线性回归分析

Homework 8

尹秋阳, 2015011468

2017 年 6 月 3 日

1 Problem 1(KNNL 16.5)

1.1 b

```
u=c(5.1,6.3,7.9,9.5)
u2=mean(u)#7.2
sigma=2.8
(E.MSTR=sigma^2+sum(100*(u-mean(u))^2)/3)
```

```
## [1] 374.5067
```

```
(E.MSE=sigma^2)
```

```
## [1] 7.84
```

这里用到了 KNNL P694 公式 16.37。

可以看到 $E(MSTR) = 374.5$ 远远大于 $E(MSE) = 7.84$，于是可以退出至少一组 μ_i 和 μ_j 不相等。(the factor level means are not equal)

1.2 c

```
u=c(5.1,5.6,9.0,9.5)
u2=mean(u)#7.3
(E.MSTR=sigma^2+sum(100*(u-mean(u))^2)/3)
```

```
## [1] 523.1733
```

可以看到的确比 part B 的 $E(MSTR)$ 大了不少。

这是由于根据 $E(MSTR)$ 的计算式子：

$$E(MSTR)=\sigma^2+\frac{n\sum(\mu_i-\mu_.)}{r-1}(whenn_i=n)$$

可以看到，在总体 $\mu.=\bar{Y_.}$ 差不多的情况下 (前者 7.2，后者 7.3)，点离中心越远，$E(MSTR)$ 越大。在第三题的改动中，5.6 和 9.0 比 6.3 和 7.9 离中心更远，也就相应地 $E(MSR)$ 更大。

