Stat 207 HW5

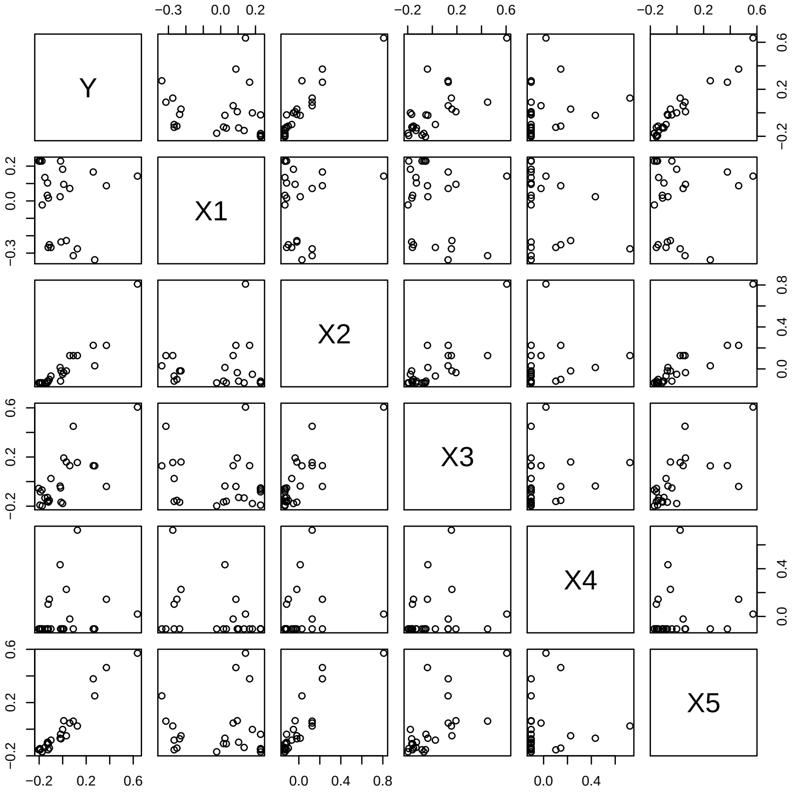
Cheng Luo 912466499
Fan Wu 912538518

February 18, 2015

1 1

(a)

```
require(gdata)

## Loading required package:  gdata
## gdata:  read.xls support for 'XLS' (Excel 97-2004) files ENABLED.
##
## gdata:  read.xls support for 'XLSX' (Excel 2007+) files ENABLED.
##
## Attaching package:  'gdata'
##
## The following object is masked from 'package:stats':
##
##    nobs
##
## The following object is masked from 'package:utils':
##
##    object.size

dat1 = read.xls("apartment.xlsx", header = TRUE)
dat.stan = dat1
for(j in 1:ncol(dat1))
  dat.stan[,j] = (dat1[,j] - mean(dat1[,j]))/(sd(dat1[,j])*(sqrt(25-1)))
dat = dat.stan
plot(dat)
```



```
cor(dat)
```

```
##              Y          X1          X2         X3          X4          X5
## Y   1.0000000 -0.11453460  0.92345442  0.7413715  0.22497528 0.96813120
## X1 -0.1145346  1.00000000 -0.01415504 -0.1885895 -0.36265249 0.02700832
## X2  0.9234544 -0.01415504  1.00000000  0.8000696  0.22412565 0.87786360
## X3  0.7413715 -0.18858951  0.80006959  1.0000000  0.16609137 0.67269398
## X4  0.2249753 -0.36265249  0.22412565  0.1660914  1.00000000 0.08929658
## X5  0.9681312  0.02700832  0.87786360  0.6726940  0.08929658 1.00000000
```

We find that Y is highly correlated with X2, X3 and X5. and X2, X3 and X5 are highly correlated with each other, which means the multicollinearity is present.

