Assignment 2

Sherida van den Bent, Chang Liu, Kai Zhang

2020/03/10

Exercise 1

a) In this experiment, we have two factors: environment and humidity. Environment has a fixed level of $I = 3$, and humidity has a fixed level of $J = 2$. Furthermore, we have 18 experimental units, and because balanced design is a common choice, we choose $N = 18 \div I \div J = 3$. We use 1, 2, 3 in the first row (the environment) to represent cold, intermediate, and warm respectively. In the second row (humidity), we use 1 and 2 to represent dry and wet. The third row is the unit index, ranging from 1 to 16. Each column represents a combination of environment, humidity, and experimental unit.

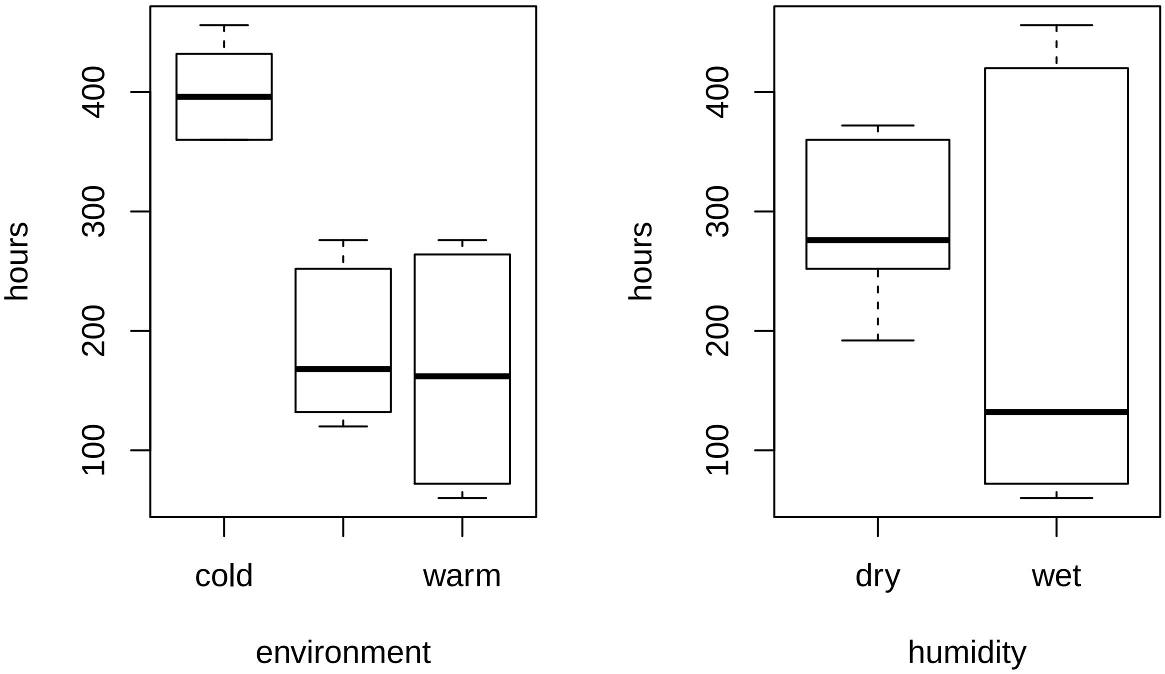
```
I=3; J=2; N=3
rbind(rep(1:I,each=N*J),rep(1:J,N*I),sample(1:(N*I*J)))
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12] [,13]
## [1,]    1    1    1    1    1    1    2    2    2     2     2     2     3
## [2,]    1    2    1    2    1    2    1    2    1     2     1     2     1
## [3,]   16    5   12   15    9   17    6    4    2     7    18    10     8
##      [,14] [,15] [,16] [,17] [,18]
## [1,]     3     3     3     3     3
## [2,]     2     1     2     1     2
## [3,]    11    14    13     3     1
```

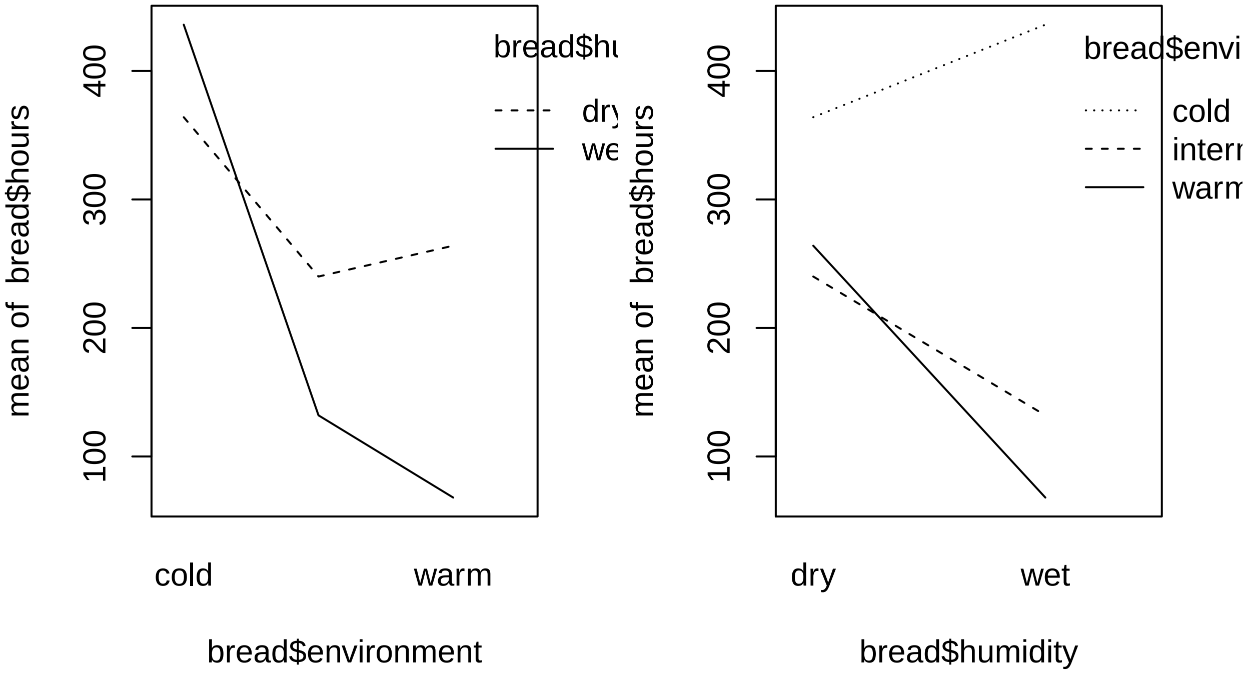
For unit 16 we use levels (‘cold’,‘dry’); for unit 5 we use levels (‘cold’,‘wet’); . . .; for unit 1 we use levels (‘warm’,‘wet’).