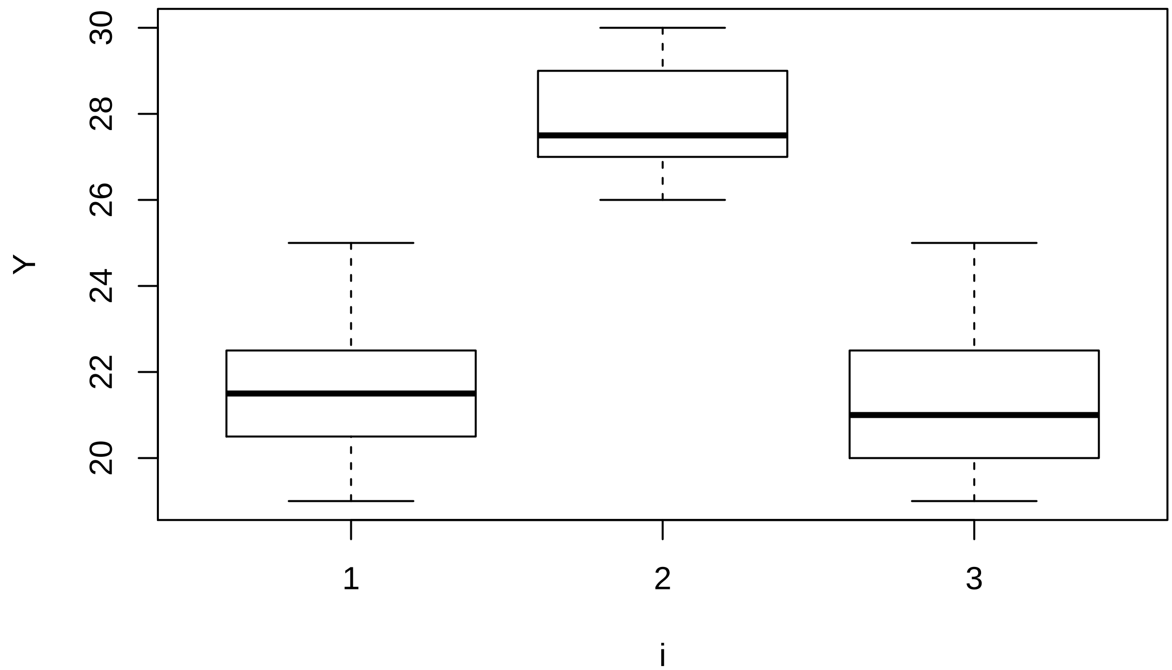
2 Problem 2

2.1 KNNL 16.10

```
dat=read.table("CH16PR10_987709608.txt")
colnames(dat)=c("Y","i","j")
dat$i=as.factor(dat$i)
```

2.1.1 a

```
plot(data=dat,Y~i)
```



看上去 Middle age 的人和其他两组在均值上还是有明显区别的。

从方差上来看，我觉得三组人也差不多。

2.1.2 b

```
fit=lm(Y~i,data = dat)
fit$fitted.values
```

```
##        1        2        3        4        5        6        7        8
## 21.50000 21.50000 21.50000 21.50000 21.50000 21.50000 21.50000 21.50000
##        9       10       11       12       13       14       15       16
## 21.50000 21.50000 21.50000 21.50000 27.75000 27.75000 27.75000 27.75000
##       17       18       19       20       21       22       23       24
## 27.75000 27.75000 27.75000 27.75000 27.75000 27.75000 27.75000 27.75000
##       25       26       27       28       29       30       31       32
## 21.41667 21.41667 21.41667 21.41667 21.41667 21.41667 21.41667 21.41667
##       33       34       35       36
## 21.41667 21.41667 21.41667 21.41667
```

2.1.3 c

```r
fit$residuals
```

```
##          1          2          3          4          5          6 
##  1.5000000  3.5000000 -0.5000000  0.5000000 -0.5000000  0.5000000 
##          7          8          9         10         11         12 
## -1.5000000  1.5000000 -2.5000000  0.5000000 -2.5000000 -0.5000000 
##         13         14         15         16         17         18 
##  0.2500000 -0.7500000 -0.7500000  1.2500000 -1.7500000  1.2500000 
##         19         20         21         22         23         24 
## -0.7500000  2.2500000  0.2500000 -0.7500000 -1.7500000  1.2500000 
##         25         26         27         28         29         30 
##  1.5833333 -1.4166667  3.5833333 -0.4166667  0.5833333  1.5833333 
##         31         32         33         34         35         36 
## -0.4166667 -1.4166667 -2.4166667 -1.4166667  0.5833333 -0.4166667
```

2.1.4 d

```r
(ano.table=anova(fit))
```

```
## Analysis of Variance Table
## 
## Response: Y
##           Df Sum Sq Mean Sq F value    Pr(>F)    
## i          2 316.72  158.36  63.601 4.769e-12 ***
## Residuals 33  82.17    2.49                      
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

2.1.5 e

$H_0: \mu_1 = \mu 2 = \cdots = \mu_r$,$H_1$: not all μ_i are equal.

Test statistic:

$$F^* = \frac{MSTR}{MSE}$$

Decision rule:

if $F^* \le F(1-\alpha; r-1; n_T - r)$ conclude H_0 if $F^* > F(1-\alpha; r-1; n_T - r)$ conclude H_1

我们可以从 d 题的 ANOVA table 中读出结论。统计量 F 的值为：

```
ano.table$`F value`[1]
```

```
## [1] 63.60142
```

相应的 p 值为：

```
ano.table$`Pr(>F)`[1]
```

```
## [1] 4.768937e-12
```

因为 p 值显著比 $\alpha(0.01)$ 小，我们拒绝原假设，认为不是所有的均值都相等。

2.1.6 f

Middle ege 的人往往在 cash offer 上比例更大，而年轻和老年人比例会少一些。这可能跟经济承受能力、急迫性有关。

TODO

2.2 KNNL 16.21

2.2.1 a

貌似 R 里面没有直接 fit 的方法。。。只能使用定义强行构造了

```
dat=dat %>% mutate(X1=if_else(i==1,1,0),X2=if_else(i==2,1,0))
dat=dat %>% mutate(X1=if_else(i==3,-1,X1),X2=if_else(i==3,-1,X2))
fit2=lm(data = dat,Y~X1+X2)
ans=summary(fit2)
ans$coefficients
```