(b)
```
require(Matrix)
## Loading required package:  Matrix
## Warning:  package 'Matrix' was built under R version 3.1.2

  X = dat[, 2:6]
  X = as.matrix(X)
  P = t(X) %*% X
  eigen(P)

## $values
## [1] 2.63414515 1.33018568 0.65704211 0.29575194 0.08287513
##
## $vectors
##             [,1]         [,2]        [,3]        [,4]        [,5]
## [1,] -0.1012189  0.720376426  0.63332131 -0.25097092  0.08203814
## [2,]  0.5913231  0.122265531  0.09155295  0.08619205 -0.78713218
## [3,]  0.5475583  0.004633543 -0.24983677 -0.73925733  0.30205730
## [4,]  0.1893656 -0.646792392  0.72651658 -0.06042441  0.11967806
## [5,]  0.5517357  0.218511047  0.01665599  0.61598053  0.51781389
```

Some eigenvalues are not too close to zero, so that it does not exist serious multicollinearity.

(c)

```
  fit = lm(Y ~ 0 + ., data = dat)
  summary(fit)

##
## Call:
## lm(formula = Y ~ 0 + ., data = dat)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.052406 -0.016609 -0.004069  0.014375  0.070701
##
## Coefficients:
##    Estimate Std. Error t value Pr(>|t|)
## X1 -0.10461    0.03556  -2.942  0.00807 **
## X2  0.24656    0.08636   2.855  0.00979 **
## X3  0.01854    0.05545   0.334  0.74159
## X4  0.06294    0.03581   1.758  0.09410 .
## X5  0.73642    0.06744  10.920 7.06e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.03121 on 20 degrees of freedom
## Multiple R-squared:  0.9805,Adjusted R-squared:  0.9757
## F-statistic: 201.4 on 5 and 20 DF,  p-value: < 2.2e-16
```

```
  anova(fit)

## Analysis of Variance Table
##
## Response: Y
##           Df  Sum Sq Mean Sq  F value    Pr(>F)
## X1         1 0.01312 0.01312  13.4704  0.001519 **
## X2         1 0.84995 0.84995 872.7651 < 2.2e-16 ***
## X3         1 0.00073 0.00073   0.7458  0.398038
## X4         1 0.00061 0.00061   0.6247  0.438588
## X5         1 0.11612 0.11612 119.2409 7.064e-10 ***
## Residuals 20 0.01948 0.00097
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

In this multiple regression model, X1,X2, and X5 are more important to predict sale price.

(d)

```
  require(faraway)
## Loading required package:  faraway

  vif(fit)

## Warning in vif.lm(fit):  No intercept term detected.  Results
may surprise.

##       X1       X2       X3       X4       X5
## 1.298654 7.657888 3.157590 1.316618 4.670186
```

All VIF >1 shows that each X variable has the intercorrelation with the rest of the X variables.

(e)

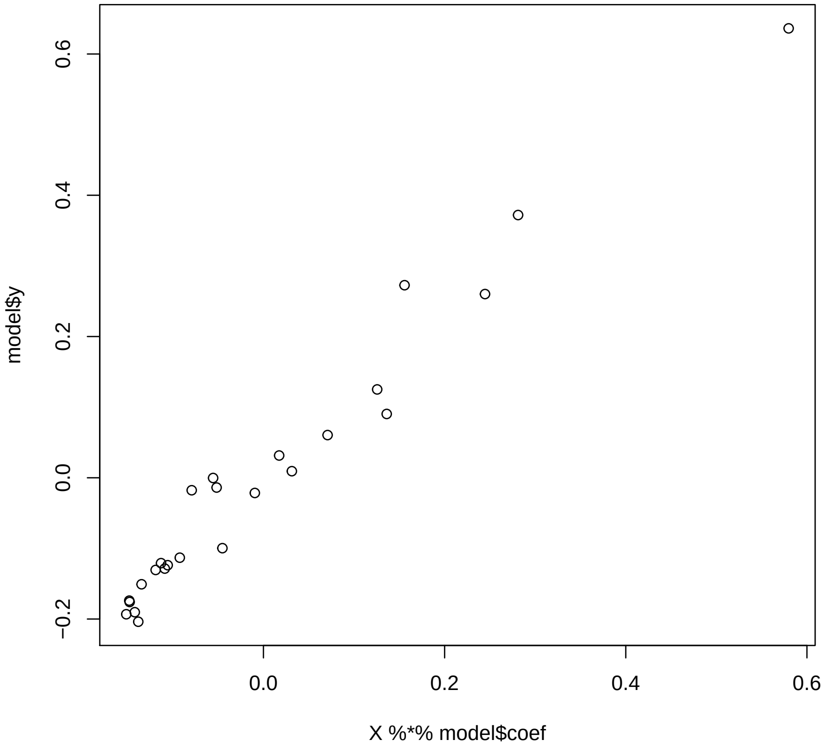
```
  library('MASS')
## Warning:  package 'MASS' was built under R version 3.1.2

