Homework5

Tianqi Wang

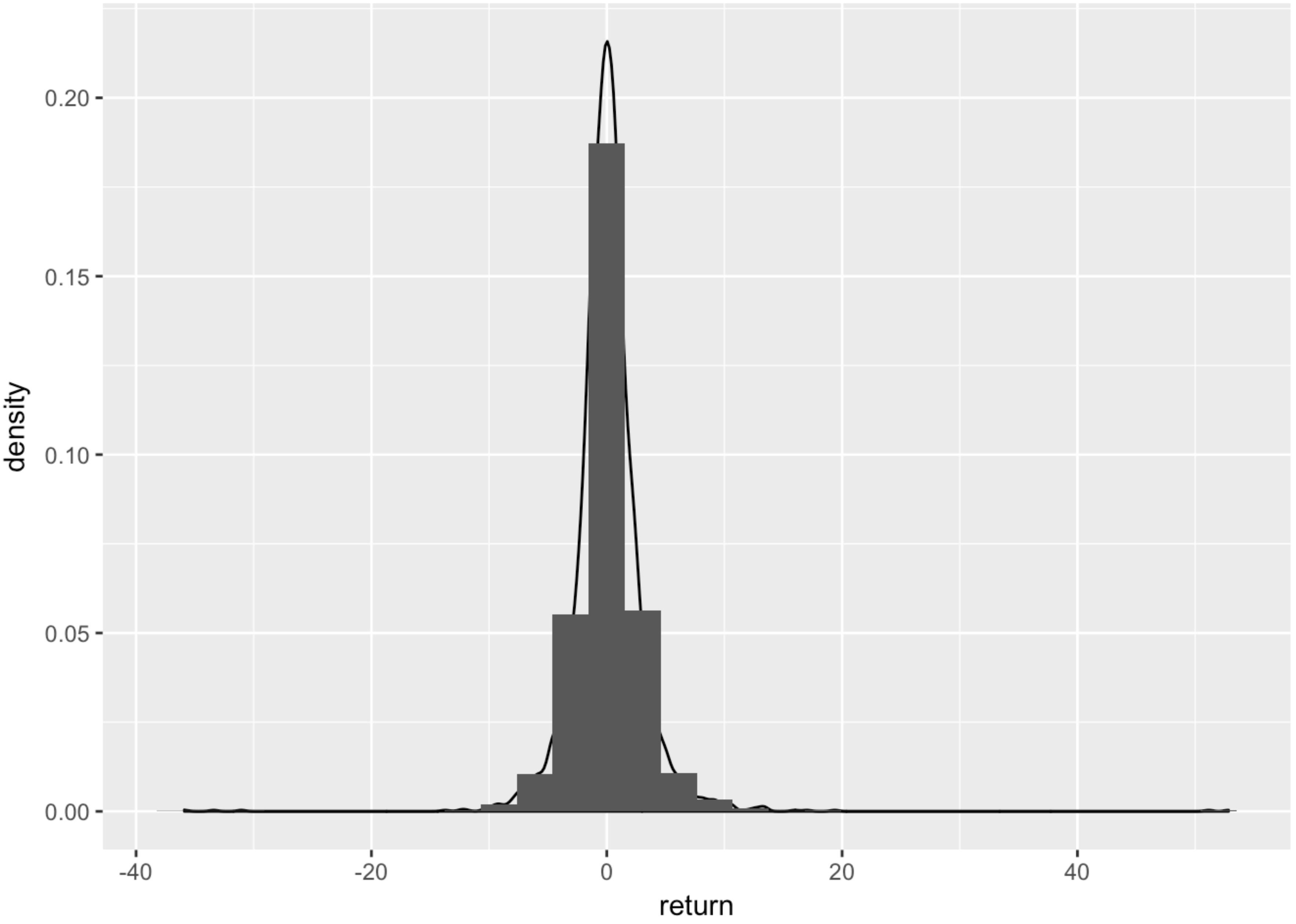
9/26/2018

Question 1

```
data1 <- read.csv("CAKEDailyReturns.csv")
return <- data1[, 1]
```

(a)

```
ggplot() + geom_density(aes(x = return, y = ..density..)) + geom_histogram(aes(x = re
turn,
    y = ..density..))
```