```
##              Estimate Std. Error   t value     Pr(>|t|)
## (Intercept) 23.555556  0.2629902 89.568177 5.561918e-41
## X1          -2.055556  0.3719243 -5.526811 3.894737e-06
## X2           4.194444  0.3719243 11.277682 7.336530e-13
```

其中这里的截距项就是 $\mu_. = \bar{Y}$，也就是所有 Y 的均值

```r
mean(dat$Y)
```

```
## [1] 23.55556
```

可以看到是一样的结果。

2.2.2 b

$H_0 : \tau_1 = \tau_2 = 0$ $H_1 : \tau_1^2 + \tau_2^2 > 0$

Test statistic:

$$F = \frac{MSR}{MSE}$$

Decision rule:

if $F^* \leq F(1-\alpha; r-1; n-r)$ conclude H_0 if $F^* > F(1-\alpha; r-1; n-r)$ conclude H_1

Conclusion:

```r
ans$fstatistic
```

```
##    value    numdf    dendf
## 63.60142  2.00000 33.00000
```

对比之前使用的方法：

```r
ano.table$`Pr(>F)`[1]
```

```
## [1] 4.768937e-12
```

可以看到 F-value 是相同的。相应的 p-value：

```
## [1] 4.768937e-12
```

也是相同的。也就是说我们这里也拒绝原假设，认为不是所有的均值都相等。

其实前后两种检验在这里是等价的。

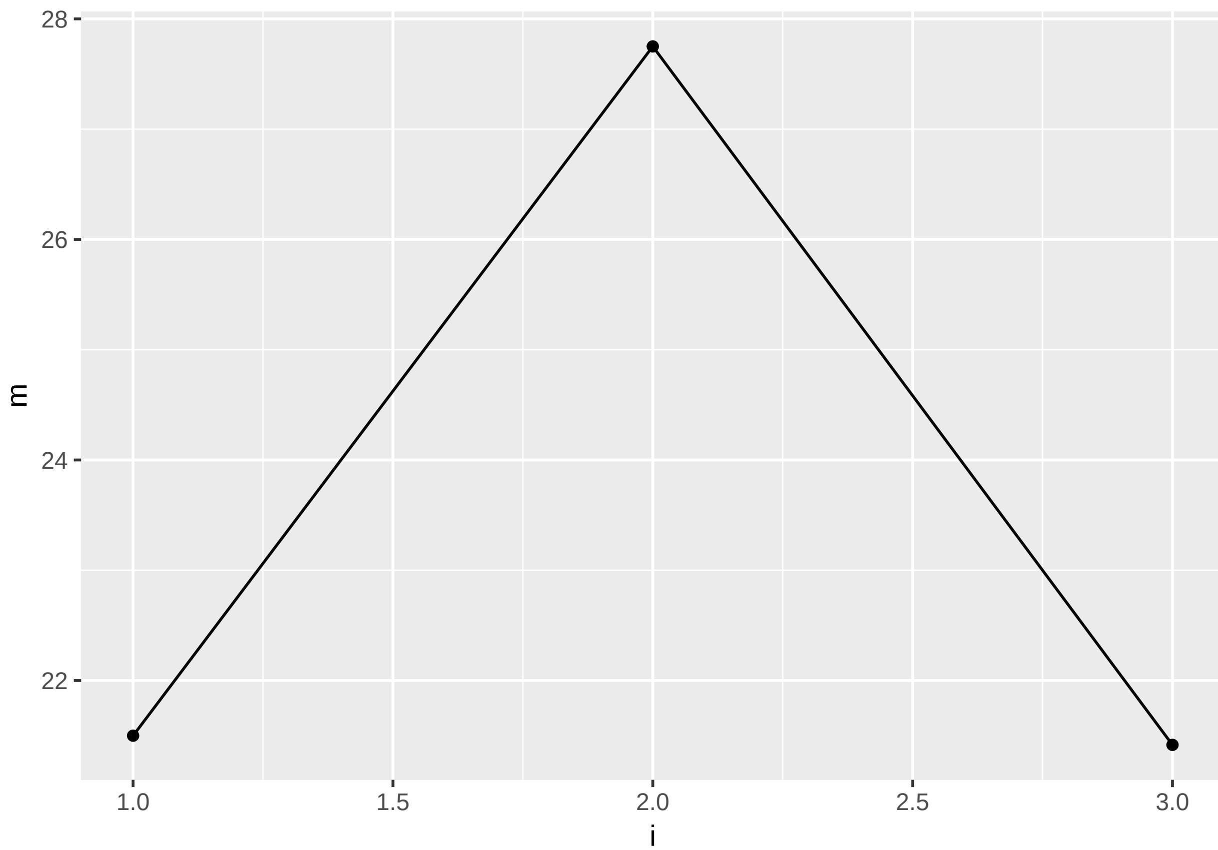
3 Problem 3

3.1 KNNL 17.11

3.1.1 a

