Stat 415/615, Lab 4, Multiple Linear Regression (II)

Jun Lu

Stat 415/615 Regression, 2023

Contents

Comments and explanations are not included here. We'll discuss them in class.

1 Example 1. Commercial Property (polynomial and interactions)

A commercial real estate company evaluates vacancy rates, square footage, rental rates and operating expense for commercial properties in a large metropolitan area in order to provide clients with quantitative information upon which to make rental decisions. The data below are taken from 82 suburban commercial properties that are the newest, best located, most attractive and expensive for five specific geographic areas. How here are the rental rates (Y), age (X1), operating expenses and taxes (X2), vacancy rates (X3), total square footage(X4). Data are available on Blackboard in file `CommercialProperty.txt` and `CommericalPreperty.sav`.

Case_i	RentalRates	Age	Expense	Vacancy	Sfootage
1	13.500	1	5.02	0.14	123000
...	...	...	...	...	...
81	14.500	14	12.68	0.03	201930

```
cpdata<-read.table("../DataSets/CommercialProperty.txt", header=T)
#cpdata
head(cpdata, 3)  # first 3 observations
```

```
##   RentalRates Age Expense Vacancy Sfootage
## 1        13.5   1    5.02    0.14   123000
## 2        12.0  14    8.19    0.27   104079
## 3        10.5  16    3.00    0.00    39998
```

```
tail(cpdata, 3)  # last 3 observations
```

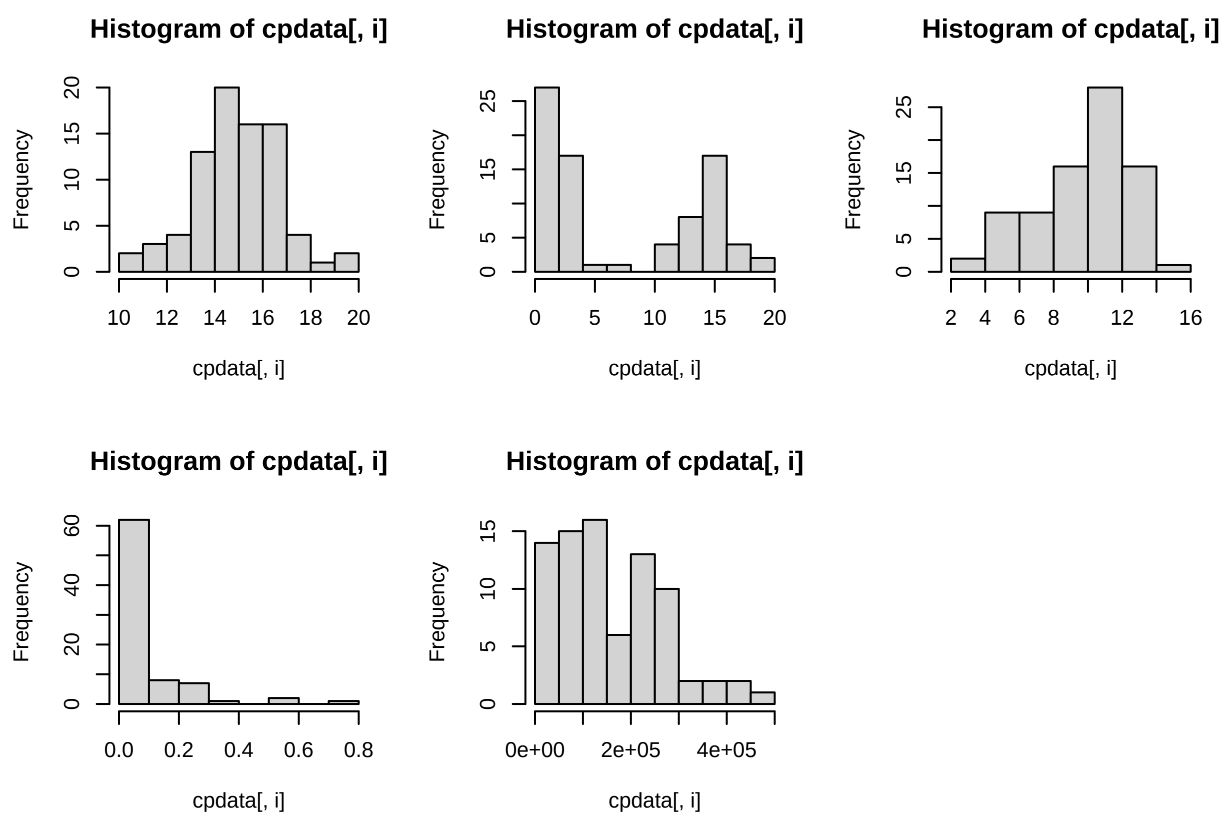
```
##    RentalRates Age Expense Vacancy Sfootage
## 79       15.00  15   11.97    0.14   254700
## 80       15.25  11   11.27    0.03   434746
## 81       14.50  14   12.68    0.03   201930
```

