36-463 / 36-663: Multilevel & Hierarchical Models
HW03 Solution

September 26, 2016

Exercise 1a

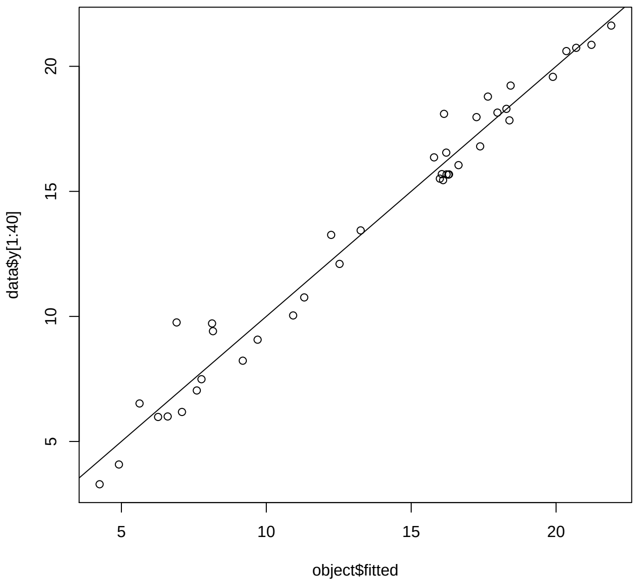
Part a

```
data <- read.table('exercise3.1.dat',T)
object <- lm(y ~ x1+ x2,data[1:40,])
summary(object)

here is the output from the summary function

Call:
lm(formula = y ~ x1 + x2, data = data[1:40, ])

Residuals:
    Min      1Q  Median      3Q     Max
-0.9585 -0.5865 -0.3356  0.3973  2.8548

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.31513    0.38769   3.392  0.00166 **
x1           0.51481    0.04590  11.216 1.84e-13 ***
x2           0.80692    0.02434  33.148  < 2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 0.9 on 37 degrees of freedom
Multiple R-squared: 0.9724,Adjusted R-squared: 0.9709
F-statistic: 652.4 on 2 and 37 DF,  p-value: < 2.2e-16
```

The R-squared measure is very high for our model(0.97). Also, all the variables in our model are significant.

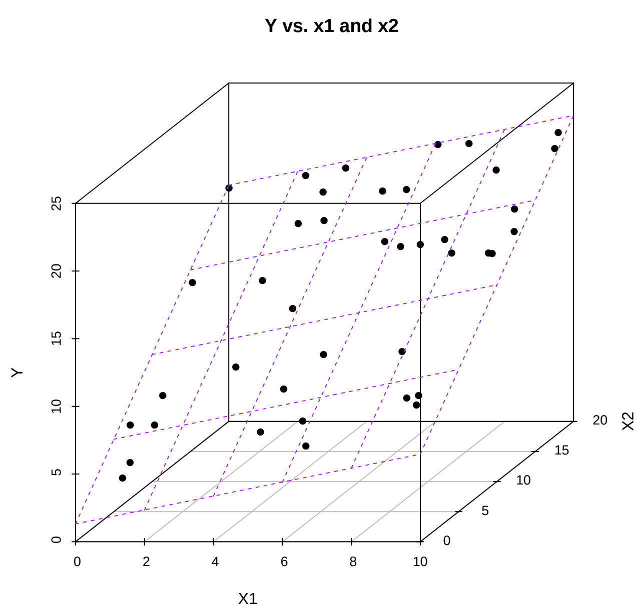
part b)

```
plot(object$fitted, data$y[1:40])
abline(0,1)
```



From the response vs the fitted value plot we can see that the model fits the data reasonably well The 3D plot is shown below:

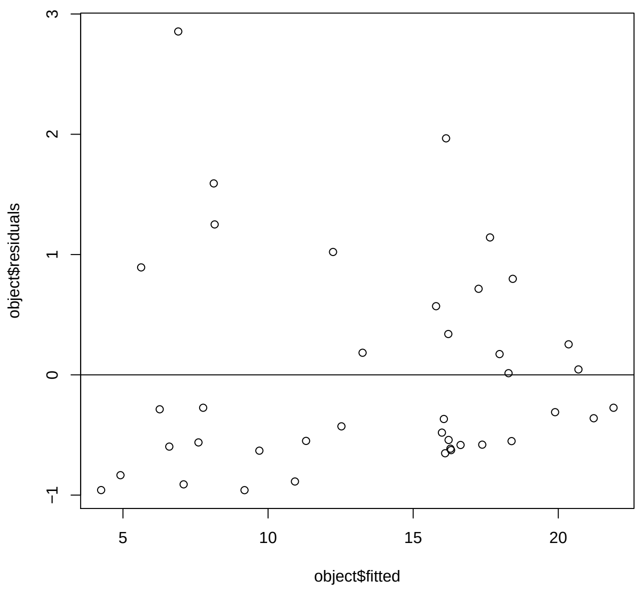
```
(Thanks to Abbas Zaidi for part of his code)
library(scatterplot3d)
plot.3d<-scatterplot3d(data$x1[1:40],data$x2[1:40],data$y[1:40],pch=16,
main="Y vs. x1 and x2",xlab="X1",ylab="X2",zlab="Y")
plot.3d$plane3d(object$coef[1],object$coef[2],object$coef[3],col="purple")
```