```
meandat=dat %>% group_by(i) %>% summarise(m=mean(Y))

meandat$m=as.numeric(meandat$m)
meandat$i=as.numeric(meandat$i)
ggplot(meandat,aes(x=i,y=m))+geom_line()+geom_point()
```



这个图显示年龄的确是一个因素。

3.1.2 b

用两种方法，一是不使用 pooling variances:

```r
summaryStat <- ddply(dat, ~i, summarise, N = length(Y), mean=mean(Y), StdDev=sd(Y),  Minimun = min
summaryStat = within(summaryStat, {StdError = StdDev/sqrt(N) } )
summaryStat = within(summaryStat, {Lower99CL = mean - StdError * qt(0.995,df=N-1)})
summaryStat = within(summaryStat, {Upper99CL = mean + StdError * qt(0.995,df=N-1)})
summaryStat$Upper99CL[1]
```

```
## [1] 23.0529
```

```r
summaryStat$Lower99CL[1]
```

```
## [1] 19.9471
```

即 μ_1 99% 的置信区间是 [19.9471,23.0529]

一种是使用 pooling variance:

```r
u1=summaryStat$mean[1]
s=ans$sigma
df=36-3
tc=qt(0.995,df = df)
(upper=u1+s*tc/sqrt(12))
```

```
## [1] 22.74504
```

```r
(lower=u1-s*tc/sqrt(12))
```

```
## [1] 20.25496
```

也可以用 Anova:

```r
confint(fit,level = 0.99)[1,]
```

```
##    0.5 %   99.5 %
## 20.25496 22.74504
```

可以看到答案是一样的。

可以看到使用 pooled variance 估计的区间比之前那个要窄一些，因为自由度更大了。

3.1.3 c

```
u=meandat$m[3]-meandat$m[1]
(upper=u+tc*s*sqrt(1/12+1/12))
```

```
## [1] 1.677421
```

```
(lower=u-tc*s*sqrt(1/12+1/12))
```

```
## [1] -1.844088
```

我们有 99% 的信心认为 $D = \mu_3 - \mu_1$ 的取值在-1.844 到 1.677421 之间

3.1.4 d

令 $L = \mu_1 - 2\mu_2 + mu_3$

$$H_0 : L = 0$$

$$H_1 : L \neq 0$$

Test statistic:

$$t = \frac{L}{s(L)}$$

其中 $s(L) = s * \sqrt{(}\sum_{i=1}^{r} \frac{c_i^2}{n_i})$

Decision rule:

if $|t| \leq t(1 - \alpha/2; n_T - r)$ H_0 is concluded; otherwise, H_1 is concluded.

Conclusion

```
L=meandat$m[1]-2*meandat$m[2]+meandat$m[3]
s.L=s*sqrt(1/12+4/12+1/12)
t=L/s.L
abs(t)
```