1.1 Plot the data and comment. Also, get a numerical summary (mean, std.dev.) of all variables. Note that the mean of Age is 7.86.

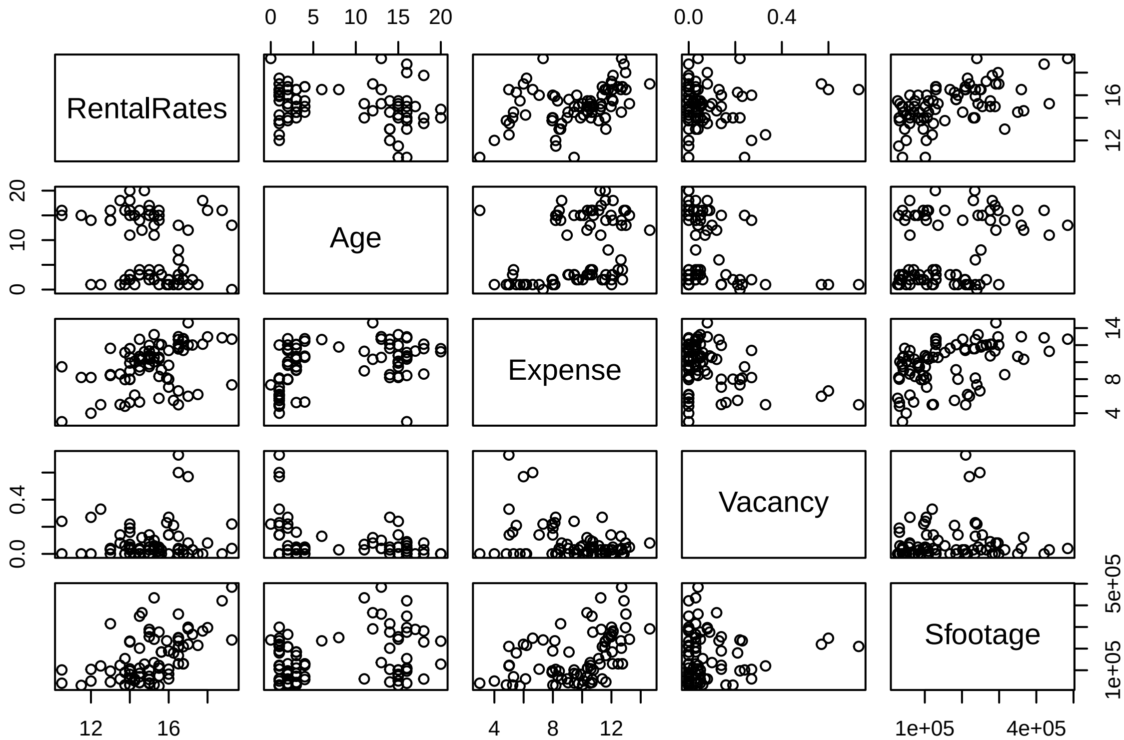
```r
par(mfrow=c(2,3))
for (i in 1:5) hist(cpdata[, i])

summary(cpdata)
```

```
##   RentalRates         Age            Expense          Vacancy
##  Min.   :10.50   Min.   : 0.000   Min.   : 3.000   Min.   :0.00000
##  1st Qu.:14.00   1st Qu.: 2.000   1st Qu.: 8.130   1st Qu.:0.00000
##  Median :15.00   Median : 4.000   Median :10.360   Median :0.03000
##  Mean   :15.14   Mean   : 7.864   Mean   : 9.688   Mean   :0.08099
##  3rd Qu.:16.50   3rd Qu.:15.000   3rd Qu.:11.620   3rd Qu.:0.09000
##  Max.   :19.25   Max.   :20.000   Max.   :14.620   Max.   :0.73000
##     Sfootage
##  Min.   : 27000
##  1st Qu.: 70000
##  Median :129614
##  Mean   :160633
##  3rd Qu.:236000
##  Max.   :484290
```



```r
pairs(cpdata)
```




```r
cor(cpdata)
```

```
##             RentalRates        Age    Expense     Vacancy   Sfootage
## RentalRates  1.00000000 -0.2502846  0.4137872  0.06652647 0.53526237
## Age         -0.25028456  1.0000000  0.3888264 -0.25266347 0.28858350
## Expense      0.41378716  0.3888264  1.0000000 -0.37976174 0.44069713
## Vacancy      0.06652647 -0.2526635 -0.3797617  1.00000000 0.08061073
## Sfootage     0.53526237  0.2885835  0.4406971  0.08061073 1.00000000
```