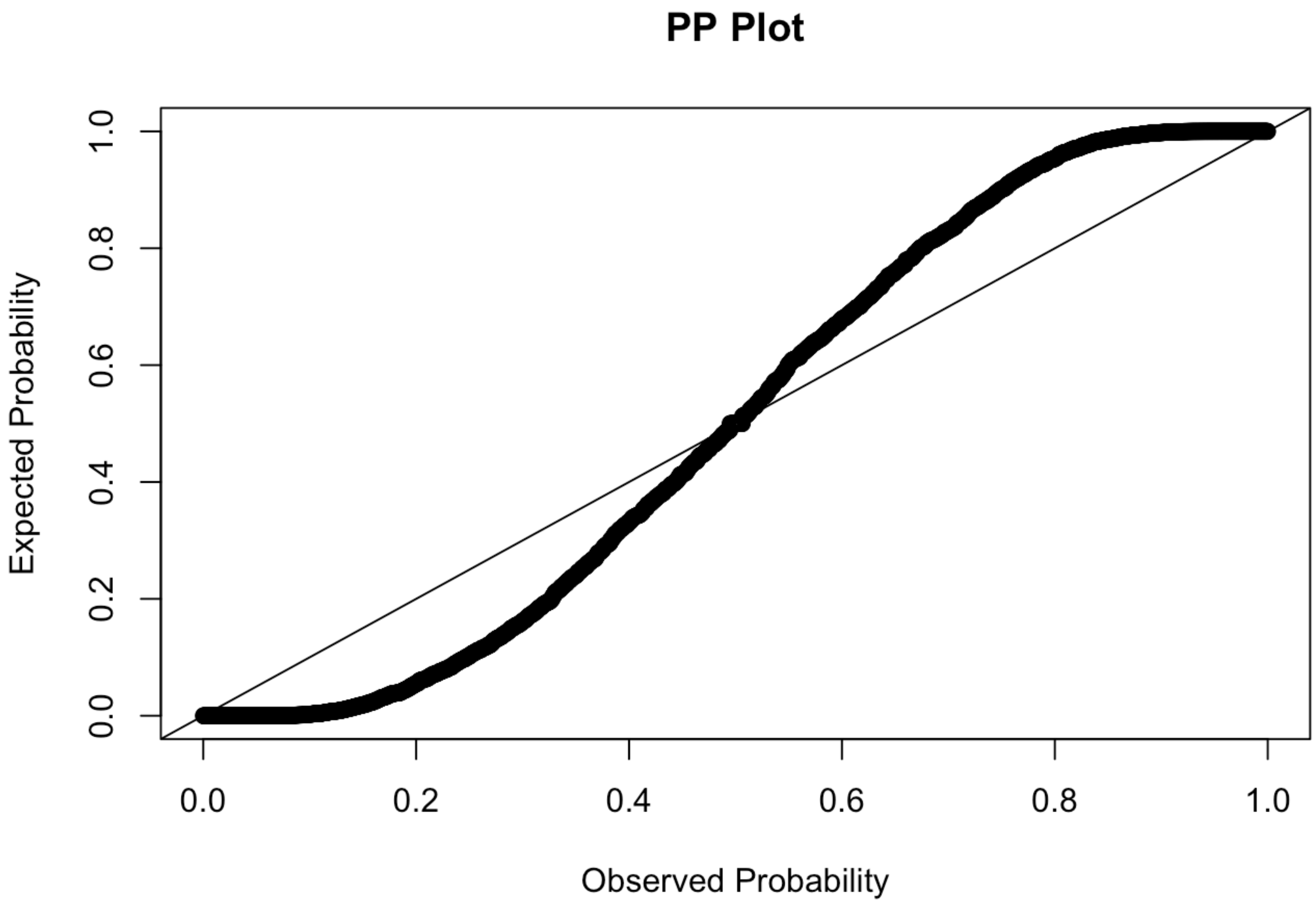


The picture shows the return data is not normal, it has more values clustered in the center, which is around 0, and more extreme value, it shows the characteristic of leptokurtic.

(b)

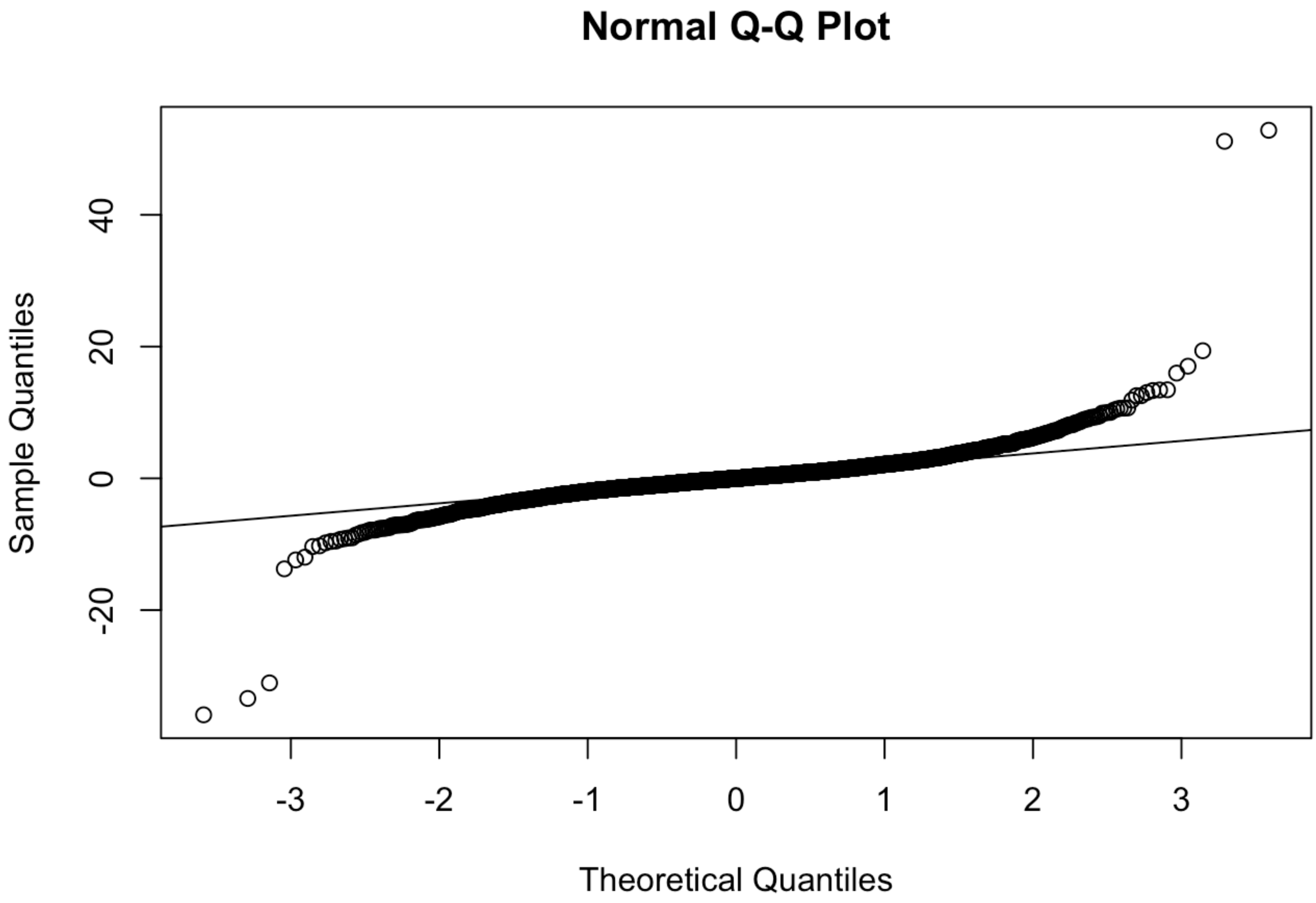
p-p plot

```
probDist <- pnorm(return)
# create PP plot
plot(ppoints(length(return)), sort(probDist), main = "PP Plot",
    xlab = "Observed Probability", ylab = "Expected Probability")
abline(0, 1)
```