b)

```
bread = read.table('bread.txt')
par(mfrow=c(1,2))
boxplot(hours~environment, data=bread); boxplot(hours~humidity, data=bread);
```



```
interaction.plot(bread$environment,bread$humidity,bread$hours); interaction.plot(bread$humidity,bread$en
```



From the box plot, we can see that `intermediate` and `warm` have a similar decay time, and `cold` can greatly increase decay time. Futhermore, the median time value of `dry` is bigger than the median time value of `wet`. From the interaction lines we can see that there is an interaction between the ennvironment and the humidity.

c) In this analysis there is no mixed effect, so only fixed effects analysis will be conducted.

```
bread = read.table('bread.txt')
bread$environment=factor(bread$environment); bread$humidity=factor(bread$humidity)
breadlm=lm(hours~environment*humidity, data=bread);
anova(breadlm)
```

```
## Analysis of Variance Table
##
## Response: hours
##                      Df Sum Sq Mean Sq F value    Pr(>F)
## environment           2 201904  100952 233.685 2.461e-10 ***
## humidity              1  26912   26912  62.296 4.316e-06 ***
## environment:humidity  2  55984   27992  64.796 3.705e-07 ***
## Residuals            12   5184     432
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

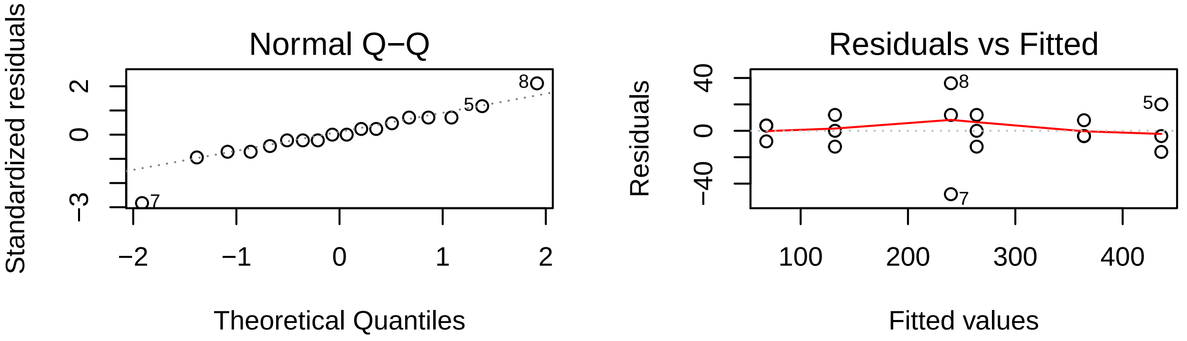
Our interest is in the main and interaction effects of the outer and inner factor. The main effects for environment and humidity are significant, at the same time the p-value of testing: $H_0 : \gamma_{i,j} = 0$ for all (i, j) is 3.705e-07, which means that interaction effect is highly significant; in other words, the bread decay times are related to the effects between the humidity and the environment. If the humidity does not change, but the environment changes, then the decay time changes as well. If the environment does not change, but the humidity changes, so does the decay time.

d) This is not a good question. The reason for this is as follows. From the summary shown in the previous question we also can see interaction effect is highly significant; we should NOT interpret the main effects without considering the interaction effect.

e)

```
par(mfrow=c(2,2))
plot(breadlm, 2); plot(breadlm, 1)
shapiro.test(residuals(object = breadlm))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(object = breadlm)
## W = 0.9296, p-value = 0.1911
```



First we need to check the assumption: normal distribution of the model residuals. After creating a QQ-plot, we cannot tell whether it is a normal distribution because of some outliers. Therefore we also use the Shapiro-Wilk normality test. Based on the p-value(0.1911 > 0.05), we could say that it is probably normally distributed. Our conclusion: the first assumption of normal distribution of the model residuals has been met. The fitted plot seems to indicate that the residuals and the fitted values are uncorrelated. Our conclusion: the second assumption of homogeneity of variance of the groups has been met. There are three outliers showed in the plot: NO.5, NO.7 and NO.8.

Exercise 2

a) In this experiment, we have two factor interface and skill. The treatment factor interface has a fixed level of $I = 3$, and the block variable skill has a fixed level of $B = 5$. Also, we have 15 experimental units, and because of balanced design, we choose $N = 15 \div I \div J = 1$.

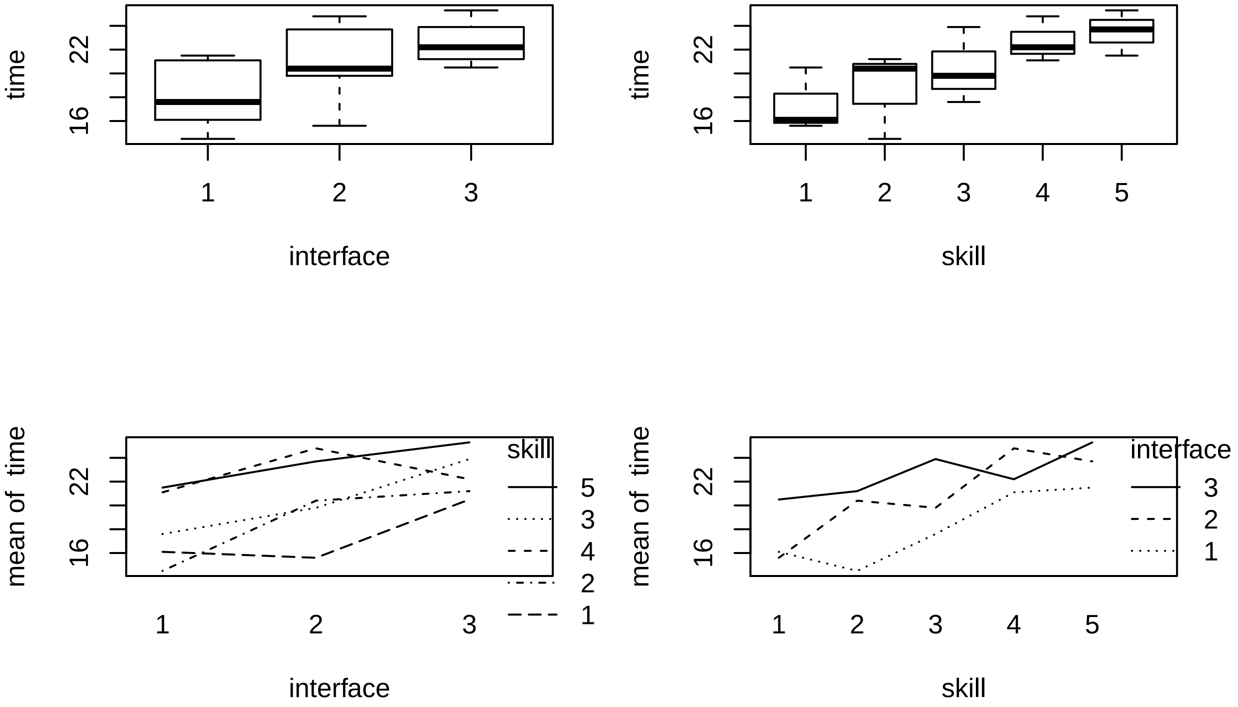
```
I=3; B=5; N=1
for (i in 1:B) print(sample(1:15)[(I*(i-1) + 1):(I*(i-1) + 3)])
```