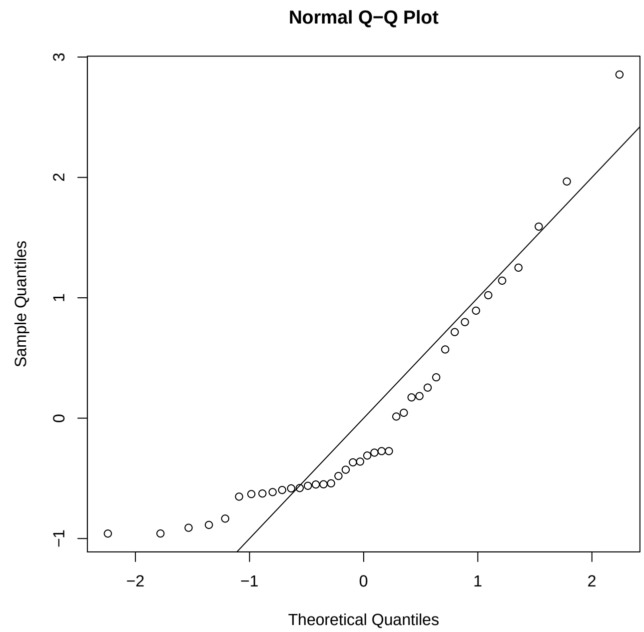


part c)

```
plot(object$fitted,object$residuals)
abline(h=0)
qqnorm(object$residuals)
abline(0,1)
```

The plot on the left checks for the equal variance assumption and linearity assumption. The plot on the right checks for the normality distribution assumption of the errors.





part d)

```
predict(object,data[41:60,],interval = "prediction",level =0.95)

         fit       lwr       upr
41 14.812484 12.916966 16.708002
42 19.142865 17.241520 21.044211
43  5.916816  3.958626  7.875005
44 10.530475  8.636141 12.424809
45 19.012485 17.118597 20.906373
46 13.398863 11.551815 15.245911
47  4.829144  2.918323  6.739965
48  9.145767  7.228364 11.063170
49  5.892489  3.979060  7.805918
50 12.338639 10.426349 14.250929
51 18.908561 17.021818 20.795303
52 16.064649 14.212209 17.917088
53  8.963122  7.084081 10.842163
54 14.972786 13.094194 16.851379
55  5.859744  3.959679  7.759808
56  7.374900  5.480921  9.268879
```

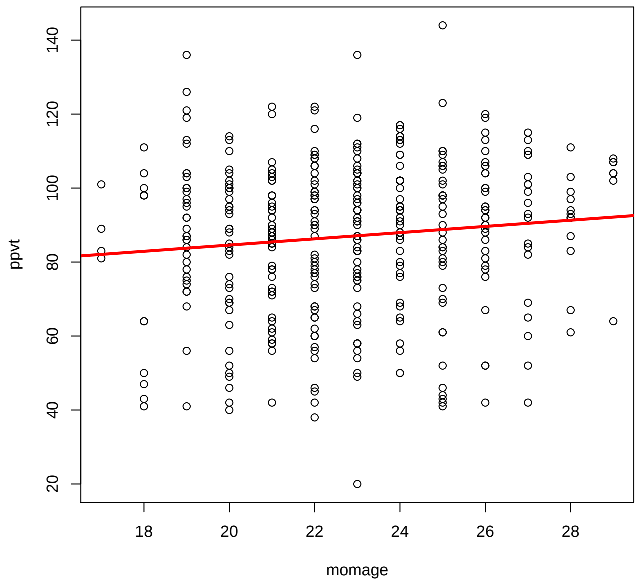
```
57  4.535267  2.616996  6.453539
58 15.133280 13.282467 16.984094
59  9.100899  7.223395 10.978403
60 16.084900 14.196990 17.972810
```

The prediction intervals have reasonable width, so we are confident about the predictions.