q-q plot

```
qqnorm(return)
qqline(return)
```



Both p-p and q-q plots show the data is far from normal, both plots consistenly show the data has fat tails and also higher peaks, which are the characteristic of leptokurtic.

(c)

J-B test

```
jarque.bera.test(return)
```

```
##
##  Jarque Bera Test
##
## data:  return
## X-squared = 459160, df = 2, p-value < 0.00000000000000022
```

K-S test

```
ks.test(return, pnorm)
```

```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  return
## D = 0.15732, p-value < 0.00000000000000022
## alternative hypothesis: two-sided
```

For both J-B test and K-S test, the p-values are small, so the null hypothesis that the data is plausibly normal has been rejected.

Question 2

```
data2 <- read.csv("BreakfastCereal.csv")
data2$calories_per_cup <- data2$calories/data2$cups
data2$sugars_per_cup <- data2$sugars/data2$cups
data2$carbo_per_cup <- data2$carbo/data2$cups
data2$protein_per_cup <- data2$protein/data2$cups
data2$fat_per_cup <- data2$fat/data2$cups
data2$fiber_per_cup <- data2$fiber/data2$cups
data2$vitamin_per_cup <- data2$vitamin/data2$cups
model2 <- lm(calories_per_cup ~ sugars_per_cup + carbo_per_cup +
    protein_per_cup + fat_per_cup + fiber_per_cup + vitamin_per_cup +
    as.factor(mfr) + as.factor(type), rating, data = data2)
summary(model2)
```

```
##
## Call:
## lm(formula = calories_per_cup ~ sugars_per_cup + carbo_per_cup +
##     protein_per_cup + fat_per_cup + fiber_per_cup + vitamin_per_cup +
##     as.factor(mfr) + as.factor(type), data = data2, subset = rating)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -26.2894  -2.2311  -0.2917   4.0756  26.2894
##
## Coefficients:
##                   Estimate Std. Error t value             Pr(>|t|)
## (Intercept)      -65.80497   13.78602  -4.773  0.00001144968212699 ***
## sugars_per_cup     3.42309    0.25110  13.632 < 0.0000000000000002 ***
## carbo_per_cup      4.47283    0.20780  21.525 < 0.0000000000000002 ***
## protein_per_cup    8.77737    1.20847   7.263  0.00000000075243984 ***
## fat_per_cup        8.63106    0.82569  10.453  0.00000000000000262 ***
## fiber_per_cup     -0.42461    0.68676  -0.618             0.538658
## vitamin_per_cup   -0.01433    0.04090  -0.350             0.727321
## as.factor(mfr)G   52.74576   13.51899   3.902             0.000238 ***
## as.factor(mfr)K   50.77502   13.36106   3.800             0.000332 ***
## as.factor(mfr)N   31.66939   12.73251   2.487             0.015577 *
## as.factor(mfr)P   53.56083   13.60699   3.936             0.000212 ***
## as.factor(mfr)Q   70.41026   12.97947   5.425  0.00000101756121772 ***
## as.factor(mfr)R   47.21501   14.30955   3.300             0.001608 **
## as.factor(type)H  40.58804   10.12513   4.009             0.000166 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.579 on 62 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.9852, Adjusted R-squared:  0.9821
## F-statistic: 318.1 on 13 and 62 DF,  p-value: < 0.00000000000000022
```

J-B test

```
jarque.bera.test(model2$residuals)
```

```
##
##  Jarque Bera Test
##
## data:  model2$residuals
## X-squared = 15.425, df = 2, p-value = 0.0004473
```