```
## [1]  4 15  5
## [1] 12  3  7
## [1] 14  5  9
## [1] 15  5  1
## [1] 11  1 13
```

For block 1 we assign unit 4 to treatment 1, unit 15 to treatment 2, etc., for block 2 assign unit 12 to treatment 1, unit 3 to treatment 2, etc.

b)

```
search = read.table('search.txt')
search$skill=as.factor(search$skill); search$interface=as.factor(search$interface);
attach(search); par(mfrow=c(2,2))
boxplot(time~interface); boxplot(time~skill);
interaction.plot(interface,skill,time); interaction.plot(skill,interface,time);
```



From looking at the boxplot, we can see that factors do have some effect. From looking at the interaction plot, the lines in interaction plot are roughly parallel; therefore we can say there are no interactions between interface and skill.

c)

```
searchaov=lm(time~interface+skill, data=search)
anova(searchaov)
```

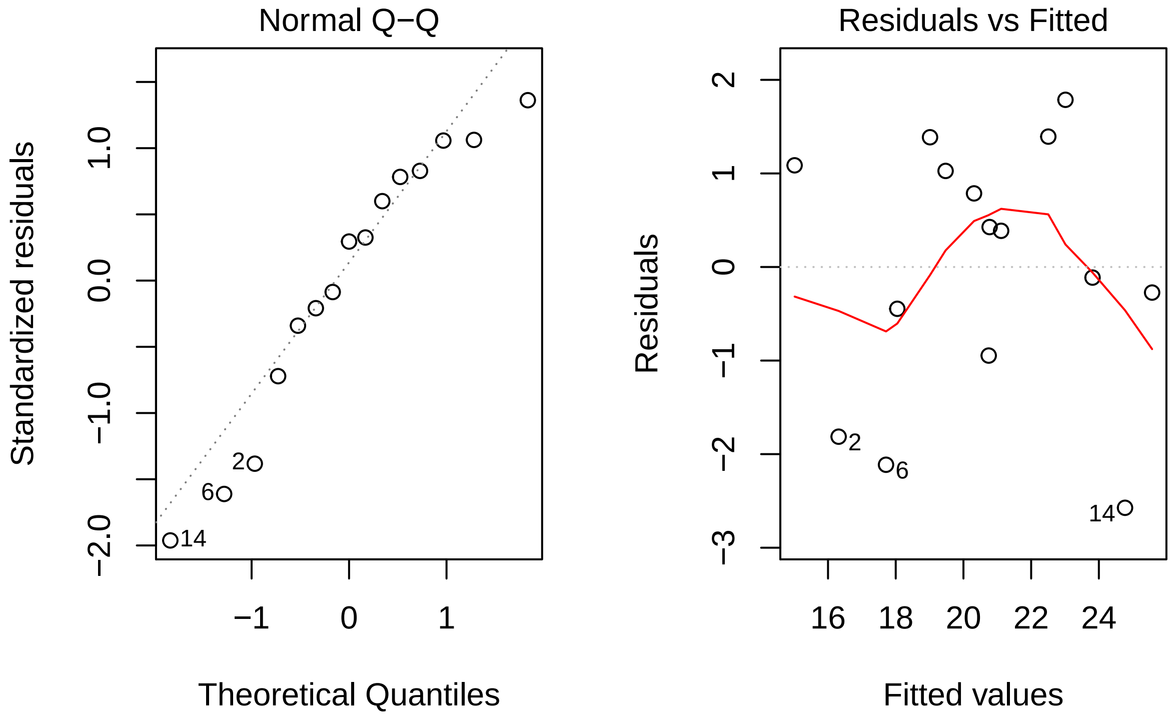
The p-value for testing $H_0 : \alpha_i = 0$ for all (i) is 0.01310. Conclusion: reject H_0 and search time is not the same for all interfaces.

```
summary(searchaov)
15.013+2.700+3.033
```

The estimate value for $\hat{\mu}$=15.013, $\hat{\alpha_2}$=2.700, $\hat{\beta}_3$=3.033. So the the estimated time for the given factors is $\hat{\mu_{23}} = \hat{\mu} + \hat{\alpha_2} + \hat{\beta}_3$=20.746.

d)

```r
par(mfrow=c(1,2)); plot(searchaov, 2); plot(searchaov, 1)
```



```r
shapiro.test(residuals(object = searchaov))
```

First we need to check the assumption: normal distribution of the model residuals. After creating a QQ-plot, we cannot tell whether it is normal distribution because of some outliers. So we also use the Shapiro-Wilk normality test. Based on the p-value(0.2817 $>$ 0.05), we could say it probably normal distributed. Conclusion: the first assumption of normal distribution of the model residuals has been met. The fitted plot seems to indicate that the residuals and the fitted values are uncorrelated. Conclusion: the second assumption of homogeneity of variance of the groups has been met.

e)

```r
attach(search); friedman.test(time, interface, skill)[[3]]
```

```
## The following objects are masked from search (pos = 3):
##
##     interface, skill, time
```

```
## [1] 0.0407622
```

p-value for testing H_0 : no treatment effect is 0.04076, so H_0 is rejected, there is an effect of interface.

f)

```r
searchoneaov = lm(time ~ interface, data = search)
anova(searchoneaov)
```

```
## Analysis of Variance Table
##
## Response: time
##           Df  Sum Sq Mean Sq F value  Pr(>F)
## interface  2  50.465  25.233  2.8605 0.09642 .
## Residuals 12 105.852   8.821
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

It is not useful to perform this test on this dataset, because randomized block design is to make the variability within blocks less than the variability between blocks, and this design reduces variability within treatment conditions and potential confounding, producing a better estimate of treatment effects. Since we already gathered enough data, we should apply the randomized block design instead of one-way ANOVA.