1.2 Follow the instruction in Problem 8.8 from the text, we will consider a regression model with predictors Age, Age^2, Expense, and Square Footage. Fit the regression model.

```r
cpdata$CentAge<-cpdata$Age-mean(cpdata$Age)

cpreg1 <- lm(RentalRates~Age+I(Age^2)+Expense+Sfootage, data=cpdata)
summary(cpreg1)
```

```
##
## Call:
## lm(formula = RentalRates ~ Age + I(Age^2) + Expense + Sfootage,
##     data = cpdata)
##
## Residuals:
```

```
##      Min       1Q   Median       3Q      Max
## -2.89596 -0.62547 -0.08907  0.62793  2.68309
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.249e+01  4.805e-01  26.000  < 2e-16 ***
## Age         -4.043e-01  1.089e-01  -3.712  0.00039 ***
## I(Age^2)     1.415e-02  5.821e-03   2.431  0.01743 *
## Expense      3.140e-01  5.880e-02   5.340 9.33e-07 ***
## Sfootage     8.046e-06  1.267e-06   6.351 1.42e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.097 on 76 degrees of freedom
## Multiple R-squared:  0.6131, Adjusted R-squared:  0.5927
## F-statistic:  30.1 on 4 and 76 DF,  p-value: 5.203e-15
```

- We can use the centered-age as suggested in the book. But it is not necessary. The regression coefficients of age^2, expense and sfootage remain unchanged.

  ```
  cpreg1b <- lm(RentalRates~CentAge+I(CentAge^2)+Expense+Sfootage, data=cpdata)
  cpreg1b$coef
  ```

  ```
  ##   (Intercept)       CentAge  I(CentAge^2)       Expense      Sfootage
  ##  1.018934e+01 -1.817749e-01  1.414773e-02  3.140313e-01  8.045878e-06
  ```

- Express the regression line in the original Age variable.

- You can also use function `poly(var, order, raw=True)` to create the polynomial terms