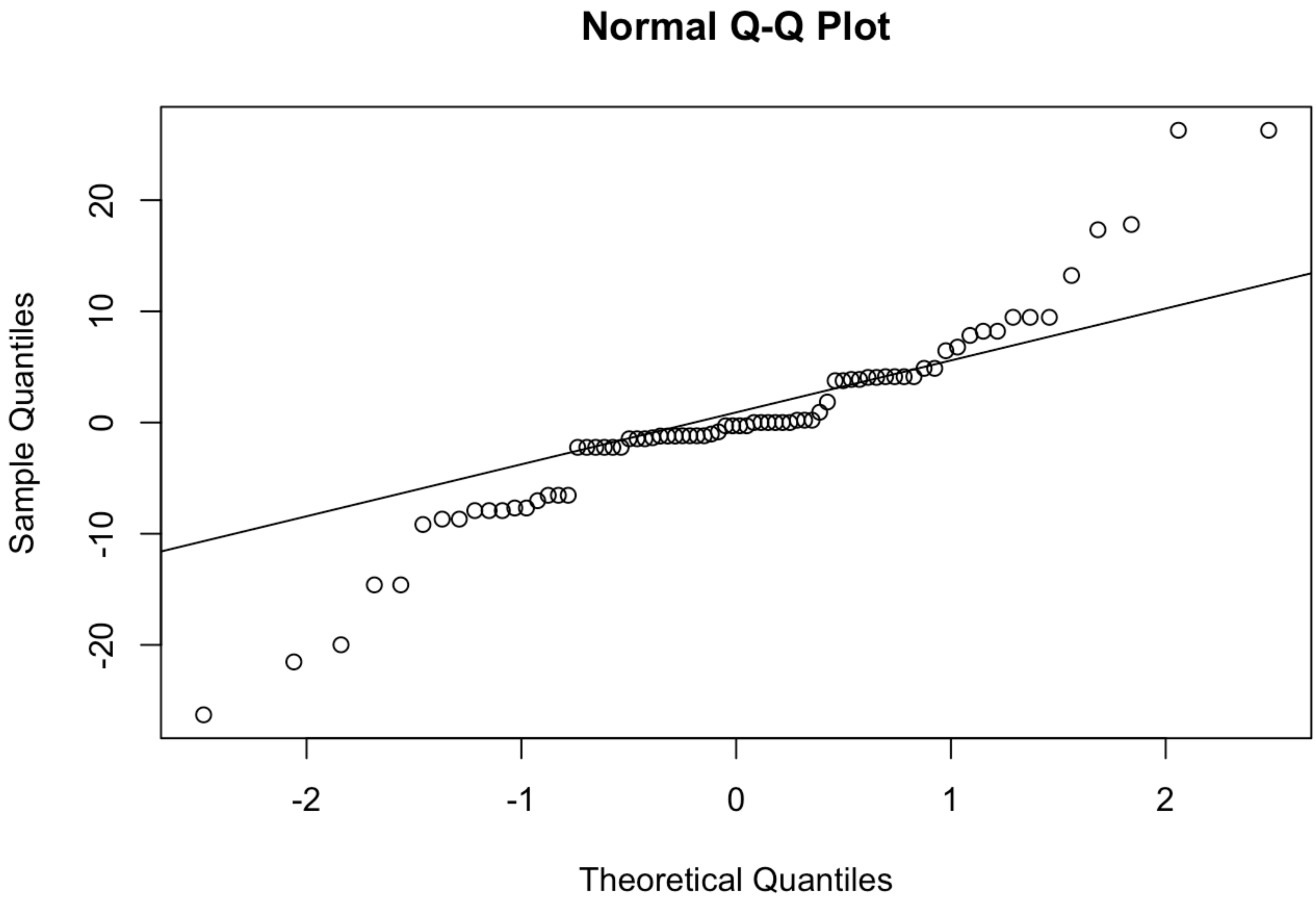
K-S test

```
ks.test(model2$residuals, pnorm)
```

```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  model2$residuals
## D = 0.33034, p-value = 0.0000001251
## alternative hypothesis: two-sided
```

P-values in both J-B test and K-S test are small, which is sufficient to reject the null hypothesis that residuals are plausibly normal, at significance level `\alpha = 0.01`.

```
res <- model2$residuals
qqnorm(res)
qqline(res)
```



From the q-q plot, it shows that the distribution of fitted residuals most stick to the referal line, except for several extreme values, so I think the deviation from normal is not severe, the rejection of null hypothesis in J-B test and K-S test are just caused by those leverage points.

Question 3

```
data3 <- read.csv("AlcoholAndLiverDisorder.csv")
model3 <- lm(GAMMAGT ~ MCV + ALKPHOS + SGPT + SGOT, data = data3)
summary(model3)
```

```
##
## Call:
## lm(formula = GAMMAGT ~ MCV + ALKPHOS + SGPT + SGOT, data = data3)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -72.839 -15.849  -5.628   5.981 255.403
##
## Coefficients:
##               Estimate Std. Error t value   Pr(>|t|)
## (Intercept) -115.21130   35.98880  -3.201    0.00150 **
## MCV            1.09742    0.40020   2.742    0.00643 **
## ALKPHOS        0.13558    0.09643   1.406    0.16063
## SGPT           0.50679    0.13329   3.802    0.00017 ***
## SGOT           1.20398    0.26215   4.593 0.00000617 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 32.42 on 340 degrees of freedom
## Multiple R-squared:  0.3258, Adjusted R-squared:  0.3178
## F-statistic: 41.07 on 4 and 340 DF,  p-value: < 0.0000000000000022
```

(a)