  select(lm.ridge(Y ~ 0 + ., data = dat,
                  lambda = seq(0, 1, .001)))

## modified HKB estimator is 0.1181183
## modified L-W estimator is 0.07448998
## smallest value of GCV  at 0.321

  k = .321
  require('ridge')
```

```
## Loading required package:  ridge
## Warning:  package ’ridge’ was built under R version 3.1.2

  model = linearRidge(Y ~ 0 + ., data = dat,
                    lambda = k, scaling = 'none'); model

##
## Call:
## linearRidge(formula = Y ~ 0 + ., data = dat, lambda = k, scaling = "none")
##
##          X1          X2          X3          X4          X5
## -0.06004494  0.30522205  0.12451487  0.05500698  0.46414629

  summary(model)

##
## Call:
## linearRidge(formula = Y ~ 0 + ., data = dat, lambda = k, scaling = "none")
##
##
## Coefficients:
##    Estimate Std. Error t value Pr(>|t|)
## X1 -0.06004    0.03763   1.596  0.11060
## X2  0.30522    0.03263   9.355  < 2e-16 ***
## X3  0.12451    0.03829   3.252  0.00114 **
## X4  0.05501    0.03774   1.457  0.14501
## X5  0.46415    0.03640  12.751  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Ridge parameter: 0.321
##
## Degrees of freedom: model 3.053 , variance 2.167 , residual 3.94
```

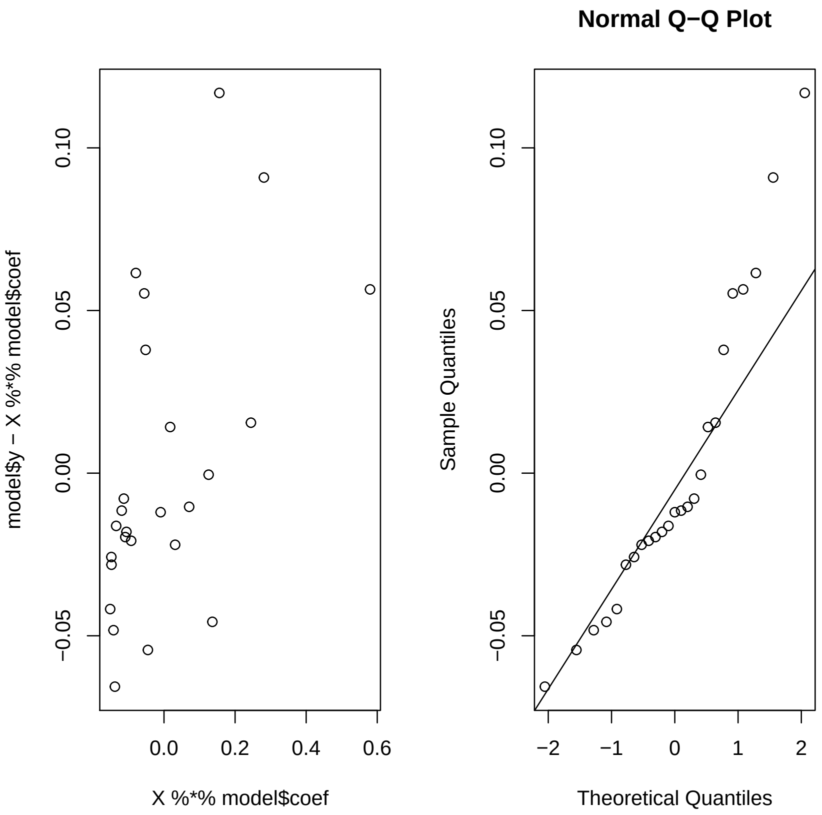
Parameter estimates and their standard errors are shown in the summary(model).

(f)