1.3 Add interaction terms to the previous model and examine their signifiance.

```
cpreg2<-lm(RentalRates~Age+I(Age^2)+Expense+Sfootage
    + Age:Expense + Age:Sfootage, data=cpdata)
summary(cpreg2)
```

```
##
## Call:
## lm(formula = RentalRates ~ Age + I(Age^2) + Expense + Sfootage +
##     Age:Expense + Age:Sfootage, data = cpdata)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.97236 -0.83548 -0.04637  0.68661  2.72955
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.296e+01  6.766e-01  19.151  < 2e-16 ***
## Age          -4.672e-01  1.211e-01  -3.856 0.000244 ***
## I(Age^2)      1.052e-02  6.079e-03   1.731 0.087564 .
## Expense       2.138e-01  8.127e-02   2.631 0.010357 *
## Sfootage      1.013e-05  2.370e-06   4.274 5.65e-05 ***
## Age:Expense   1.821e-02  9.962e-03   1.828 0.071539 .
## Age:Sfootage -3.125e-07  2.220e-07  -1.408 0.163392
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.085 on 74 degrees of freedom
## Multiple R-squared:  0.6319, Adjusted R-squared:  0.602
## F-statistic: 21.17 on 6 and 74 DF,  p-value: 2.672e-14
```

```r
# Using centered-variables is OK, but it is not a "must" in practice.
cpreg2b<-lm(RentalRates~CentAge+I(CentAge^2)+Expense+Sfootage
    + CentAge:Expense + CentAge:Sfootage, data=cpdata)
cpreg2b
```

```
##
## Call:
## lm(formula = RentalRates ~ CentAge + I(CentAge^2) + Expense +
##     Sfootage + CentAge:Expense + CentAge:Sfootage, data = cpdata)
##
## Coefficients:
##      (Intercept)           CentAge      I(CentAge^2)           Expense
##        9.935e+00        -3.016e-01         1.052e-02         3.570e-01
##         Sfootage   CentAge:Expense  CentAge:Sfootage
##        7.670e-06         1.821e-02        -3.125e-07
```

```r
anova(cpreg1, cpreg2)
```

```
## Analysis of Variance Table
##
## Model 1: RentalRates ~ Age + I(Age^2) + Expense + Sfootage
## Model 2: RentalRates ~ Age + I(Age^2) + Expense + Sfootage + Age:Expense +
##     Age:Sfootage
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1     76 91.535
## 2     74 87.086  2    4.4488 1.8901 0.1583
```

```r
anova(cpreg1b, cpreg2b)  # For comparison.
```

```
## Analysis of Variance Table
##
## Model 1: RentalRates ~ CentAge + I(CentAge^2) + Expense + Sfootage
## Model 2: RentalRates ~ CentAge + I(CentAge^2) + Expense + Sfootage + CentAge:Expense +
##     CentAge:Sfootage
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1     76 91.535
## 2     74 87.086  2    4.4488 1.8901 0.1583
```

2 Example 2. Qualitative/Categorical Predictors and Interactions

The data in file `twins.txt` and `twins.sav` are from a 1966 paper by Cyril Burt entitled "The genetic determination of differences in intelligence: A study of monozygotic twins reared apart". The data consist of IQ scores for identical twins, one raised by foster parents, the other by the natural parents. We also know the social class of natural parents (high, middle or low). We are interested in predicting the IQ of the twin with foster parents from the IQ of the twin with the natural parents and the social class of natural parents.

Case_i	Y:IQF(Forster)	X1:IQN(Natrual)	SocialClass
1	82	82	h
...	...	...	...
27	98	111	l

```
twindata<-read.table("../DataSets/twins.txt", header=T)
head(twindata, 2)
```

```
##   IQF IQN status
## 1  82  82      h
## 2  80  90      h
```

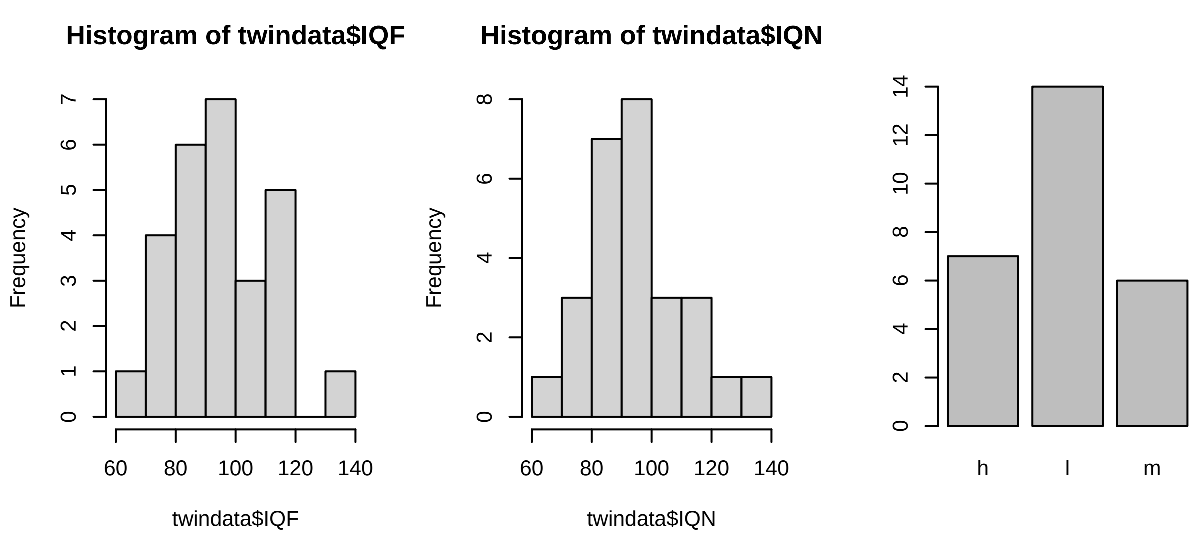
```
tail(twindata, 2)
```

```
##    IQF IQN status
## 26 107 106      l
## 27  98 111      l
```