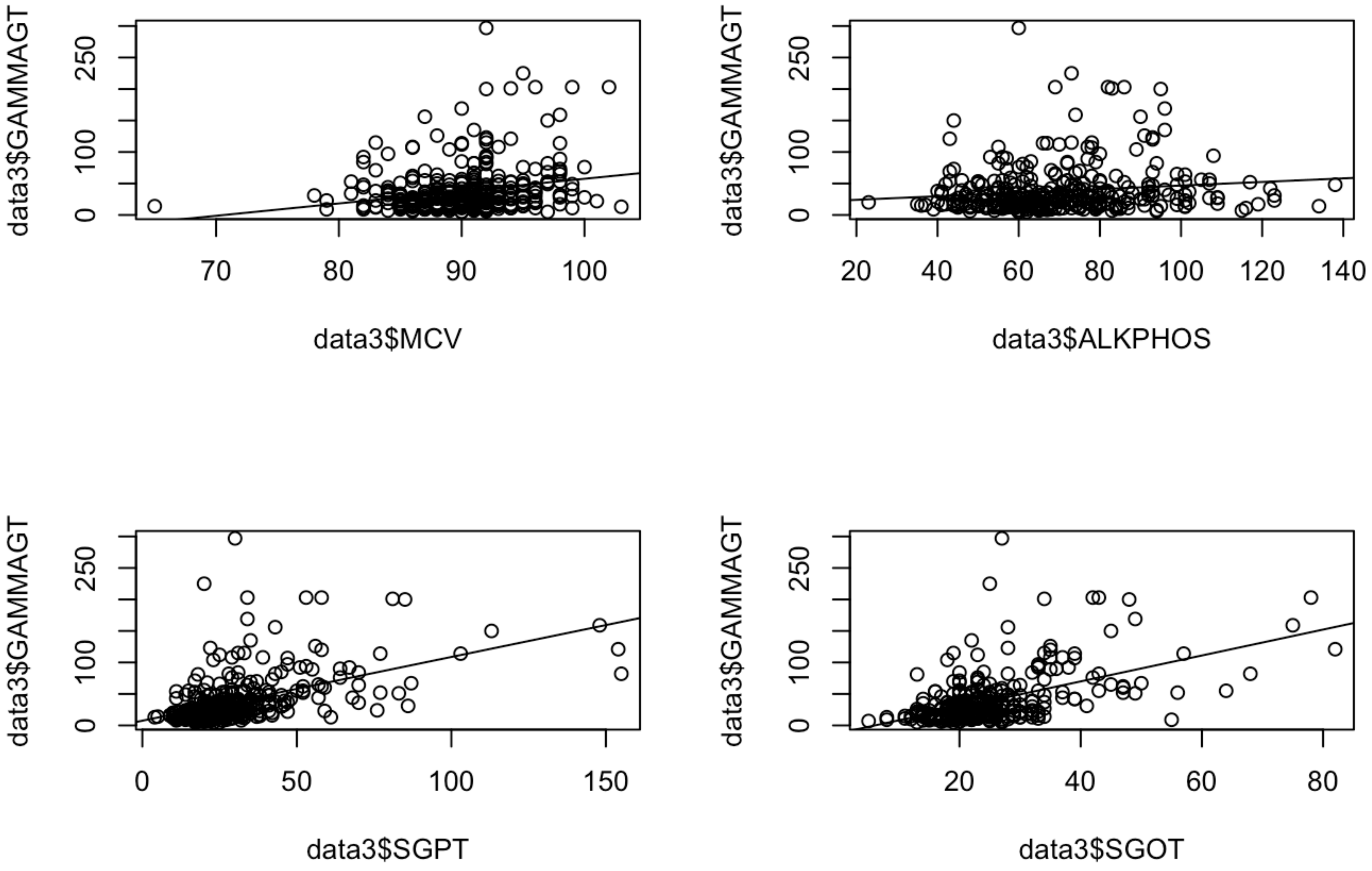
```
resettest(model3, power = 2:3, type = "regressor")
```

```
##
##  RESET test
##
## data:  model3
## RESET = 1.9817, df1 = 8, df2 = 332, p-value = 0.04807
```

Since the p-value is lower than 0.05, which means at the significance level$\alpha = 0.05$, it means the second power and third power of the regressors jointly have significant predictive power on dependent variable, so there could exist missing value problem and we might need to add the proper second and (or) third power of the regressors in the model.

(b)

```
par(mfrow = c(2, 2))
plot(data3$GAMMAGT ~ data3$MCV)
abline(lm(data3$GAMMAGT ~ data3$MCV))
plot(data3$GAMMAGT ~ data3$ALKPHOS)
abline(lm(data3$GAMMAGT ~ data3$ALKPHOS))
plot(data3$GAMMAGT ~ data3$SGPT)
abline(lm(data3$GAMMAGT ~ data3$SGPT))
plot(data3$GAMMAGT ~ data3$SGOT)
abline(lm(data3$GAMMAGT ~ data3$SGOT))
```



To check the linear relationship between GAMMAGT and the independent variables MCV, ALKPHOS, SGPT, and SGOT, I draw several graphs above, it seems that the relationships between GAMMAGT and MCV and SGOT are plausibly linear, while the relationships between GAMMAGT and ALKPHOS, and GMMAGT and SGPT are not.

(c)

I think the third power of `SGPT` should be included in our model, since their relationship seems not so linear.