```
  plot(X%*%model$coef, model$y)
```



```r
par( mfrow = c(1, 2))
plot(X%*%model$coef, model$y - X%*%model$coef)
qqnorm(model$y - X%*%model$coef)
qqline(model$y - X%*%model$coef)
```



The residuals versus fitted values plots shows no sign for unequal variance.And the QQ-plot indicates approximately normal distribution with heavy tail, so that normality assumption seems to be reasonable, we can use model here.

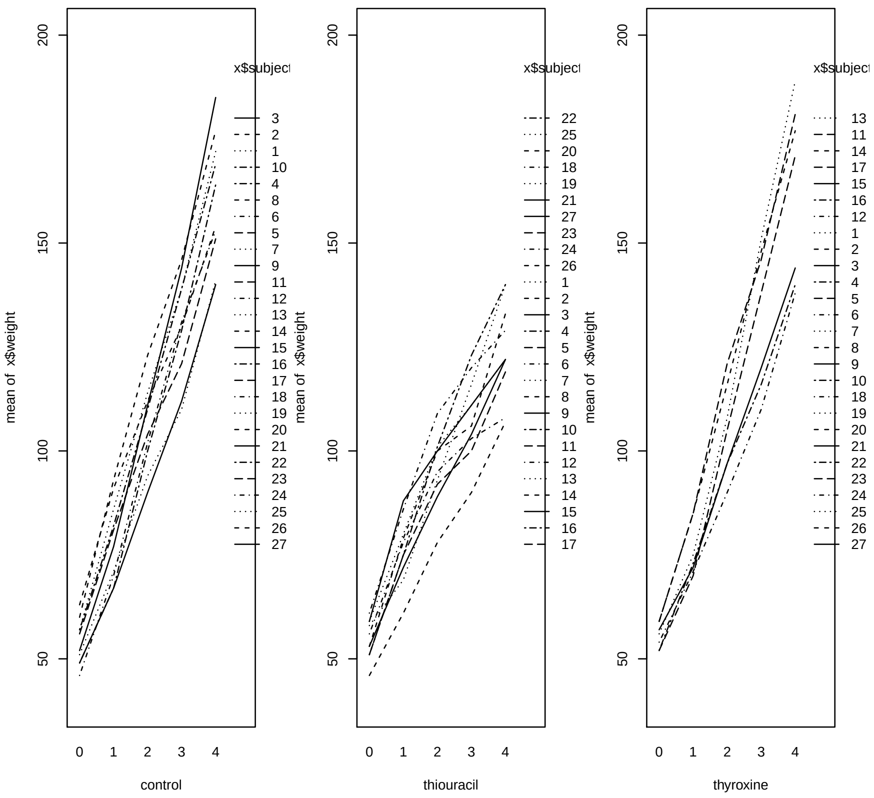
(g)
```
ans = solve(P + diag(k,5,5)) %*% P %*% solve(P + diag(k,5,5))
diag(ans)