exercise 1b

part a

```
data <- read.dta("child.iq.dta")
attach(data)
fit <- lm(ppvt~momage)
summary(fit)
plot(ppvt ~ momage)
abline(fit,col="red",lwd=3)

Call:
lm(formula = ppvt ~ momage)

Residuals:
    Min      1Q  Median      3Q     Max
-67.109 -11.798   2.971  14.860  55.210

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  67.7827     8.6880   7.802 5.42e-14 ***
momage        0.8403     0.3786   2.219    0.027 *
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 20.34 on 398 degrees of freedom
Multiple R-squared: 0.01223,Adjusted R-squared: 0.009743
F-statistic: 4.926 on 1 and 398 DF,  p-value: 0.02702
```

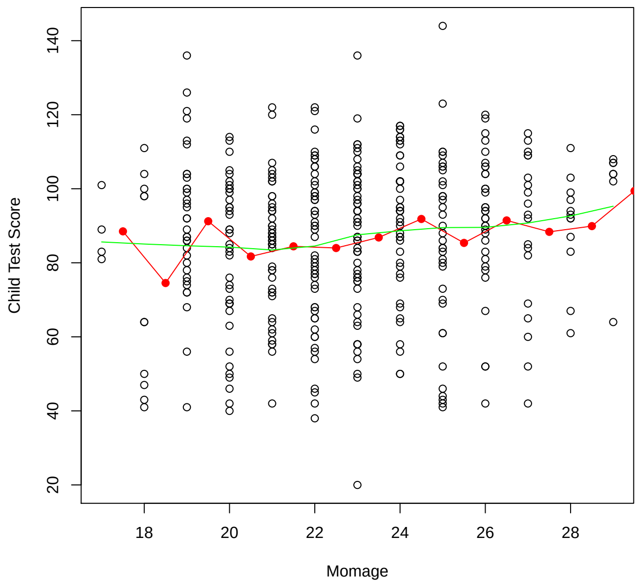


The R-squared value is pretty small which means the linear assumption is not satisfied. The slope coefficient means the child score will increase 0.84 if moms age of giving birth increases 1. Under the linear model assumption, the best age to give birth for a mom is 29. But here we assume linear relationship between mom's age and children's iq and the R-square value shows this is not an appropriate model to interpret the data.

Nonlinear check:

```
bin <- NULL
for(i in 17:29)
{
 bin <- cbind(bin,ifelse((momage>=i&momage<(i+1)),1,0))
}
fit.00 <- lm(ppvt~bin)
fit.01 <- lm(ppvt~bin-1)
plot(ppvt~momage,xlab="Momage",ylab="Child Test Score")
points((17:29)+0.5,coef(fit.01),pch=19,col="Red")
lines((17:29)+0.5,coef(fit.01),col="Red")
ord<-order(momage)
x<-momage[ord]
y<-loess(ppvt~momage)$fitted[ord]
lines(x,y,col="green")

summary(fit.00)
Call:
lm(formula = ppvt ~ bin)

Residuals:
    Min      1Q  Median      3Q     Max
-66.847 -12.455   2.545  14.153  58.622

Coefficients: (1 not defined because of singularities)
```

```
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   99.429      7.673  12.959   <2e-16 ***
bin1         -10.929     12.724  -0.859   0.3909
bin2         -24.883      9.815  -2.535   0.0116 *
bin3          -8.193      8.426  -0.972   0.3315
bin4         -17.711      8.333  -2.125   0.0342 *
bin5         -14.970      8.213  -1.823   0.0691 .
bin6         -15.429      8.173  -1.888   0.0598 .
bin7         -12.581      8.115  -1.550   0.1219
bin8          -7.584      8.248  -0.920   0.3584
bin9         -14.050      8.367  -1.679   0.0939 .
bin10         -7.974      8.447  -0.944   0.3458
bin11        -11.060      8.976  -1.232   0.2186
bin12         -9.512      9.655  -0.985   0.3251
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 20.3 on 387 degrees of freedom
Multiple R-squared: 0.0433,Adjusted R-squared: 0.01363
F-statistic:  1.46 on 12 and 387 DF,  p-value: 0.1369


summary(fit.01)
Call:
lm(formula = ppvt ~ bin - 1)

Residuals:
    Min      1Q  Median      3Q     Max
-66.847 -12.455   2.545  14.153  58.622

Coefficients:
      Estimate Std. Error t value Pr(>|t|)
bin1    88.500     10.150   8.719   <2e-16 ***
bin2    74.545      6.121  12.179   <2e-16 ***
bin3    91.235      3.481  26.206   <2e-16 ***
bin4    81.718      3.251  25.139   <2e-16 ***
bin5    84.458      2.930  28.824   <2e-16 ***
bin6    84.000      2.815  29.839   <2e-16 ***
bin7    86.847      2.643  32.861   <2e-16 ***
bin8    91.844      3.026  30.350   <2e-16 ***
bin9    85.378      3.337  25.583   <2e-16 ***
bin10   91.455      3.534  25.880   <2e-16 ***
bin11   88.368      4.657  18.975   <2e-16 ***
bin12   89.917      5.860  15.344   <2e-16 ***
bin13   99.429      7.673  12.959   <2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 20.3 on 387 degrees of freedom
Multiple R-squared:  0.95,Adjusted R-squared: 0.9483
F-statistic: 565.6 on 13 and 387 DF,  p-value: < 2.2e-16
```



It can be seen the children's iq and mom's age doesn't have non-linear relationship from the plot either.