(d)

```
model3.new <- lm(GAMMAGT ~ MCV + ALKPHOS + SGPT + SGOT + I(SGPT^3),
    data = data3)
summary(model3.new)
```

```
##
## Call:
## lm(formula = GAMMAGT ~ MCV + ALKPHOS + SGPT + SGOT + I(SGPT^3),
##     data = data3)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -64.856 -15.747  -5.363   6.483 253.500
##
## Coefficients:
##                 Estimate    Std. Error t value   Pr(>|t|)
## (Intercept) -123.42565542   35.67163430  -3.460   0.000609 ***
## MCV            1.11295667    0.39555276   2.814   0.005184 **
## ALKPHOS        0.09339877    0.09632366   0.970   0.332920
## SGPT           0.85678326    0.17553337   4.881 0.00000163 ***
## SGOT           1.25594417    0.25965663   4.837 0.00000200 ***
## I(SGPT^3)     -0.00002534    0.00000840  -3.017   0.002748 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 32.04 on 339 degrees of freedom
## Multiple R-squared:  0.3434, Adjusted R-squared:  0.3337
## F-statistic: 35.46 on 5 and 339 DF,  p-value: < 0.0000000000000022
```

```
resettest(model3.new)
```

```
##
##  RESET test
##
## data:  model3.new
## RESET = 1.872, df1 = 2, df2 = 337, p-value = 0.1554
```

After adding the third power of `SGPT` , the p-value of Ramsey RESET test increase a lot, which is reluctant to reject null hypothesis.

Question 4

When X_1 and X_2, X_2 and Y are both positive correlated or both negative correlated, then $\tilde{\beta}_1$ would be upward biased, if X_1 and X_2 are positive correlated and X_1 and Y are negative correlated, or X_1 and X_2 are negative correlated and X_1 and Y are positive correlated, then $\tilde{\beta}_1$ would be downsides biased. If there is no linear correlation between X_1 and X_2, or between X_1 and Y, then $\tilde{\beta}_1$ would be unbiased.

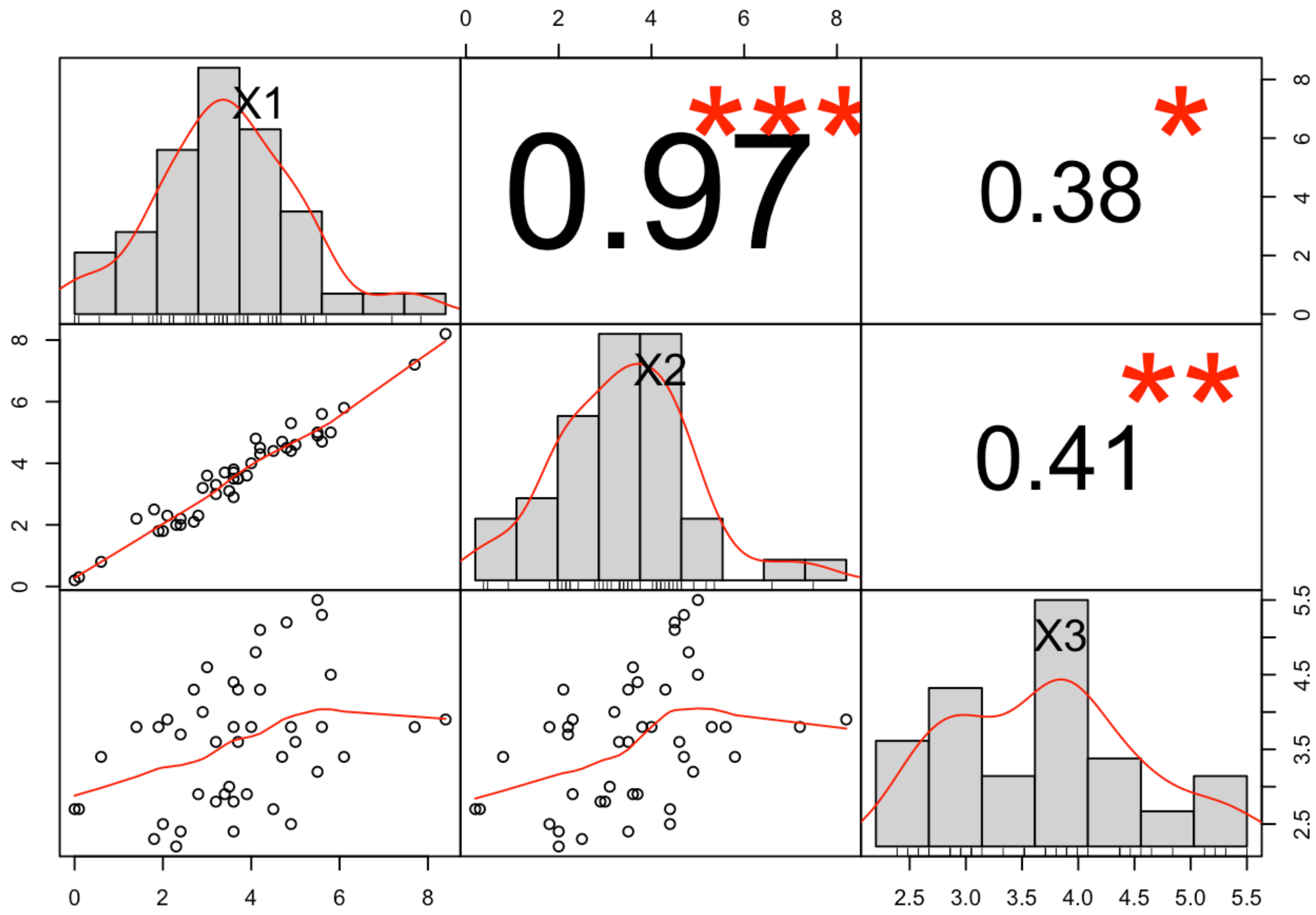
Question 5

```
data5 <- read.csv("CosmeticsSales.csv")
model5 <- lm(Y ~ X1 + X2 + X3, data = data5)
summary(model5)
```

```
## 
## Call:
## lm(formula = Y ~ X1 + X2 + X3, data = data5)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -5.4217 -0.9115  0.0703  1.1420  3.5479 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)  
## (Intercept)   1.0233     1.2029   0.851   0.4000  
## X1            0.9657     0.7092   1.362   0.1809  
## X2            0.6292     0.7783   0.808   0.4237  
## X3            0.6760     0.3557   1.900   0.0646 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 1.825 on 40 degrees of freedom
## Multiple R-squared:  0.7417, Adjusted R-squared:  0.7223 
## F-statistic: 38.28 on 3 and 40 DF,  p-value: 0.000000000007821
```