```
## [1] 11.27768
```

```
tc
```

```
## [1] 2.733277
```

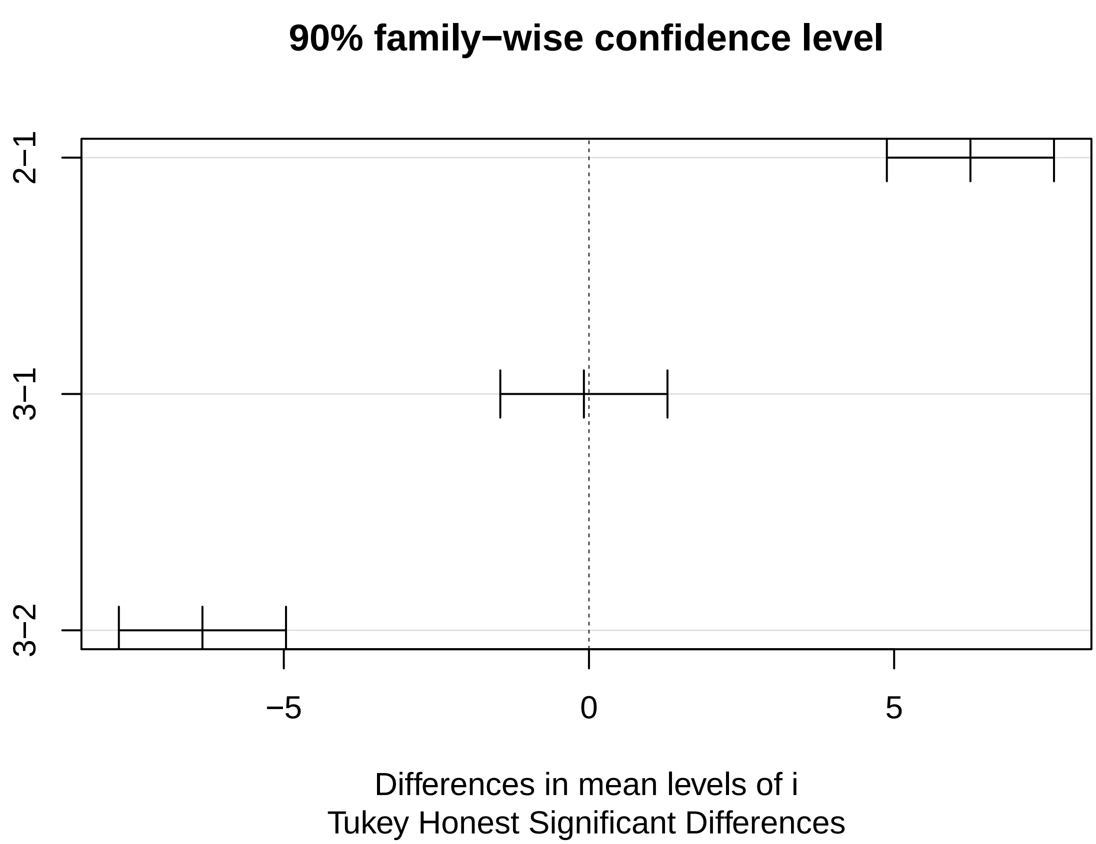
可以看到，$|t| = 11.277 > tc = 2.733$, 我们拒绝原假设，认为 $L \neq 0$，即 $\mu_2 - \mu_1 \neq mu_3 - \mu_2$

3.1.5 e

```r
mod1=aov(Y~i,data = dat)
mod1.Tukey = TukeyHSD(mod1,conf.level=0.9)
mod1.Tukey
```

```
##   Tukey multiple comparisons of means
##     90% family-wise confidence level
##
## Fit: aov(formula = Y ~ i, data = dat)
##
## $i
##            diff       lwr       upr     p adj
## 2-1  6.25000000  4.880508  7.619492 0.0000000
## 3-1 -0.08333333 -1.452825  1.286158 0.9908192
## 3-2 -6.33333333 -7.702825 -4.963842 0.0000000
```