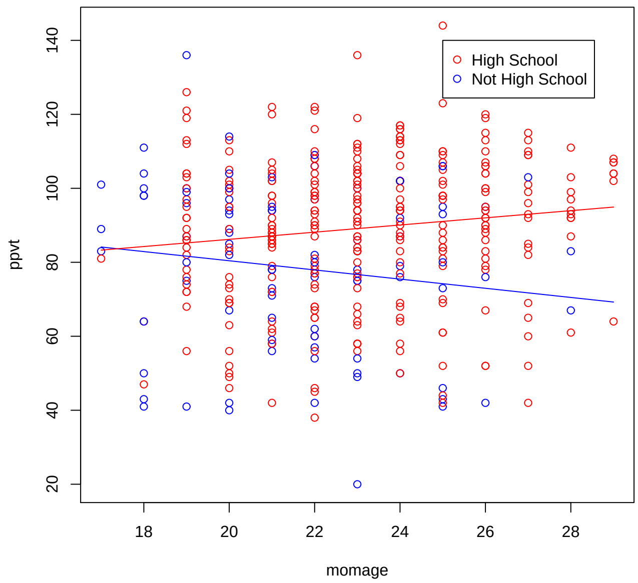
part b

```
fit.03 <- lm(ppvt ~ educ_cat +momage)
summary(fit.03)
Call:
lm(formula = ppvt ~ educ_cat + momage)

Residuals:
    Min      1Q  Median      3Q     Max
-61.763 -13.130   2.495  14.620  55.610

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  69.1554     8.5706   8.069 8.51e-15 ***
educ_cat      4.7114     1.3165   3.579 0.000388 ***
momage        0.3433     0.3981   0.862 0.389003
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 20.05 on 397 degrees of freedom
Multiple R-squared: 0.04309,Adjusted R-squared: 0.03827
F-statistic: 8.939 on 2 and 397 DF,  p-value: 0.0001594
```

The R-squared value is still pretty small, but larger than the model in part (a). And the estimation for momage becomes insignificant when including the educcat variable. The slope coefficients mean the child score will increase 0.34 if moms age of giving birth increases 1 with educcat remaining constant and the child score will increase 4.71 if the mom's education level increases 1 with mom's age remaining constant. The coefficient of momage is positive. We still recommend moms give birth late. But this is not a major effect according to the estimates and its p-value.

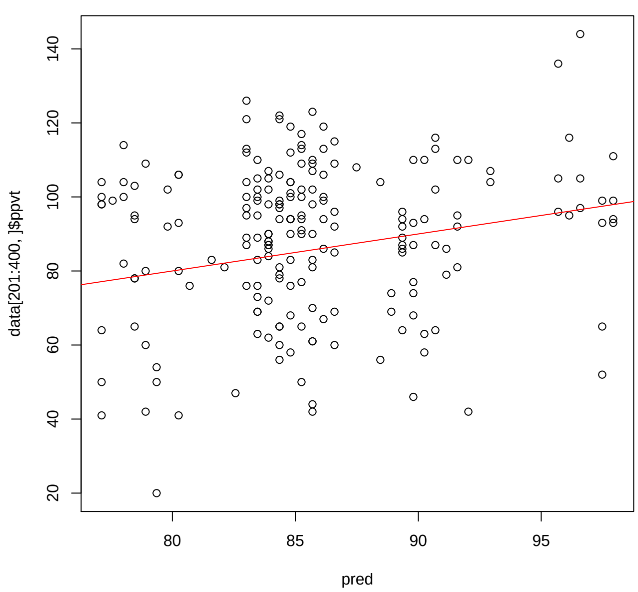
part c

```
indicator <- ifelse((educ_cat==1),0,1)
fit.04<- lm(ppvt~indicator+momage+indicator:momage)
summary(fit.04)
Call:
lm(formula = ppvt ~ indicator + momage + indicator:momage)

Residuals:
    Min      1Q  Median      3Q     Max
-56.696 -12.407   2.022  14.804  54.343

Coefficients:
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)      105.2202    17.6454   5.963 5.49e-09 ***
indicator        -38.4088    20.2815  -1.894   0.0590 .
momage            -1.2402     0.8113  -1.529   0.1271
indicator:momage   2.2097     0.9181   2.407   0.0165 *
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 19.85 on 396 degrees of freedom
Multiple R-squared: 0.06417,Adjusted R-squared: 0.05708
F-statistic: 9.051 on 3 and 396 DF,  p-value: 8.276e-06

(Thanks to Lisha Sun for part of his/her code)
plot(ppvt~momage,col=c("Blue","Red")[indicator+1])
legend(25,140,pch=1,col=c("Red","Blue"),legend=c("High School","Not High School"))
curve(cbind(1,0,x,0*x) %*% coef(fit.04),add=T,col="Blue")
curve(cbind(1,1,x,1*x) %*% coef(fit.04),add=T,col="Red")
```