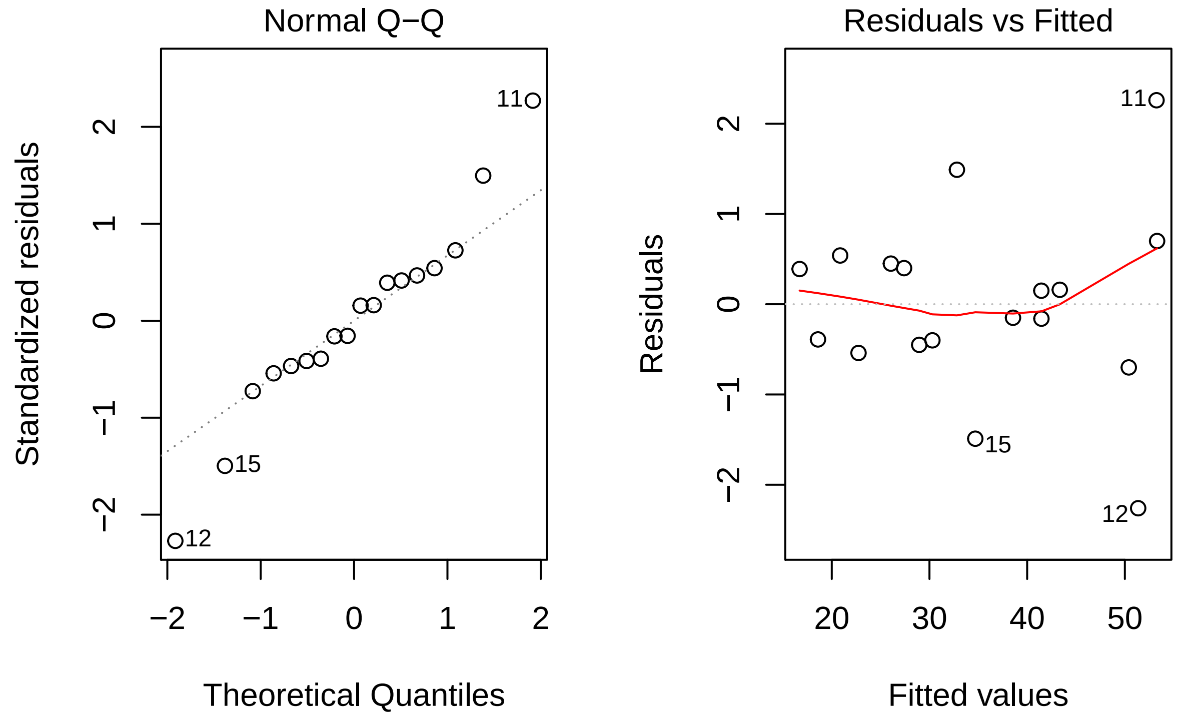
Exercise 3

a)

```
cow=read.table("cow.txt",header=TRUE,sep=" ");
cow$id=factor(cow$id); cow$per=factor(cow$per)
cowlm=lm(milk~treatment+per+id,data=cow); anova(cowlm)
```

```
## Analysis of Variance Table
##
## Response: milk
##           Df  Sum Sq Mean Sq  F value    Pr(>F)
## treatment  1    0.27   0.269   0.1085   0.75147
## per        1   25.39  25.387  10.2462   0.01505 *
## id         8 2467.47 308.434 124.4832 7.494e-07 ***
## Residuals  7   17.34   2.478
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
par(mfrow=c(1,2)); plot(cowlm, 2); plot(cowlm, 1)
```



Before making any conclusion, we should check the assumption of ANOVA. The result seems fine, and we can rely on the result. The sequence effect is left out, because it cannot be estimated in a fixed effects model. We do not have enough information to estimate all effects as fixed effects from the available data. In the mixed effects model this is possible.

```
summary(cowlm)
```

```
##
## Call:
## lm(formula = milk ~ treatment + per + id, data = cow)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.2600 -0.4375  0.0000  0.4375  2.2600
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  30.3000     1.2444  24.349 5.02e-08 ***
## treatmentB   -0.5100     0.7466  -0.683 0.516536
## per2         -2.3900     0.7466  -3.201 0.015046 *
## id2          23.0000     1.5741  14.612 1.68e-06 ***
## id3          11.1500     1.5741   7.084 0.000196 ***
## id4          -1.3500     1.5741  -0.858 0.419480
## id5          -7.0500     1.5741  -4.479 0.002870 **
## id6          23.4500     1.5741  14.898 1.47e-06 ***
## id7          13.5500     1.5741   8.608 5.69e-05 ***
## id8           4.9000     1.5741   3.113 0.017011 *
## id9         -11.2000     1.5741  -7.115 0.000191 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.574 on 7 degrees of freedom
## Multiple R-squared:  0.9931, Adjusted R-squared:  0.9832
## F-statistic: 100.6 on 10 and 7 DF,  p-value: 1.349e-06
```

The p-value for treatment is 0.75147. There is no evidence that feedingstuffs are different and the negative feedingstuffs gives 0.5100 less milk production. Furthermore, the p-value of testing: $H_0 : \gamma_{psn} = 0$ for all (psn) is 0.015046, which means learning (=per) effect is highly significant.

b)

```
library(lme4); attach(cow)
cowlmer=lmer(milk~treatment+order+per+(1|id),REML=FALSE)
summary(cowlmer)
```

```
## Linear mixed model fit by maximum likelihood  ['lmerMod']
## Formula: milk ~ treatment + order + per + (1 | id)
##
##      AIC      BIC   logLik deviance df.resid
##    119.3    124.7    -53.7    107.3       12
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.53112 -0.37104  0.02686  0.26748  1.72489
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  id       (Intercept) 133.145  11.539
##  Residual               1.927   1.388
## Number of obs: 18, groups:  id, 9
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)  38.5000     5.8110   6.625
## treatmentB   -0.5100     0.6585  -0.775
## orderBA      -3.4700     7.7685  -0.447
## per2         -2.3900     0.6585  -3.630
##
## Correlation of Fixed Effects:
##            (Intr) trtmnB ordrBA
## treatmentB -0.063
## orderBA    -0.743  0.000
## per2       -0.063  0.111  0.000
```

The estimated treatment and period effects under fixed effects are identical to those in the previous `lm()`'s results. The negative feedingstuffs gives 0.5100 less milk production.

```
cowlmer1=lmer(milk~order+per+(1|id),REML=FALSE)
anova(cowlmer1,cowlmer)
```

```
## Data: NULL
## Models:
## cowlmer1: milk ~ order + per + (1 | id)
## cowlmer: milk ~ treatment + order + per + (1 | id)
##          Df    AIC    BIC  logLik deviance  Chisq Chi Df Pr(>Chisq)
## cowlmer1  5 117.89 122.34 -53.946   107.89
## cowlmer   6 119.31 124.65 -53.656   107.31 0.5807      1      0.446
```

As illusstrated for treatment, there is no significicant effect.

c)

```
attach(cow)
```

```
## The following objects are masked from cow (pos = 3):
##
##     id, milk, order, per, treatment
```

```
t.test(milk[treatment=="A"],milk[treatment=="B"],paired=TRUE)
```

```
##
##  Paired t-test
##
## data:  milk[treatment == "A"] and milk[treatment == "B"]
## t = 0.22437, df = 8, p-value = 0.8281
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -2.267910  2.756799
## sample estimates:
## mean of the differences
##               0.2444444
```

No, it is not a valid test for a difference in milk production. This is because when we use t-test for seeing effects, we assume that there is no order effect; but we previously concluded that there is an order effect. Its conclusion is compatible with the one obtained in a, because the result in a) is fixed effect analysis, it does not consider the order effect so does `t.test()` in this question. So the p-value for treatment is identical to the reuslt found in a). The p-value for id is not interesting.