##        X1        X2        X3        X4        X5
## 0.5841713 0.4390908 0.6045876 0.5875868 0.5465513
```

VIF of the estimated ridge regression are shown in the above. All VIF are smaller than 1, which means they have little intercorrelation between X variables.

2 2

(a)

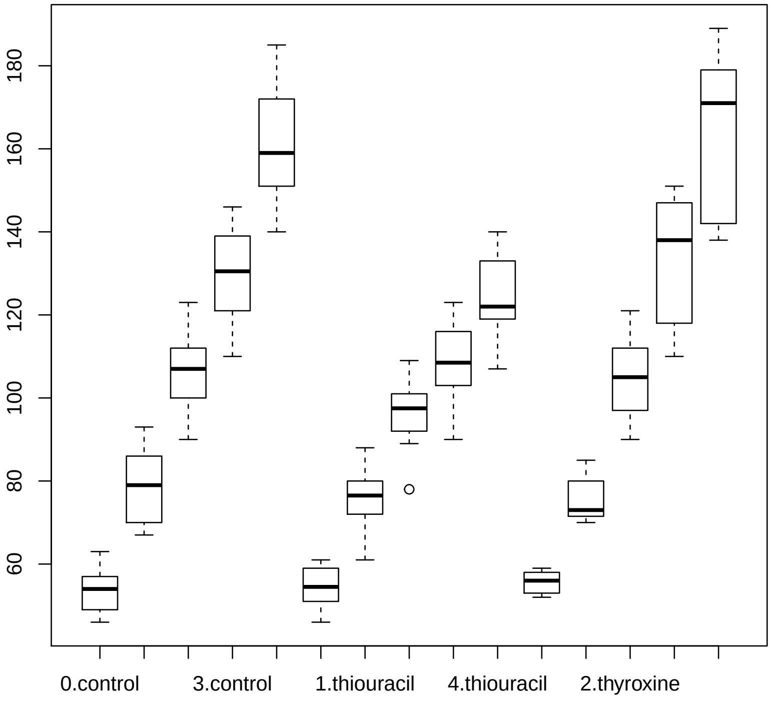
```
  require(gdata)
  dat = read.xls("ratdrink.xlsx")
  dat = dat[, 1:4]
  dat$wk = dat$weeks - mean(dat$weeks); dat$wk

##   [1] -2 -1  0  1  2 -2 -1  0  1  2 -2 -1  0  1  2 -2 -1  0  1  2 -2 -1  0
##  [24]  1  2 -2 -1  0  1  2 -2 -1  0  1  2 -2 -1  0  1  2 -2 -1  0  1  2 -2
##  [47] -1  0  1  2 -2 -1  0  1  2 -2 -1  0  1  2 -2 -1  0  1  2 -2 -1  0  1
##  [70]  2 -2 -1  0  1  2 -2 -1  0  1  2 -2 -1  0  1  2 -2 -1  0  1  2 -2 -1
##  [93]  0  1  2 -2 -1  0  1  2 -2 -1  0  1  2 -2 -1  0  1  2 -2 -1  0  1  2
## [116] -2 -1  0  1  2 -2 -1  0  1  2 -2 -1  0  1  2 -2 -1  0  1  2

  dat$wk2 = dat$wk^2; dat$wk2

##   [1] 4 1 0 1 4 4 1 0 1 4 4 1 0 1 4 4 1 0 1 4 4 1 0 1 4 4 1 0 1 4 4 1 0 1 4
##  [36] 4 1 0 1 4 4 1 0 1 4 4 1 0 1 4 4 1 0 1 4 4 1 0 1 4 4 1 0 1 4 4 1 0 1 4
##  [71] 4 1 0 1 4 4 1 0 1 4 4 1 0 1 4 4 1 0 1 4 4 1 0 1 4 4 1 0 1 4 4 1 0 1 4
## [106] 4 1 0 1 4 4 1 0 1 4 4 1 0 1 4 4 1 0 1 4 4 1 0 1 4 4 1 0 1 4

  dat$weeks = as.factor(dat$weeks)
  dat$subject = as.factor(dat$subject)
  datt = split(dat, dat$treat)
  par(mfrow = c(1, 3))
  sapply(datt, function(x){interaction.plot(x$weeks, x$subject, x$weight, ylim = c(40, 2
```