part d

```
fit.05 <- lm(ppvt~momage+educ_cat, data[1:200,])
pred<-predict(fit.05, newdata=data[201:400,])
plot(data[201:400,]$ppvt~pred)
abline(0,1,col=2)
```

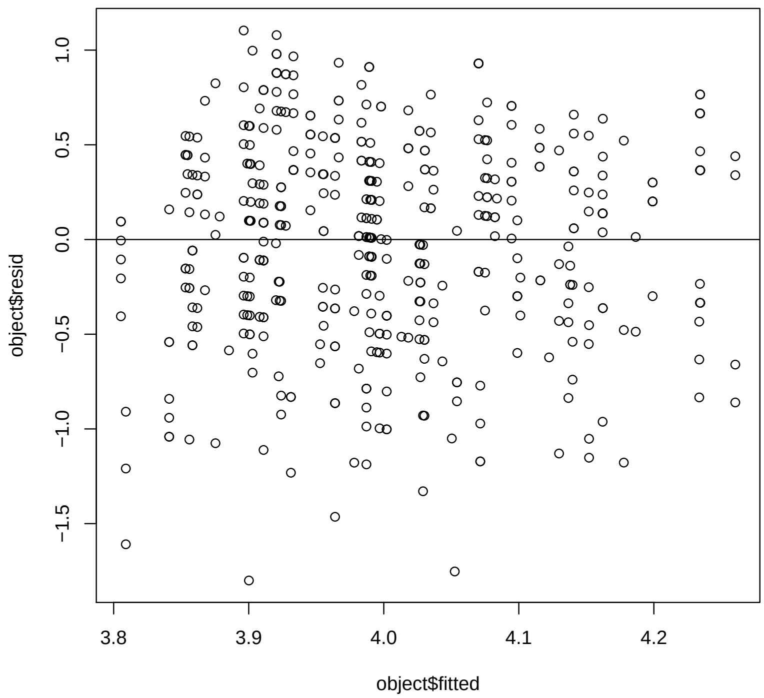


exercise 1c

part a

```
data <- read.csv('ProfEvaltnsBeautyPublic.csv',T)
object <- lm(courseevaluation ~ btystdave,data)
summary(object)


Call:
lm(formula = courseevaluation ~ btystdave, data = data)

Residuals:
     Min       1Q   Median       3Q      Max
-1.80015 -0.36304  0.07254  0.40207  1.10373

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  4.01002    0.02551 157.205  < 2e-16 ***
btystdave    0.13300    0.03218   4.133 4.25e-05 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 0.5455 on 461 degrees of freedom
Multiple R-squared: 0.03574,Adjusted R-squared: 0.03364
F-statistic: 17.08 on 1 and 461 DF,  p-value: 4.247e-05
```

the residual standard deviation is 0.5455.



part b

```
object <- lm(courseevaluation ~ profevaluation + female + female*profevaluation,data)
summary(object)

Call:
lm(formula = courseevaluation ~ profevaluation + female + female *
    profevaluation, data = data)

Residuals:
     Min       1Q   Median       3Q      Max
-0.97354 -0.12996  0.01517  0.14338  0.76507

Coefficients:
                      Estimate Std. Error t value Pr(>|t|)
(Intercept)           -0.17990    0.09709  -1.853 0.064538 .
profevaluation         1.00345    0.02276  44.093  < 2e-16 ***
female                 0.44992    0.14089   3.193 0.001503 **
profevaluation:female -0.11628    0.03360  -3.461 0.000588 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 0.1941 on 459 degrees of freedom
Multiple R-squared: 0.8785,Adjusted R-squared: 0.8777
F-statistic:  1106 on 3 and 459 DF,  p-value: < 2.2e-16
```

So this model has the professor evaluation, beauty, gender and the interaction between professor evaluation and gender as predictors. All of the predictors are significant. Female professors on average score 0.45 units better than male professors in course evaluation. We can view the female variable as the difference in intercept between the linear model for male and female. We can view the interaction term as the difference in slope between the linear model for male and female.