Exercise 4

a)

```
nausea = read.table('nauseatable.txt')
df = data.frame(matrix(0,304,2))
names(df) <- c("naus", "medicin")
df[1:180, 1] = 1; df[181: 304, 1] = 2
df[1:100, 2] = 1; df[101:132, 2] = 2; df[133:180, 2] = 3
df[181:232, 2] = 1; df[233:267, 2] = 2; df[268:304, 2] = 3
xtabs(~medicin+naus, data=df)
```

```
##          naus
## medicin    1   2
##        1 100  52
##        2  32  35
##        3  48  37
```

We use number 1 in column "naus" to indicate "Incidence of no nausea", number 2 to indicate "Incidence of Nausea". Number 1 in column 'medicin' to indicate "Chlorpromazine" and 2, 3 for "Pentobarbital(100mg)", "Pentobarbital(150mg)" respectively. xtabs are a convenient function to creat contingency table, so the result is same as the data in "nauseatable.txt".

b)

```
B=1000
tstar=numeric(B)
for (i in 1:B) {
  treatstar=df
  treatstar[,2] = sample(df[,2])
  tstar[i] = chisq.test(xtabs(~medicin+naus, data = treatstar))[[1]]
}
myt = chisq.test(xtabs(~medicin+naus, data = df))[[1]]
pr=sum(tstar>myt)/B
pr
```

```
## [1] 0.032
```

The p-value for testing $H_{0}: $the different medicines work equally well against nausea is 0.031. Conclusion: we reject H_0 and we have confidence that there exists difference between the different medicines.

c)

```
chisq.test(xtabs(~medicin+naus, data = df))[[3]]
```

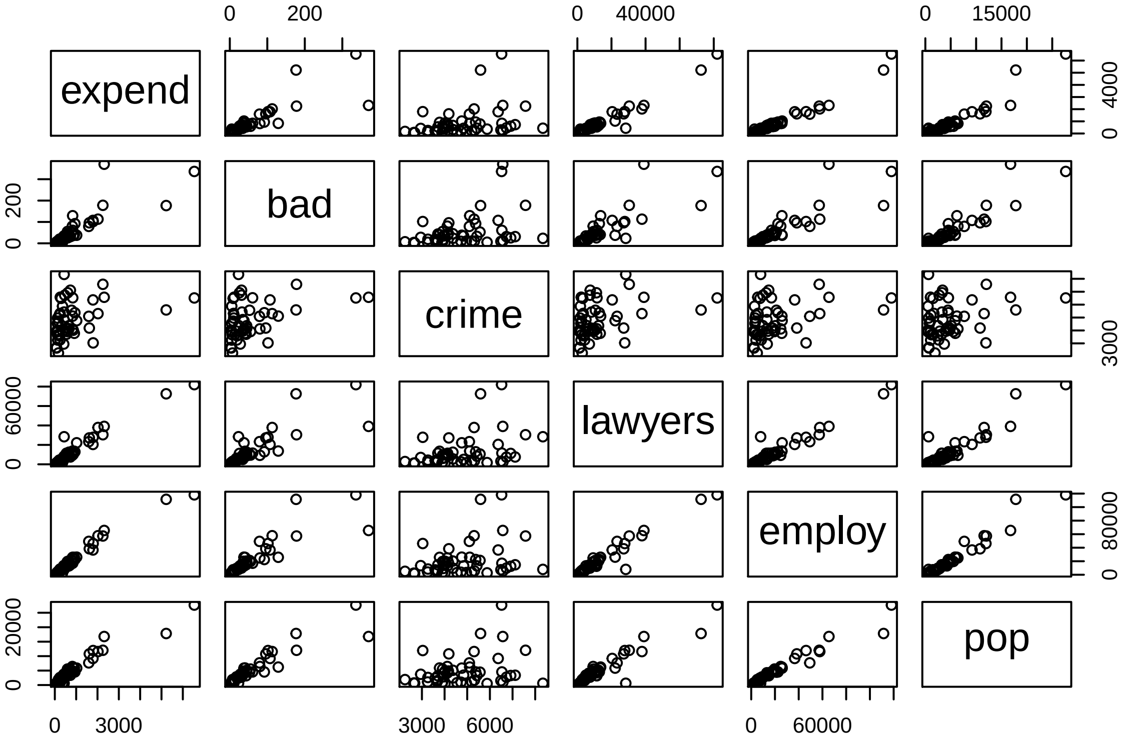
```
## [1] 0.03642928
```

The p-value from permutation test and chi-square test for contingency tables are very close.