(a)

```
predictors <- data5[, -1]
chart.Correlation(predictors)
```



From the graph above, we can see the correlation coefficient between X_1 and X_2 is significant large (close to 1), also combined from the graph, we can observe there is strong linear correlation between X_1 and X_2, so multicollinearity problem exists in our model.

(b)

```
vif(model5)
```

```
##        X1        X2        X3
## 20.072031 20.716101  1.217973
```

The VIF values of X_1 and X_2 are quite large (much larger than 10), so it also shows there is evident multicollinearity problem for these two predictor variables.

Question 6

```
data6 <- read.csv("BreakfastCereal.csv")
model6 <- lm(calories ~ sugars + carbo + protein + fat + fiber +
    vitamins + as.factor(mfr) + as.factor(type) + rating, data = data6)
```

(a)

```
summary(model6)
```

```
##
## Call:
## lm(formula = calories ~ sugars + carbo + protein + fat + fiber +
##     vitamins + as.factor(mfr) + as.factor(type) + rating, data = data6)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -17.846  -2.837   0.000   2.673  17.846
##
## Coefficients:
##                  Estimate Std. Error t value            Pr(>|t|)
## (Intercept)      38.17324   17.43060   2.190            0.032294 *
## sugars            1.93188    0.42358   4.561 0.00002458400649562 ***
## carbo             2.98030    0.28416  10.488 0.00000000000000229 ***
## protein           5.98468    0.98767   6.059 0.00000008831892050 ***
## fat               6.37897    1.18333   5.391 0.00000115759253745 ***
## fiber             1.04606    0.64504   1.622            0.109943
## vitamins         -0.02758    0.04007  -0.688            0.493799
## as.factor(mfr)G  19.43955    8.87298   2.191            0.032229 *
## as.factor(mfr)K  25.47072    8.65851   2.942            0.004585 **
## as.factor(mfr)N  27.35301    8.22755   3.325            0.001490 **
## as.factor(mfr)P  26.23029    8.84705   2.965            0.004294 **
## as.factor(mfr)Q  21.35204    8.84352   2.414            0.018725 *
## as.factor(mfr)R  22.51882    8.86464   2.540            0.013595 *
## as.factor(type)H 23.95281    5.62517   4.258 0.00007113486769782 ***
## rating           -0.82468    0.20154  -4.092            0.000126 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.312 on 62 degrees of freedom
## Multiple R-squared:  0.9144, Adjusted R-squared:  0.895
## F-statistic: 47.29 on 14 and 62 DF,  p-value: < 0.000000000000000022
```

(b)

For testing the multicollinearity problem, I will check the VIF for each predictor variables.

```
vif(model6)
```

```
##                       GVIF Df GVIF^(1/(2*Df))
## sugars            6.760994  1        2.600191
## carbo             2.819853  1        1.679242
## protein           2.229978  1        1.493311
## fat               2.705377  1        1.644803
## fiber             4.507888  1        2.123179
## vitamins          1.528969  1        1.236515
## as.factor(mfr)    8.170363  6        1.191297
## as.factor(type)   2.289469  1        1.513099
## rating           15.287387  1        3.909909
```

Then from the tables I notice there are several predictors whose VIF are quite large, 3 of them are larger than 5, 1 is larger than 10, which could show some evidence for the existence of multicollinearity problem.

(c)

Firstly, I drop the variable who has highest VIF value `rating` , and re-run the regression and then check the VIF again.

```
model6.new1 <- lm(calories ~ sugars + carbo + protein + fat +
    fiber + vitamins + as.factor(mfr) + as.factor(type), data = data6)
summary(model6.new1)
```

```
##
## Call:
## lm(formula = calories ~ sugars + carbo + protein + fat + fiber +
##     vitamins + as.factor(mfr) + as.factor(type), data = data6)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -22.5036  -3.4968  -0.0337   3.5643  22.5036
##
## Coefficients:
##                   Estimate Std. Error t value             Pr(>|t|)
## (Intercept)      -18.59257   11.79825  -1.576             0.120063
## sugars             3.35178    0.27157  12.342 < 0.0000000000000002 ***
## carbo              3.14745    0.31439  10.011    0.000000000000012 ***
## protein            4.46722    1.02341   4.365    0.000048182555192 ***
## fat                9.39329    1.03532   9.073    0.000000000000485 ***
## fiber             -0.70995    0.53839  -1.319             0.192063
## vitamins           0.02837    0.04211   0.674             0.503015
## as.factor(mfr)G   33.68036    9.12495   3.691             0.000468 ***
## as.factor(mfr)K   37.70143    9.08500   4.150             0.000102 ***
## as.factor(mfr)N   32.10141    9.10633   3.525             0.000795 ***
## as.factor(mfr)P   38.43741    9.31158   4.128             0.000110 ***
## as.factor(mfr)Q   33.17564    9.34428   3.550             0.000734 ***
## as.factor(mfr)R   35.30143    9.27477   3.806             0.000322 ***
## as.factor(type)H  30.20670    6.05228   4.991    0.000005022294883 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.057 on 63 degrees of freedom
## Multiple R-squared:  0.8912, Adjusted R-squared:  0.8688
## F-statistic: 39.72 on 13 and 63 DF,  p-value: < 0.00000000000000022
```

```
vif(model6.new1)
```

```
##                     GVIF Df GVIF^(1/(2*Df))
## sugars          2.223515  1        1.491145
## carbo           2.761581  1        1.661800
## protein         1.915587  1        1.384047
## fat             1.656901  1        1.287207
## fiber           2.512579  1        1.585112
## vitamins        1.350942  1        1.162300
## as.factor(mfr)  5.439519  6        1.151587
## as.factor(type) 2.120455  1        1.456178
```