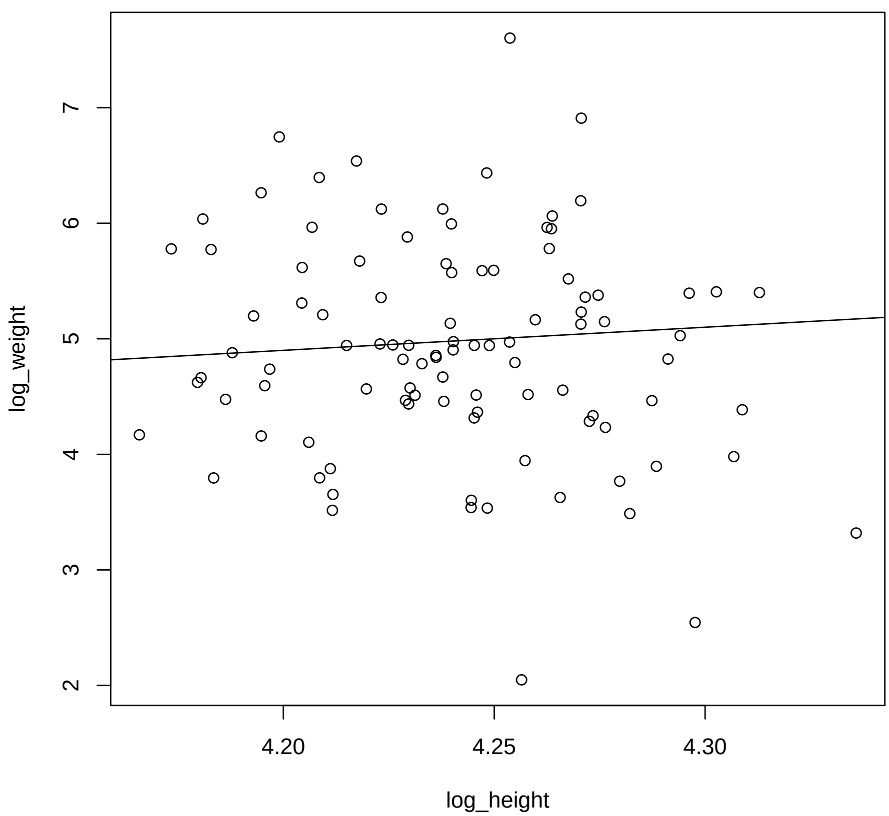
Exercise 2

part a

In the log domain, 68% of the persons will have weights within a factor of 0.25 of their predicted values from regression

so in the last homework, we saw the data that the heights of men in the US are approximately normally distributed with mean 69.1 inches and standard deviation 2.9 inches. we can use the following code:

```
height <-rnorm(100,69.1,sd =2.9)
error <- rnorm(100,0,1)
y <- -3.5 + 2*log(height)+error
plot(log(height),y,xlab = 'log_height',ylab='log_weight')
abline(-3.5,2)
```