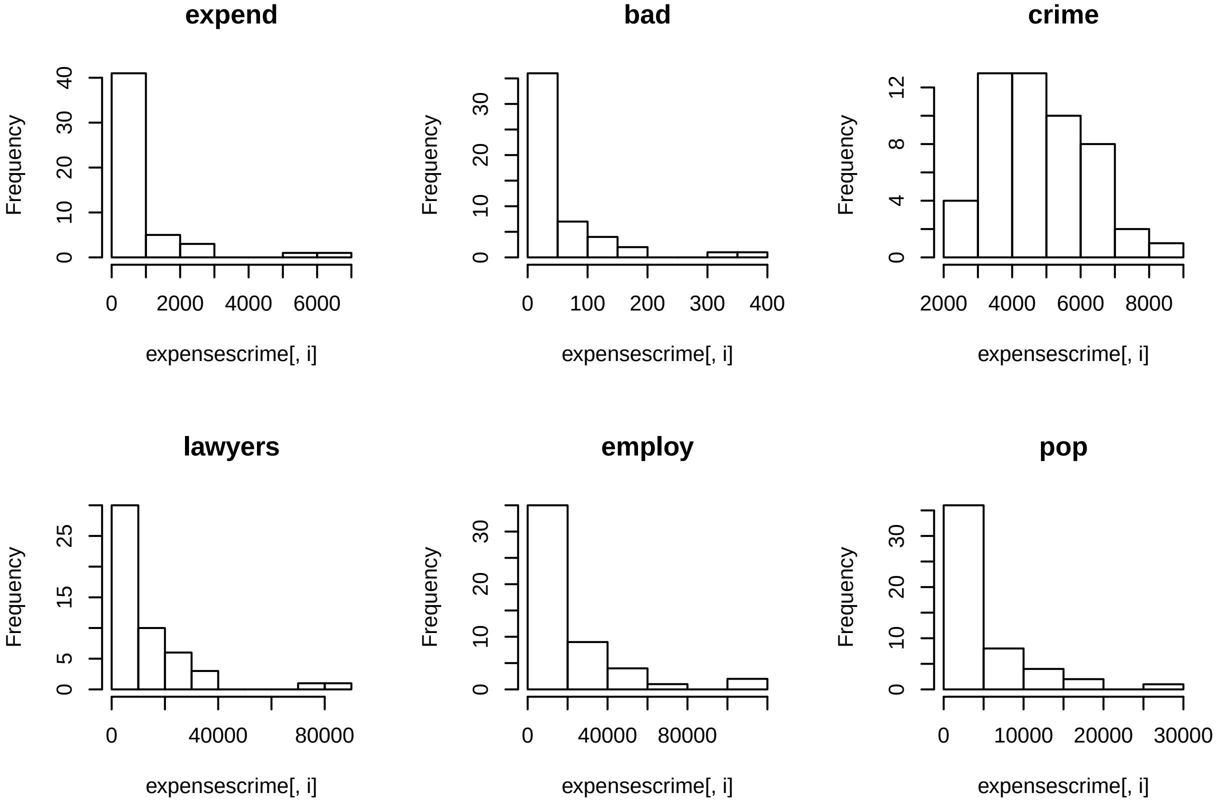
Exercise 5

a) 5.a graphical summaries

```
expensescrime=read.table("expensescrime.txt",header=TRUE,sep=" ");
plot(expensescrime[,c(2,3,4,5,6,7)]);par(mfrow=c(2,3));
```



```r
for (i in c(2,3,4,5,6,7)) hist(expensescrime[,i],main=names(expensescrime)[i])
```

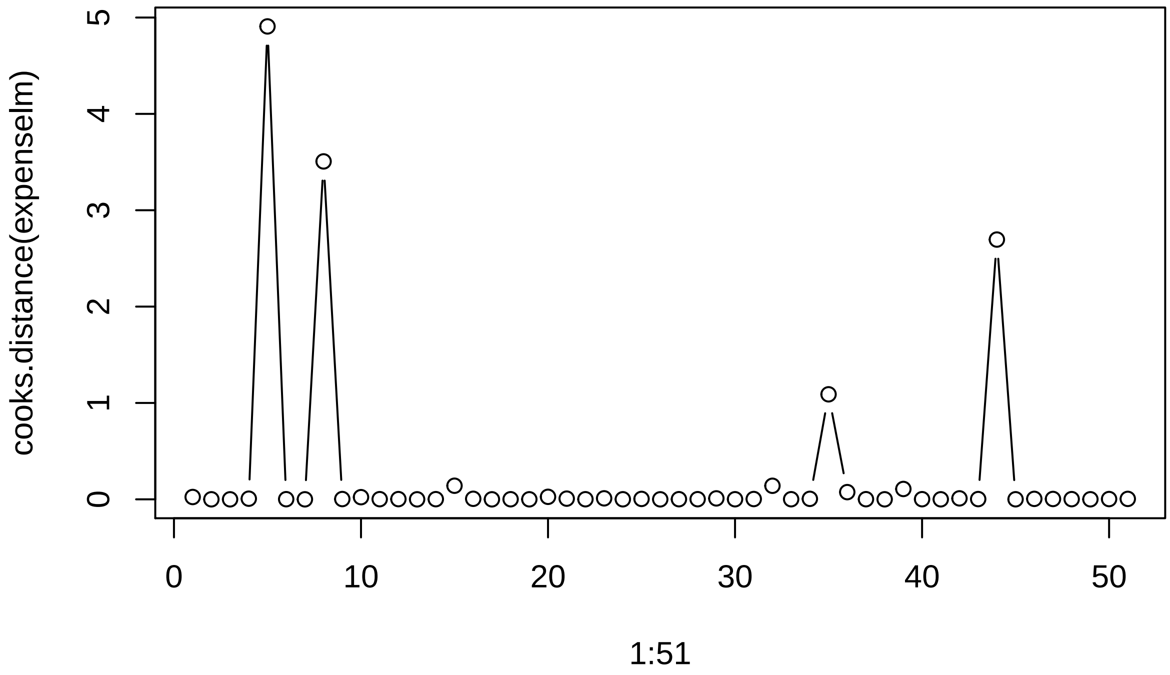


From the histogram graphs, we can see that the features `bad`, `lawyers`, `employ`, `pop` have similar shapes, so there is a highly collinearity problem in linear regression.

Potential points A potential point is an observation with an outlying value in an explanatory variable X_i. As it shows in the histogram, there are some extreme values in the `bad`, `lawyers`, `employ`, `pop` columns when X is very big, so it has potential points.

Influence points We will use Cook's distance and draw the corresponding plot chart to investigate influence points between expense data and other data. If the Cook's distance is bigger than 1, then it is considered to be an influence point.

```
expenselm=lm(expend~bad+crime+lawyers+employ+pop, data=expensescrime);
plot(1:51,cooks.distance(expenselm),type="b");
```



```
cooks.distance(expenselm)
```

According to model expense~bad+crime+lawyers+employ+pop, we got 4 inluence points in the following way: * Point 5 is an influence point, it's Cook distance is 4.91 $>$ 1. * Point 8 is an influence point, it's Cook distance is 3.51 $>$ 1. * Point 35 is an influence point, it's Cook distance is 1.09 $>$ 1. * Point 44 is an influence point, it's Cook distance is 2.70 $>$ 1.