Seems the VIF decrease a lot after dropping variable `rating` , then we continue to drop another variable whos VIF larger than 5, which is `mfr` .

```
model6.new2 <- lm(calories ~ sugars + carbo + protein + fat +
    fiber + vitamins + as.factor(type), data = data6)
summary(model6.new2)
```

```
##
## Call:
## lm(formula = calories ~ sugars + carbo + protein + fat + fiber +
##     vitamins + as.factor(type), data = data6)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -21.493  -3.084  -0.655   3.632  33.527
##
## Coefficients:
##                  Estimate Std. Error t value            Pr(>|t|)
## (Intercept)      17.13852    6.09147   2.814             0.00638 **
## sugars            3.44197    0.26114  13.180 < 0.0000000000000002 ***
## carbo             3.08493    0.28110  10.974 < 0.0000000000000002 ***
## protein           4.61053    1.10631   4.167     0.00008785071615 ***
## fat               8.71076    1.01386   8.592     0.00000000000166 ***
## fiber            -0.52532    0.50771  -1.035             0.30443
## vitamins          0.02843    0.04317   0.659             0.51240
## as.factor(type)H 16.80581    5.34896   3.142             0.00247 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.755 on 69 degrees of freedom
## Multiple R-squared:  0.8562, Adjusted R-squared:  0.8416
## F-statistic: 58.67 on 7 and 69 DF,  p-value: < 0.00000000000000022
```

```
vif(model6.new2)
```

```
##          sugars           carbo         protein             fat
##        1.702540        1.828160        1.853665        1.315751
##           fiber        vitamins as.factor(type)
##        1.850248        1.175521        1.371514
```

Then VIF for every predictor variables are smaller than 2 now, so the final model is `calories~sugars+carbo+protein+fat+fiber+vitamins+as.factor(type)` .

(d)

```
model6.new3 <- lm(calories ~ sugars + carbo + protein + fat +
    fiber + vitamins + as.factor(mfr) + as.factor(type) + rating,
    data = data6)
x <- model.matrix(model6.new3, data6)[, -1]
y <- data6$calories
lambdas <- 10^seq(3, -2, by = -0.1)
cv_fit <- cv.glmnet(x, y, alpha = 0, nfolds = 5, lambda = lambdas)
coef(cv_fit, s = "lambda.min")
```

```
## 15 x 1 sparse Matrix of class "dgCMatrix"
##                             1
## (Intercept)      67.961929099
## sugars            1.660016507
## carbo             2.243596208
## protein           5.017778217
## fat               6.531355912
## fiber             0.168441096
## vitamins          0.009029553
## as.factor(mfr)G  -1.147991913
## as.factor(mfr)K   4.883326999
## as.factor(mfr)N   5.970070326
## as.factor(mfr)P   5.021852055
## as.factor(mfr)Q  -2.167208846
## as.factor(mfr)R   3.196470861
## as.factor(type)H 12.251247190
## rating           -0.658458035
```

```
coeffcients <- data.frame(OLS = model6$coefficients, Ridge = coef(cv_fit,
    s = "lambda.min")[, 1])
kable(coeffcients)
```

	OLS	Ridge
(Intercept)	38.1732390	67.9619291
sugars	1.9318806	1.6600165
carbo	2.9802975	2.2435962
protein	5.9846759	5.0177782
fat	6.3789738	6.5313559
fiber	1.0460633	0.1684411
vitamins	-0.0275848	0.0090296
as.factor(mfr)G	19.4395547	-1.1479919
as.factor(mfr)K	25.4707167	4.8833270

as.factor(mfr)N	27.3530142	5.9700703
as.factor(mfr)P	26.2302939	5.0218521
as.factor(mfr)Q	21.3520367	-2.1672088
as.factor(mfr)R	22.5188154	3.1964709
as.factor(type)H	23.9528148	12.2512472
rating	-0.8246810	-0.6584580

From the table above, we can see there is not much change for those variables has small vifs, but for those variable who has large vifs like `manufacturer` , the coefficient decrease a lot.

Question 7

Three symptoms (or signs) of multicollinearity:

- There are only a few coefficients that have t-static significant rejecting the null hypothesis, but the F-test for the regression is highly significant.
- When adding and dropping variables, there are huge changes to the magnitude and sometimes even the sign of fitted coefficients.
- If the VIF of some variables are large (>5), there could exist multicollinearity problem in the model.

Three possible solutions:

- Drop one or more obnoxious variables.
- Use ridge or lasso regression.
- Collect additional data, which reduces the total impact on $Var(\hat{\beta}_j)$.