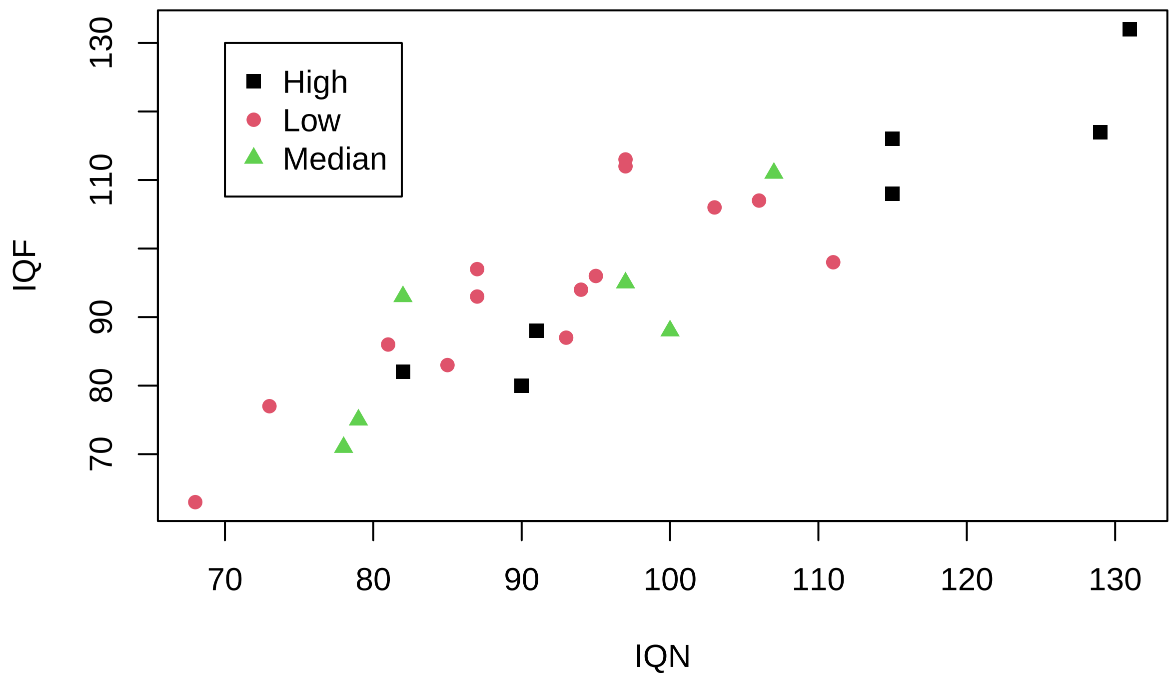
```
# Covert status to factor to simplify future code.
twindata$status <- as.factor(twindata$status)
```

2.1 Plot the data

```
par(mfrow = c(1, 3))
hist(twindata$IQF)
hist(twindata$IQN)
barplot(table(twindata$status))
```



```r
plot(IQF ~ IQN, col=status, pch=14+as.numeric(status), data=twindata)
legend(70, 130, legend=c("High", "Low", "Median"), col=c(1:3), pch=14+c(1:3))
```



- You can make nicer plots when you use **`ggplot()`** from the **ggplot2** package.

