - \bar{x})^2$

$SSR = \sum_{i=1}^{n}(Y_i - \hat{\alpha} - \hat{\beta}x_i)^2$

So the 95% confidence interval for β is

$\hat{\beta} \pm t_{0.975,n-2}\sqrt{\frac{(n-2)(Y_i-\hat{\alpha}-\hat{\beta}x_i)^2}{(x_i-\bar{x})^2}}$

```
residual <- sum(res^2)
sxx <- sum(data$liquor^2) - nrow(data)*(mean(data$liquor)^2)
interval <- qt(0.975,nrow(data)-2) * sqrt(residual/(nrow(data)-2)/sxx)
l <- as.numeric(coef(model)['liquor'])-interval
r <- as.numeric(coef(model)['liquor'])+interval
print(c(l,r))
```

```
## [1] 0.4869010 0.9575696
```

So the confidence interval is (0.487,0.958)

From Exercise 3 Q4.4 we know that without normality assumption, we have:

$/sqrt(n)(\hat{\beta}_{LS} - \beta) \rightarrow^{D} N(0, \sigma^2/\sigma_X^2)$

where $\sigma_X^2 = lim_{n\rightarrow\infty} \frac{1}{n} \sum_{i=1}^{n} (X_i - \bar{X})^2$

Use the Slutsky's Theorem we can get the aymptotic confidence interval is $\beta \pm z_{0.975} \frac{\hat{\sigma}}{\sqrt{\sum_{i=1}^{n}(X_i - \bar{X})^2}}$

We can use $\sqrt{\frac{SSR}{n-2}}$ as $\hat{\sigma}$.

```
s_hat <- sd(res)*(nrow(data)-1)/(nrow(data)-2)
interval2 <- qnorm(0.975)*s_hat/sqrt(sxx)
l2 <- as.numeric(coef(model)['liquor'])-interval2
r2 <- as.numeric(coef(model)['liquor'])+interval2
print(c(l2,r2))
```

```
## [1] 0.4907843 0.9536863
```

So the asymptotic confidence interval is (0.491,0.954)

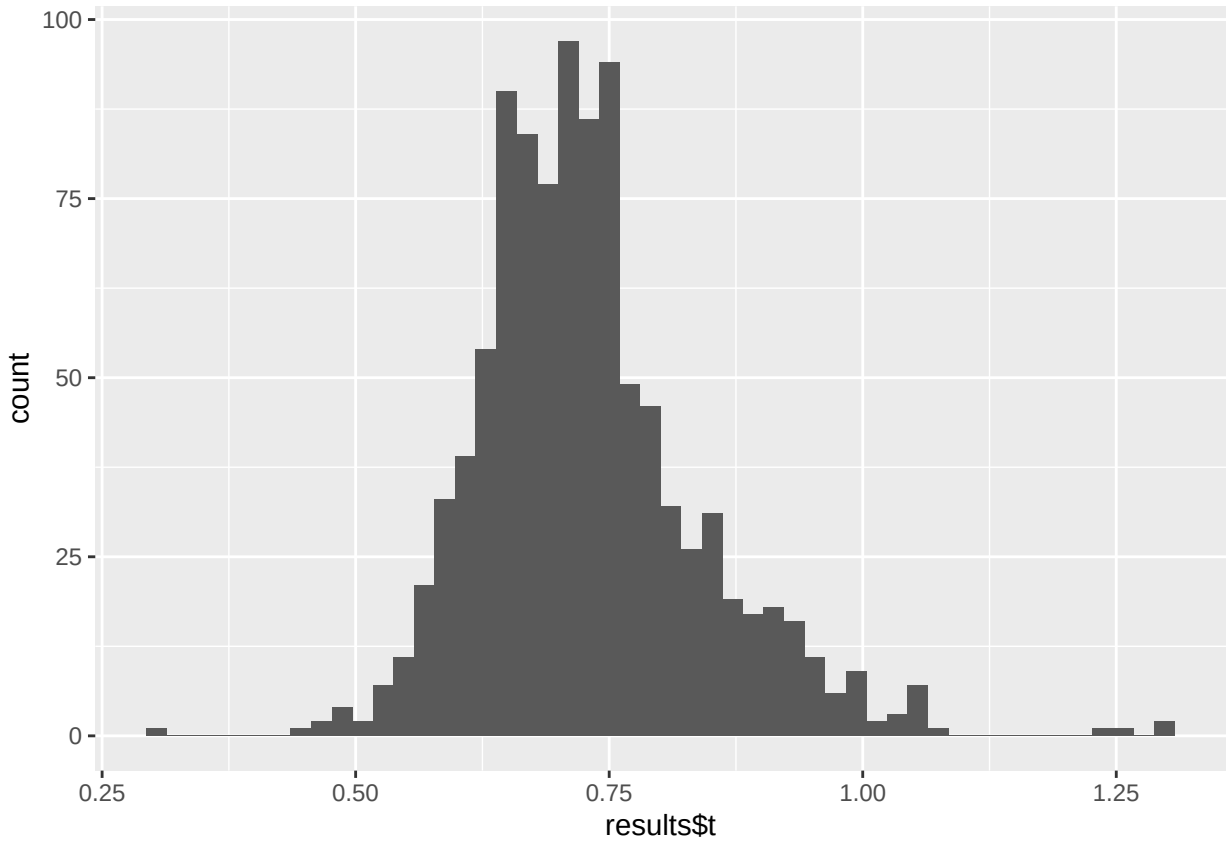
The two confidence intervals are similar. The first interval is under normal assumption. The second one is under the condition of large n. The advantage of the first one is that it can be used when n is small. The advantage of the second one is that we don't need normal assumption.