Collinearity We will use the variance inflation factor (VIF) to investigate potential collinearity in the dataset, for the β_j in model Y_n=$\beta_0 + \beta_1 X_{n1} + \beta_2 X_{n2} + \ldots + \beta_p X_{np} + e_n, n = 1, 2, \ldots, N$,if the β_j's VIF is bigger than 5, then it is considered β_j is unreliable.

```
expensescrimelm=lm(expend~bad+crime+lawyers+employ+pop, data=expensescrime)
vif(expensescrimelm)
```

```
##       bad     crime   lawyers    employ       pop
##  8.364321  1.487978 16.967470 33.591361 32.937517
```

```
expensescrimelm2=lm(expend~crime+lawyers+employ+pop, data=expensescrime)
vif(expensescrimelm2)
```

```
##     crime   lawyers    employ       pop
##  1.233263 16.372292 33.106158 17.576977
```

```
expensescrimelm3=lm(expend~crime+employ+pop, data=expensescrime)
vif(expensescrimelm3)
```

```
##     crime    employ       pop
##  1.121163 17.967808 17.568906
```

```
expensescrimelm4=lm(expend~crime+pop, data=expensescrime)
vif(expensescrimelm4)
```

```
##   crime     pop
## 1.08213 1.08213
```

```
expensescrimelm5=lm(expend~crime, data=expensescrime)
```

In the 1st model to 3rd model, only 1 out of 5 VIF is lower than 5, so there are collinearity problems in those models. The last 2 models are ok with respect to collinearity problems.

b) We use the summary function, and its 8th output is the R^2 determination coefficient. We don't print all the results, but the p-value here is equality important. First we use the step-up method.

```
summary(lm(expend~bad,data=expensescrime))[[8]]
summary(lm(expend~crime,data=expensescrime))[[8]]
summary(lm(expend~lawyers,data=expensescrime))[[8]]
summary(lm(expend~pop,data=expensescrime))[[8]]
```

```
summary(lm(expend~employ,data=expensescrime))[[8]]
```

```
## [1] 0.9539745
```

The model expend~employ has max determination coefficient: 0.954, so we chose this model for the next step.

```
summary(lm(expend~employ+bad,data=expensescrime))[[8]]
summary(lm(expend~employ+crime,data=expensescrime))[[8]]
summary(lm(expend~employ+pop,data=expensescrime))[[8]]
```

```
summary(lm(expend~employ+lawyers,data=expensescrime))[[8]]
```

```
## [1] 0.9631745
```

In those four models only `lawyers` in expend~employ+lawyers is significant (p-value=0.00113 < 0.05), and it has determination coefficient 0.9632, larger than 0.954, so we chose this model for thenext step.

```
summary(lm(expend~employ+lawyers+bad,data=expensescrime))[[8]]
summary(lm(expend~employ+lawyers+crime,data=expensescrime))[[8]]
summary(lm(expend~employ+lawyers+pop,data=expensescrime))[[8]]
```

All of those newly added features yield insignificant explanatory variables, so we can stop and take the model expend~employ+lawyers as our final step-up model. Then we use the step-up method.

```
summary(lm(expend~bad+crime+lawyers+employ+pop,data=expensescrime))
```

Feature `crime` has the largest p-value 0.25534, and it is larger than 0.05, so we remove crime from the model.

```
summary(lm(expend~bad+lawyers+employ+pop,data=expensescrime))
```

Feature pop has the largest p-value 0.06012, and it is larger than 0.05, so we remove pop from the model.

```
summary(lm(expend~bad+lawyers+employ,data=expensescrime))
```

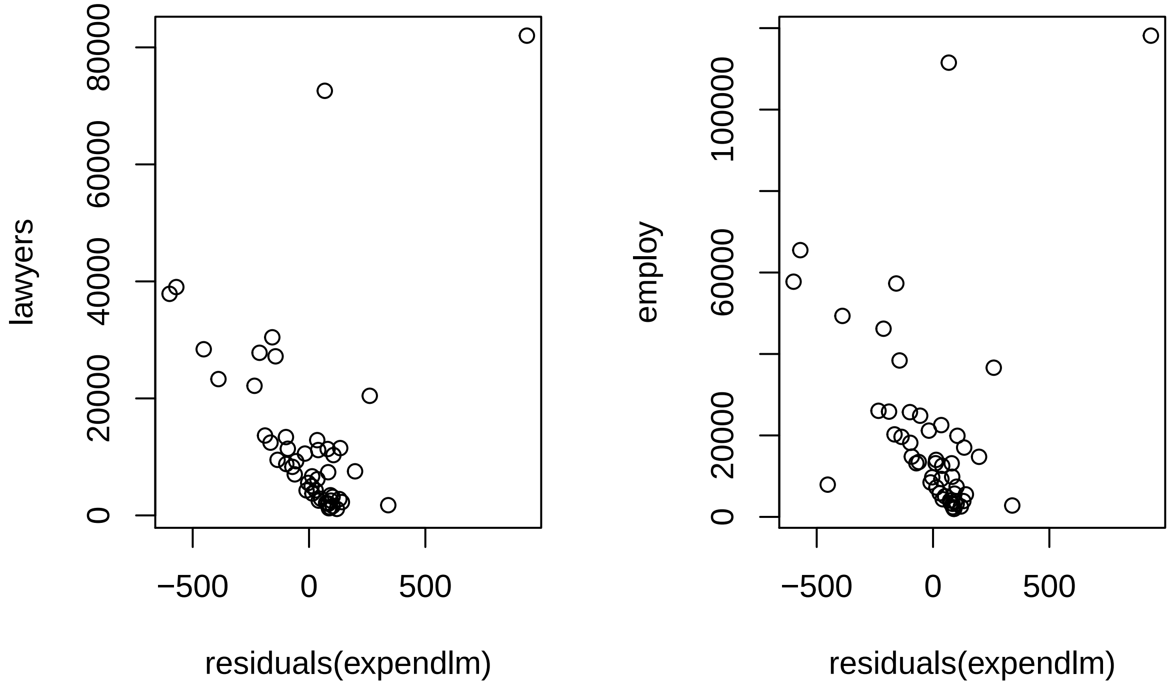
Feature bad has the largest p-value 0.34496, and it is larger than 0.05, so we remove bad from the model.

```
summary(lm(expend~lawyers+employ,data=expensescrime))
```

All remaining explanatory variables in the model are significant, so we can stop and take model expend~employ+lawyers as our final step-up model. Both method generate the same model: expend~employ+lawyers.