```
## $control
## NULL
##
## $thiouracil
## NULL
##
## $thyroxine
## NULL
```

(b)

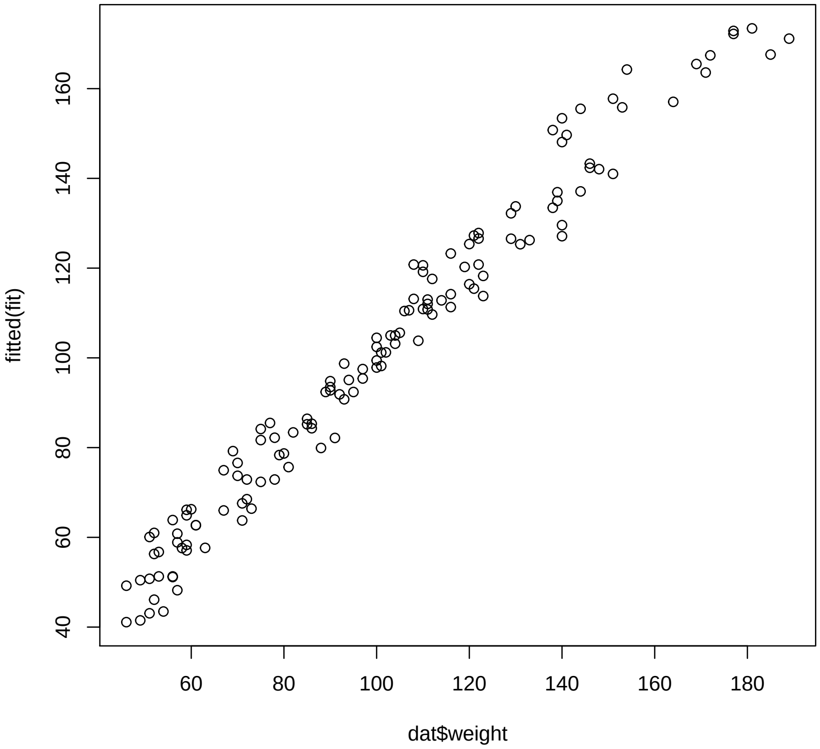
```
par(mfrow = c(1, 1))
boxplot(weight ~ factor(weeks)*treat, data = dat)
```



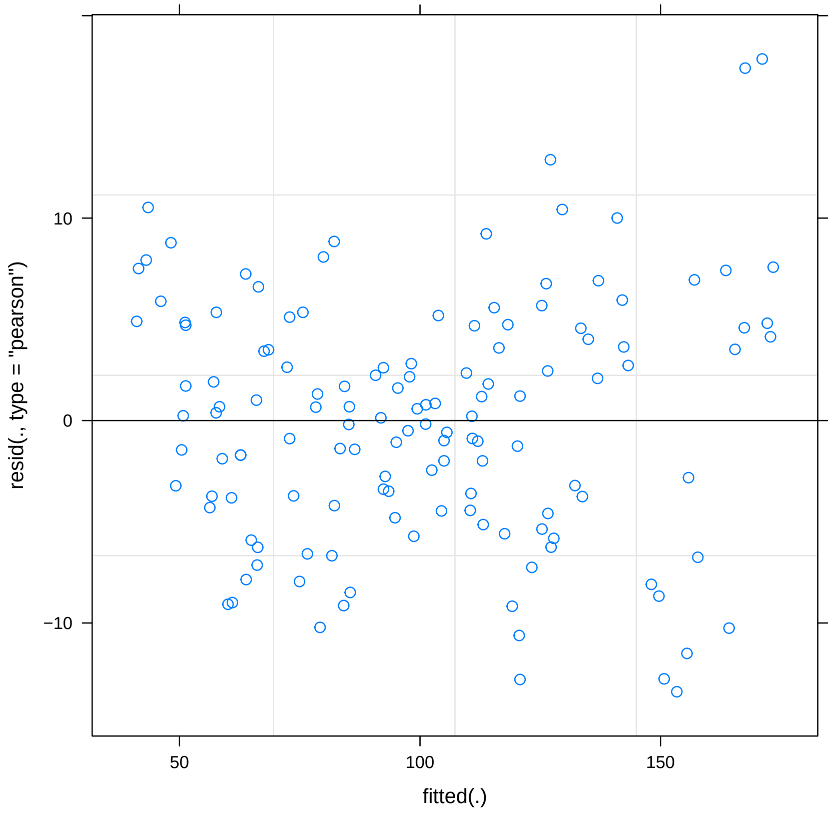
The mean weight becomes larger over time. And the variability of weight change over time gets bigger, treatment thyroxine has the biggest variability over time.

(c)

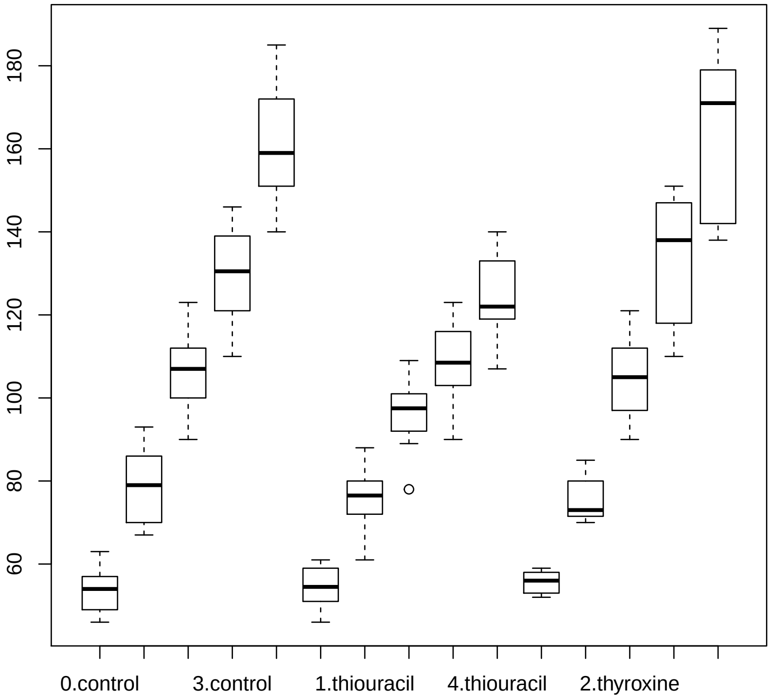
```
  require(lme4)