(10)

```
# Bootstrap 95% CI
library(boot)

bootstrap <- function(data, indices) {
  d <- data[indices,] # allows boot to select sample
  fit <- lm(cirrhosis~liquor, data=d)
  return(coef(fit)['liquor'])
}
# bootstrapping with 1000 replications
set.seed(31415)
results <- boot(data=data, statistic=bootstrap, R=1000)

# view results
ggplot() + geom_histogram(aes(x = results$t), bins = 50)
```




```
# get 95% confidence interval
boot.ci(results, type="bca")
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 1000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = results, type = "bca")
##
## Intervals :
## Level       BCa
## 95%   ( 0.5738,  1.0466 )
## Calculations and Intervals on Original Scale
```

The confidence interval of bootstrap is (0.5738,1.0466). It's similar to the results in (9).

(11)

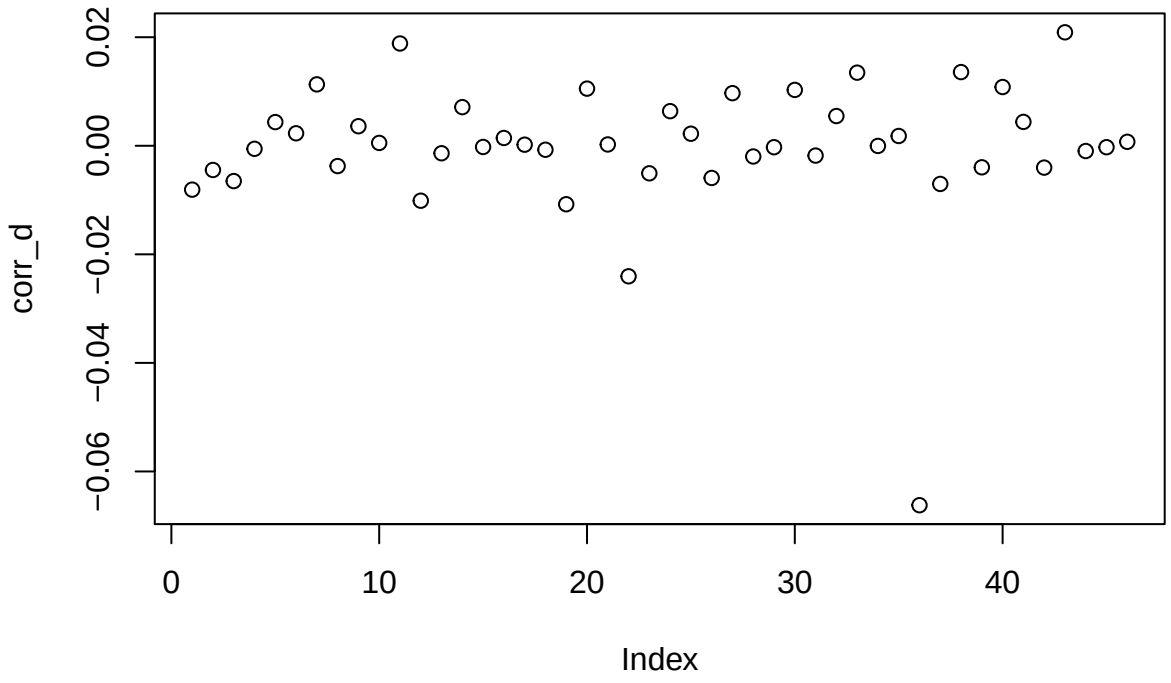
```
library(dplyr)
```

```
##
## 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
```

```
try <- function(i) {
  if (i == nrow(data)) data_loo <- data[0:(i-1),]
  else data_loo <- data[c(0:(i-1),(i+1):(nrow(data))),]
  return(corr(data_loo)-corr(data))
}
corr_d <- unlist(Map(try,1:nrow(data)))
data$corr_d <- corr_d
```

```
plot(corr_d)
```




```
data[order(corr_d),][1,]
```

```
##    liquor cirrhosis      corr_d
## 36    149     122.5 -0.06621551
```

We can see that the data (149,122.5) have particular influence in the analysis.