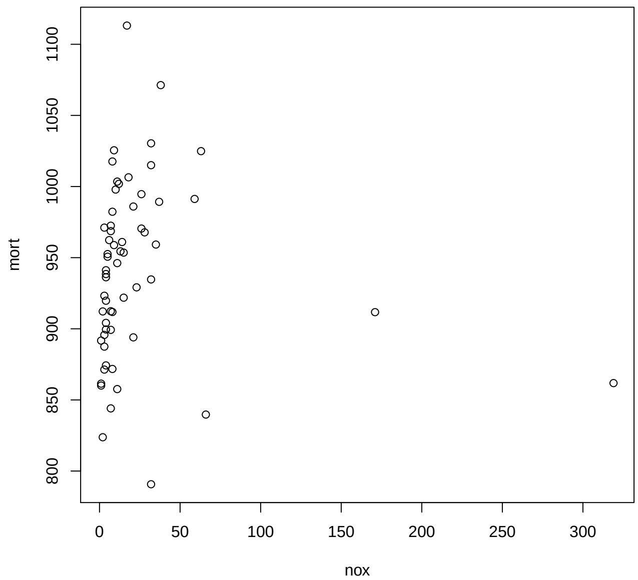


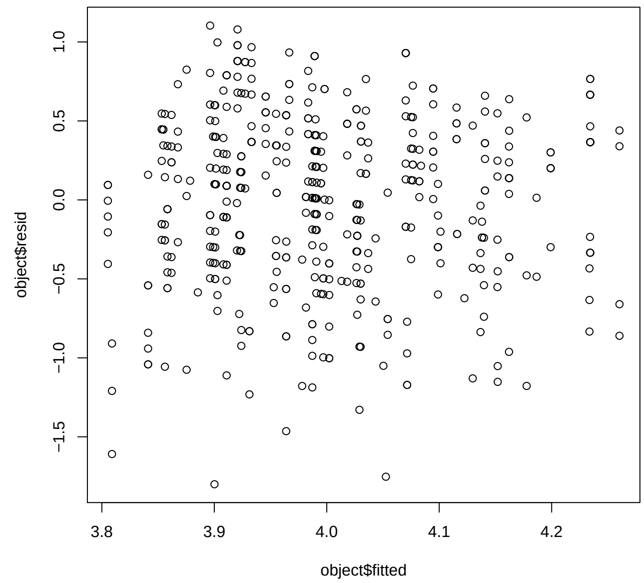
part b

part (a)

```
library(foreign)
data <- read.dta("pollution.dta")
attach(data)
plot(nox,mort)

object <- lm(mort ~ nox)
summary(object)

Call:
lm(formula = mort ~ nox)

Residuals:
     Min       1Q   Median       3Q      Max
-148.654  -43.710    1.751   41.663  172.211

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 942.7115     9.0034 104.706   <2e-16 ***
nox          -0.1039     0.1758  -0.591    0.557
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 62.55 on 58 degrees of freedom
Multiple R-squared: 0.005987,Adjusted R-squared: -0.01115
F-statistic: 0.3494 on 1 and 58 DF,  p-value: 0.5568

 plot(object$fitted,object$resid)
```

no the linear regression model won't fit these data well. The residual plot from the regression look very bad.

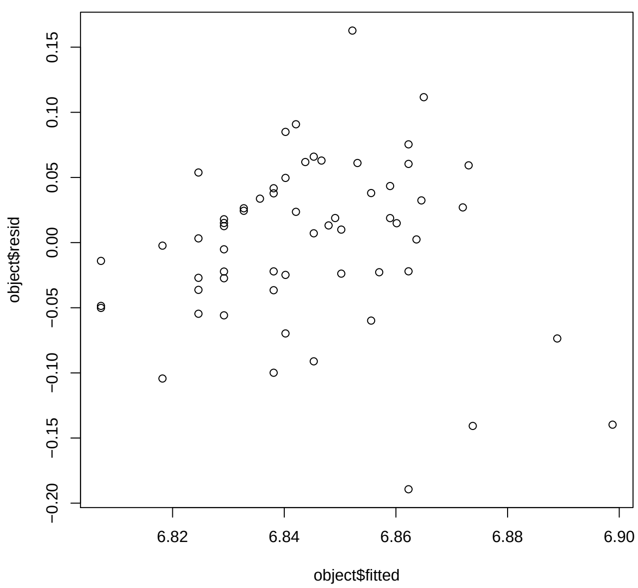




part (b) applying the log transformation

```
object <- lm(log(mort) ~ log(nox))
summary(object)
 plot(object$fitted,object$resid)

 Call:
lm(formula = log(mort) ~ log(nox))

Residuals:
     Min       1Q   Median       3Q      Max
-0.18930 -0.02957  0.01132  0.03897  0.16275

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 6.807175   0.018349 370.975   <2e-16 ***
log(nox)    0.015893   0.007048   2.255   0.0279 *
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 0.06412 on 58 degrees of freedom
Multiple R-squared: 0.08061,Adjusted R-squared: 0.06476
F-statistic: 5.085 on 1 and 58 DF,  p-value: 0.02792
```