```r
plot(mod1.Tukey, sub="Tukey Honest Significant Differences")
```



Tukey procedure 给出的置信区间由 `mod1.Tukey$i` 给出，这里 family confidence coefficient 是 0.9。表明我们有 90% 的信心认为 $\mu_2-\mu_1$,$\mu_3-\mu_1$,$\mu_3-\mu_2$ 分别落入 [4.88508,7.619492],[-1.4528,1.286158] 和 [-7.70,-4.96] 中。

三个检验的 pvalue 也已经给出，表明 1 和 3 没有明显区别，而 2 和 1，2 和 3 都有明显的区别。

这与 (a) 的图的结论一样。

3.1.6 f

```
(tc1=qtukey(0.9,3,33)/sqrt(2)) # tukey
```

```
## [1] 2.125907
```

```
(tc2=qt(1-0.1/2/3,df = 33)) # bofferoni
```

```
## [1] 2.220913
```

可以看到这里 tc1<tc2，说明 tukey 的效率更高，bofferoni 的效率没有提升。

事实上，根据 KNNLP757 的论述，当执行全部的 pairwise comparison 时，tukey 的效率往往比 bofferoni 更高效，

3.2 KNNL 17.16

3.2.1 a

```
L=meandat$m[1]-2*meandat$m[2]+meandat$m[3]
s.L=s*sqrt(1/12+4/12+1/12)
tc=qt(0.995,df = 33)
(upper=L+tc*s.L)
```

```
## [1] -9.533617
```

```
(lower=L-tc*s.L)
```

```
## [1] -15.63305
```

我们有 99% 的信心认为 $L=\mu_3-\mu_2-(\mu_2-\mu_1)$ 落在 [-15.633,-9.534] 之间。

3.2.2 b

```
(tc1=qtukey(0.9,3,33)/sqrt(2)) # tukey
```

```
## [1] 2.125907
```

```
(tc2=qt(1-0.1/2/4,df = 33)) # bofferoni
```

```
## [1] 2.348338
```

```
(tc3=sqrt((3-1)*qf(0.9,2,33)))
```

```
## [1] 2.223057
```

好像 tukey 方法的结果最小，因而最为有效；我们取 tukey 方法的值作为后续 tc。

```
c1 <- c(-1,1,0)
c2 <- c(0,-1,1)
c3 <- c(-1,0,1)
c4 <- c(1,-2,1)

cntrMat <- rbind("1 v 2"=c1,
                 "2 v 3"=c2,
                 "1 v 3"=c3,
                 "1&3 v2"=c4)
result=glht(mod1, linfct=mcp(i=cntrMat), alternative="two.sided")
confint(result,level = 0.9,calpha = tc1)
```