2.2 R users do not need to create dummy variables mannually.

2.3 Fit regression models.

Recall that we converted the categorical variable `status` into a *factor* variable earlier. Hence it can be used in the `lm()` function directly. If a categorical predictor is not converted into a *factor*, you must use `as.factor(status)` inside the `lm()` function.

```r
twinreg<-lm(IQF~IQN+status, data=twindata)
# twinreg<-lm(IQF~IQN+as.factor(status), data=twindata)
summary(twinreg)
```

```
##
## Call:
## lm(formula = IQF ~ IQN + status, data = twindata)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -14.8235  -5.2366  -0.1111   4.4755  13.6978
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)  -0.6076    11.8551  -0.051    0.960
## IQN           0.9658     0.1069   9.031 5.05e-09 ***
## statusl       6.2264     3.9171   1.590    0.126
## statusm       2.0353     4.5908   0.443    0.662
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.571 on 23 degrees of freedom
## Multiple R-squared:  0.8039, Adjusted R-squared:  0.7784
## F-statistic: 31.44 on 3 and 23 DF,  p-value: 2.604e-08
```

```
twinreg2<-lm(IQF~IQN+status+IQN:status, data=twindata)
summary(twinreg2)
```

```
##
## Call:
## lm(formula = IQF ~ IQN + status + IQN:status, data = twindata)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -14.479  -5.248  -0.155   4.582  13.798
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.872044  17.808264  -0.105    0.917
## IQN          0.977562   0.163192   5.990 6.04e-06 ***
## statusl      9.076654  24.448704   0.371    0.714
## statusm      2.688068  31.604178   0.085    0.933
## IQN:statusl -0.029140   0.244580  -0.119    0.906
## IQN:statusm -0.004995   0.329525  -0.015    0.988
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.921 on 21 degrees of freedom
## Multiple R-squared:  0.8041, Adjusted R-squared:  0.7574
## F-statistic: 17.24 on 5 and 21 DF,  p-value: 8.31e-07
```

```
twinreg2b<-lm(IQF~IQN*status, data=twindata)
twinreg2b
```

```
##
## Call:
## lm(formula = IQF ~ IQN * status, data = twindata)
##
## Coefficients:
## (Intercept)          IQN      statusl      statusm  IQN:statusl  IQN:statusm
##   -1.872044     0.977562     9.076654     2.688068    -0.029140    -0.004995
```

```
anova(twinreg, twinreg2)
```

```
## Analysis of Variance Table
##
## Model 1: IQF ~ IQN + status
## Model 2: IQF ~ IQN + status + IQN:status
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1     23 1318.4
```

```
## 2     21 1317.5  2   0.93181 0.0074 0.9926
```

- R treats the first level of the categorical variable as the baseline/reference level. By default, it is done alphabetically or numerically. In this case, it will be "h" for variable `status`. To change the baseline/reference level of the categorical variable, we can use the function relevel(). relevel() does not change the values of the variable, it only changes which level is considered as the reference.

```
twindata$baseH<-relevel(twindata$status, ref="h")
twindata$baseM<-relevel(twindata$status, ref="m")
twindata$baseL<-relevel(twindata$status, ref="l")

as.factor(twindata$status)
```

```
##  [1] h h h h h h h m m m m m m l l l l l l l l l l l l l l
## Levels: h l m
```

```
twindata$baseH
```

```
##  [1] h h h h h h h m m m m m m l l l l l l l l l l l l l l
## Levels: h l m
```

```
twindata$baseM
```

```
##  [1] h h h h h h h m m m m m m l l l l l l l l l l l l l l
## Levels: m h l
```

```
twindata$baseL
```

```
##  [1] h h h h h h h m m m m m m l l l l l l l l l l l l l l
## Levels: l h m
```

- The above new variables (`baseH`, `baseM`, `baseL`) are already factors. Check the following output and see whether the results are consistent.

```
twinreg.baseH <- lm(IQF~IQN+baseH, data=twindata)
twinreg.baseM <- lm(IQF~IQN+baseM, data=twindata)
twinreg.baseL <- lm(IQF~IQN+baseL, data=twindata)
```

2.4 Interpret the regression coefficients in the context of the problem.

2.5 The above model assumes that that the slope of IQN (IQ of the twin with the natural parents) remains the same for all social classes. Conduct appropriate test to determine if such assumption is reasonable.

- Think and practice:
  - How to define the hypothesis to reflect the above question?
  - How to set up the model(s) to evaluate the hypothesis?
  - How to get the numerical results from software? Can you confirm the calculation using fomula(s)?
  - How to state the conclusion in the context of the problem?

—— This is the end of Lab 4. ——