The new residual plot looks much better after taking the log transformation

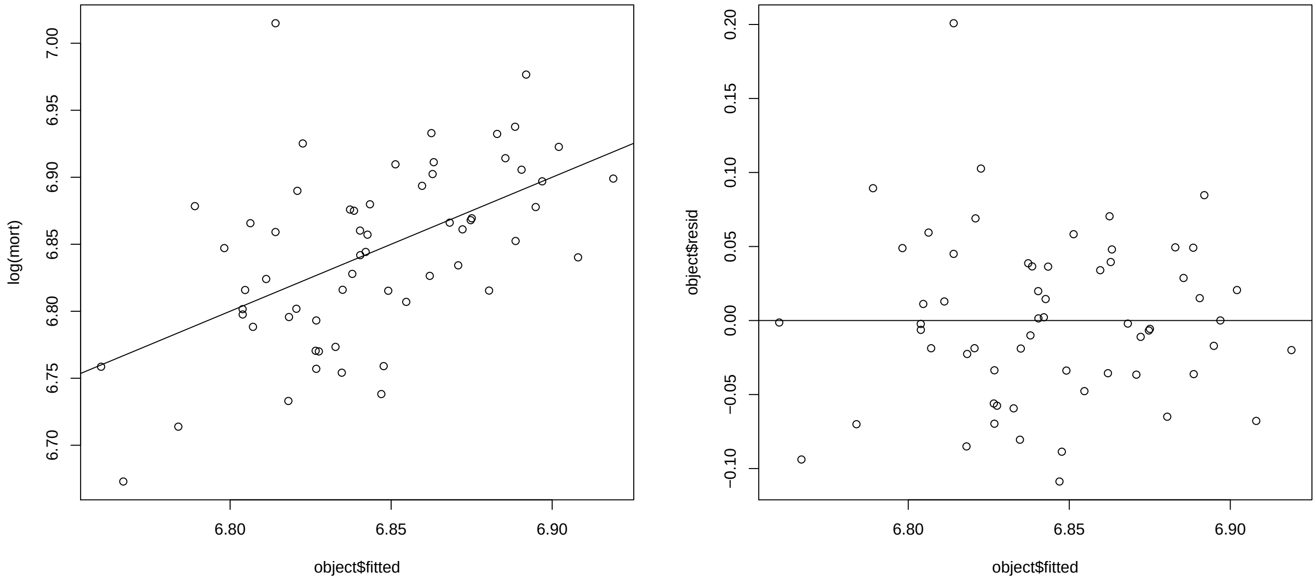


part (c) the slope is 0.015 and it is significant. we can see that for one unit increase in log nitric oxides, the log age-adjusted mortality rate goes up by 0.015.

part(d)

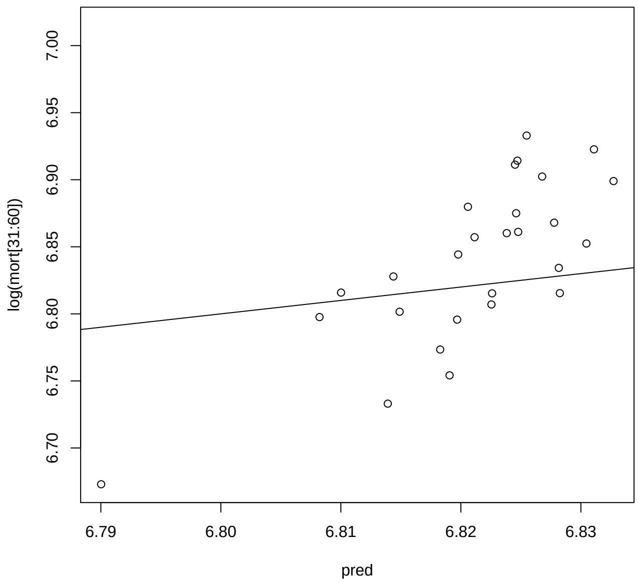
```
> object = lm(log(mort) ~ log(nox) + log(so2) + log(hc) )
> plot(object$fitted,object$resid)
> summary(object)

Call:
lm(formula = log(mort) ~ log(nox) + log(so2) + log(hc))

Residuals:
      Min        1Q    Median        3Q       Max
-0.108743 -0.035743 -0.002180  0.037092  0.200851

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  6.826749   0.022701 300.726  < 2e-16 ***
log(nox)     0.059837   0.023021   2.599  0.01192 *
log(so2)     0.014309   0.007584   1.887  0.06436 .
log(hc)     -0.060812   0.020553  -2.959  0.00452 **
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 0.05753 on 56 degrees of freedom
Multiple R-squared: 0.2852,Adjusted R-squared: 0.2469
F-statistic: 7.449 on 3 and 56 DF,  p-value: 0.0002777

 plot(object$fitted,log(mort))
 abline(0,1)
plot(object$fitted,object$resid)
```

```
abline(h=0)
```



part (e)

```
object <- lm(log(mort[1:30]) ~ log(nox[1:30]) + log(so2[1:30]) + log(hc[1:30]) )
test <- data[31:60,c(12,13,14,16)]
test <- log(test)
pred <- predict(object, test)
plot(pred, log(mort[31:60]))
abline(0,1)
```



part c

part (a) obviously, the ratio and logarithmic difference $log\frac{D_i}{R_i}$ has the disadvantage that $\frac{D_i}{R_i}$ may not be defined if either D_i or R_i is zero or very small. Also, each district has different population size. so the simple difference may not be the best across different district i. The ratio and relative proportion have less of this problem though.

part (b) There are many ways to do this problem. Here is one way: similarly to the example on page 65, we can use logistic regression to model the probabilities that both D_i and R_i are positive, then apply say the relative proportion transformation in the second step and incorporate it into a regression model. This of course makes the inference more difficult than a regular regression model.