```
##
##   Simultaneous Confidence Intervals
##
## Multiple Comparisons of Means: User-defined Contrasts
##
##
## Fit: aov(formula = Y ~ i, data = dat)
##
## Quantile = 2.1259
## 90% confidence level
##
```

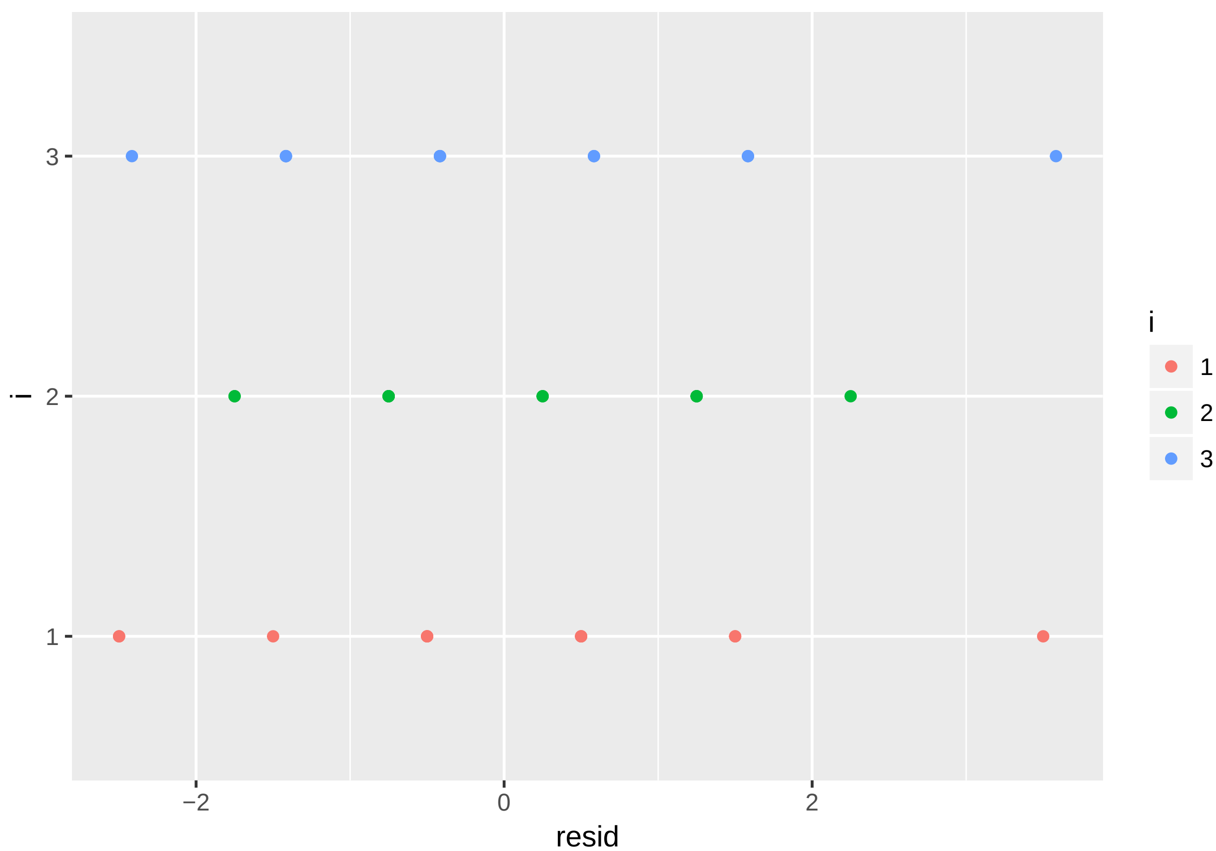
```
##
## Linear Hypotheses:
##               Estimate  lwr       upr
## 1 v 2 == 0     6.25000   4.88051   7.61949
## 2 v 3 == 0    -6.33333  -7.70283  -4.96384
## 1 v 3 == 0    -0.08333  -1.45283   1.28616
## 1&3 v2 == 0 -12.58333 -14.95536 -10.21130
```

可以看到，我们有 90% 的信心认为 $\mu_2-\mu_1$ 落于 [4.88,7.62], $\mu_3-\mu_2$ 落于 [-7.70,-4.96], $\mu_3-\mu_1$ 落于 [-1.45,1.29], 而 $\mu_1-2\mu_2+\mu_3$ 落于 [-14.96,-10.2]。

4 Problem 4

4.1 a