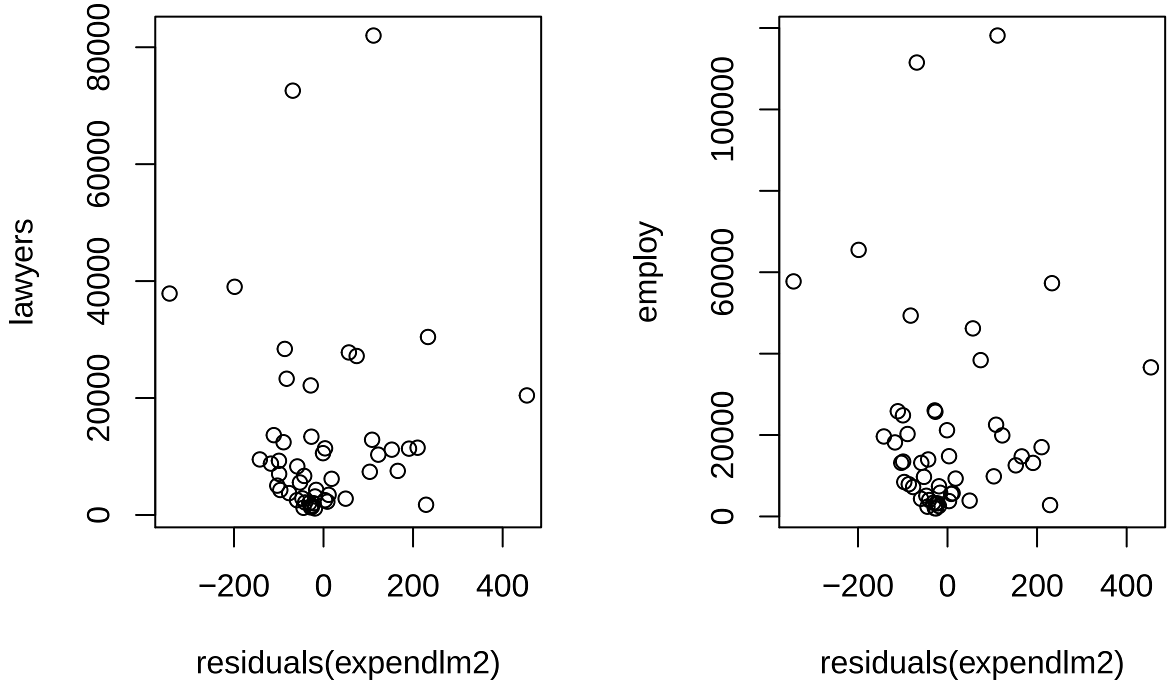
c) (1)Scatter plot: plot residuals against each X_k in the model separately

```
attach(expensescrime); expendlm=lm(expend~lawyers+employ)
par(mfrow=c(1,2)); plot(residuals(expendlm),lawyers); plot(residuals(expendlm),employ)
```



There are V curve shown in two charts, include $lawyers^2$ and $employ^2$

```
expensescrime$lawyers2=expensescrime$lawyers^2
expensescrime$employ2=expensescrime$employ^2
expendlm2=lm(expend~expend+lawyers+lawyers2+employ+employ2,data=expensescrime)
par(mfrow=c(1,2)); plot(residuals(expendlm2),lawyers); plot(residuals(expendlm2),employ)
```



```
summary(expendlm2)
```

```
##
## Call:
## lm(formula = expend ~ expend + lawyers + lawyers2 + employ +
##     employ2, data = expensescrime)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -343.80  -72.72  -25.93   53.04  454.15
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.749e+01  3.514e+01   1.067  0.29164
## lawyers     -2.605e-02  9.632e-03  -2.704  0.00957 **
## lawyers2     1.056e-06  2.321e-07   4.550 3.92e-05 ***
## employ       4.947e-02  7.341e-03   6.740 2.24e-08 ***
## employ2     -3.166e-07  1.148e-07  -2.757  0.00833 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 132.4 on 46 degrees of freedom
## Multiple R-squared:  0.9886, Adjusted R-squared:  0.9876
## F-statistic: 994.7 on 4 and 46 DF,  p-value: < 2.2e-16
```

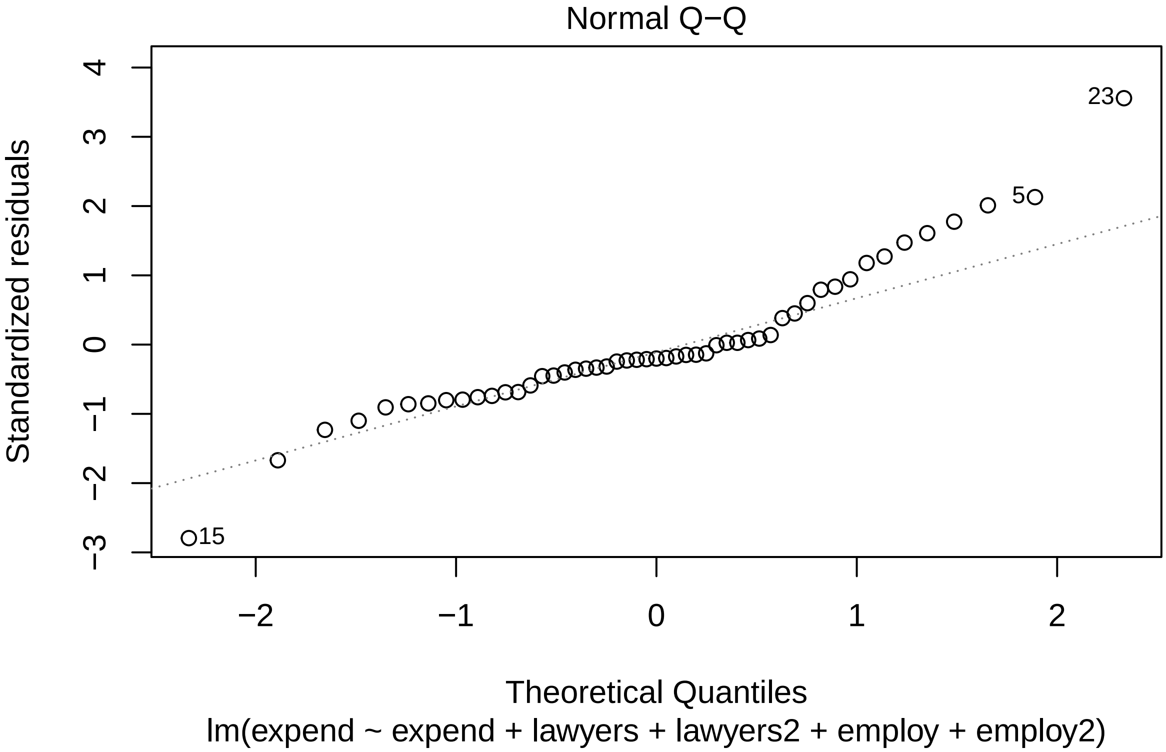
There is no specific curved pattern visible, good!

(2)normal QQ-plot of the residuals

```
plot(expendlm2, 2)
```

```
## Warning in model.matrix.default(object, data = structure(list(expend =
## c(360L, : the response appeared on the right-hand side and was dropped
```

```
## Warning in model.matrix.default(object, data = structure(list(expend =
## c(360L, : problem with term 1 in model.matrix: no columns are assigned
```



```
shapiro.test(residuals(expendlm2))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(expendlm2)
## W = 0.91566, p-value = 0.001464
```

From looking at the QQ-plot we can not see if it is a normal distribution. We did the Shapiro-Wilk normality test, and we got p-value $0.001464 < 0.05$, which means it is normally distributed.