```r
resid=mod1$residuals
ggplot(dat,aes(x=resid,y=i,color=i))+geom_point()
```



- outliers

好像有两个点比其他点更远离中心 (>3)，需要进一步测试

- 方差

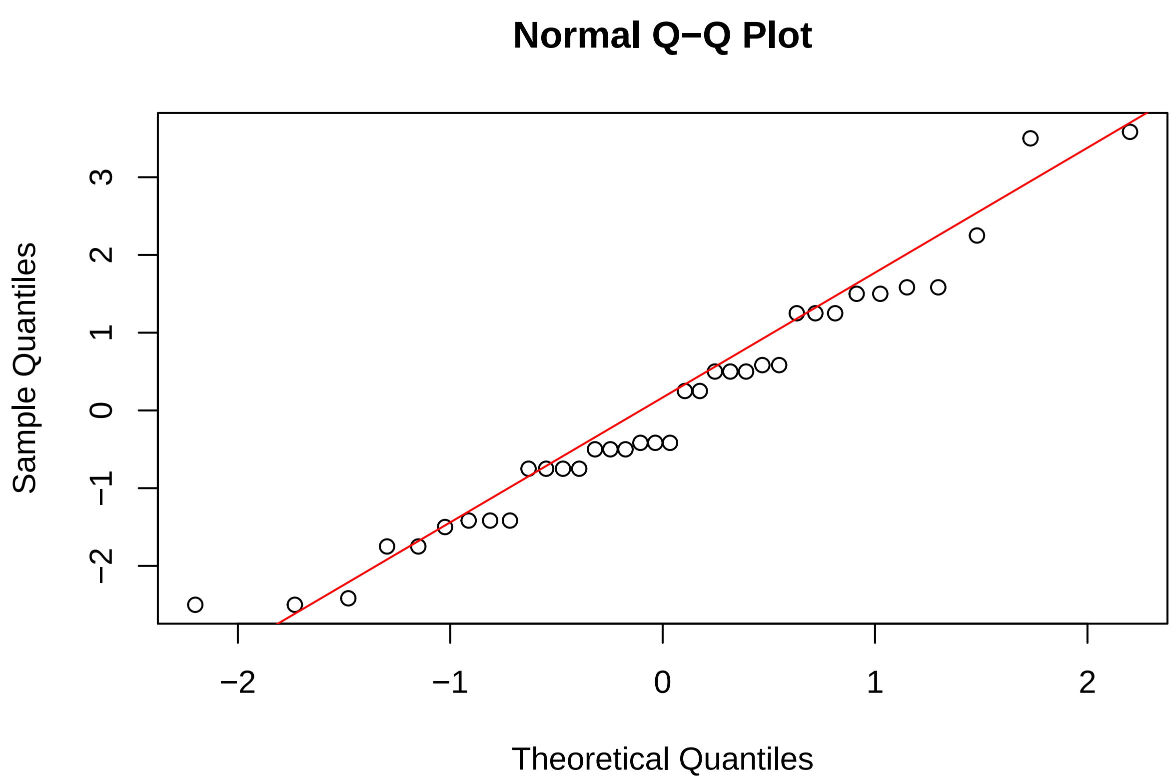
总体来说差不多，group 2(middle age) 的人可能稍微小一点。

- 正态性

好像不是很正态，基本上处于均匀分布的样子，需要进一步测试

4.2 b

```
qqnorm(resid)
qqline(resid,col=2)
```



```r
sx=sort(resid)
sy=qnorm((1:36-0.5)/36)
cor(sx,sy)
```

```
## [1] 0.9831774
```

可以看到，基本上点还是在线旁边，而且相关系数也相当的大，可以认为正态性还是近似满足的。有可能去掉两个离群值后效果会更好。(一头一尾)

4.3 d

The boferroni outlier test is shown below:

Decision rule:

For each case:

$$t_i = \frac{e_i}{MSE_{(i)}(1-h_{ii})}$$

If $|ti| < t(1-\alpha/2n; n-p-1)$ we conclude that case i is not outlying; otherwise we conclude that case i is an outltying case.(Here n=36 and p=3)

```r
stl.del.resid=rstudent(mod1)
head(stl.del.resid)
```

```
##          1          2          3          4          5          6
##  0.9926550  2.4930646 -0.3264475  0.3264475 -0.3264475  0.3264475
```

```r
outlierTest(mod1)
```

```
##
## No Studentized residuals with Bonferonni p < 0.05
## Largest |rstudent|:
##    rstudent unadjusted p-value Bonferonni p
## 27 2.564452           0.01523      0.54827
```

可以看到最小 p 值的点的 pvalue 是 0.5428，我们不拒绝原假设，认为没有离群值。