## Loading required package:  lme4

## Warning:  package 'lme4' was built under R version 3.1.2

## Loading required package:  Rcpp

## Warning:  package 'Rcpp' was built under R version 3.1.2

  fit = lmer(weight ~ factor(weeks) + treat + factor(weeks):treat + (1|subject), data =
  par(mfrow = c(1, 1))
  plot(dat$weight, fitted(fit))
```



```
plot(fit, which = 1)
```



```
par(mfrow = c(1, 1))
boxplot(weight ~ weeks*treat, data = dat)
```



The residuals versus fitted values plots shows no sign for unequal variance.

(d)
```
summary(fit)

## Linear mixed model fit by REML ['lmerMod']
## Formula:
## weight ~ factor(weeks) + treat + factor(weeks):treat + (1 | subject)
##    Data: dat
##
## REML criterion at convergence: 892.1
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.90369 -0.60408  0.03345  0.64957  2.53828
##
## Random effects:
```

```
##  Groups   Name        Variance Std.Dev.
##  subject  (Intercept) 71.55    8.459   
##  Residual             49.51    7.036   
## Number of obs: 135, groups:  subject, 27
## 
## Fixed effects:
##                                Estimate Std. Error t value
## (Intercept)                     54.0000     3.4794   15.52
## factor(weeks)1                  24.5000     3.1468    7.79
## factor(weeks)2                  52.0000     3.1468   16.52
## factor(weeks)3                  76.1000     3.1468   24.18
## factor(weeks)4                 106.6000     3.1468   33.88
## treatthiouracil                  0.7000     4.9206    0.14
## treatthyroxine                   1.5714     5.4222    0.29
## factor(weeks)1:treatthiouracil  -2.9000     4.4503   -0.65
## factor(weeks)2:treatthiouracil -10.9000     4.4503   -2.45
## factor(weeks)3:treatthiouracil -22.4000     4.4503   -5.03
## factor(weeks)4:treatthiouracil -37.1000     4.4503   -8.34
## factor(weeks)1:treatthyroxine   -4.2143     4.9039   -0.86
## factor(weeks)2:treatthyroxine   -2.7143     4.9039   -0.55
## factor(weeks)3:treatthyroxine    1.0429     4.9039    0.21
## factor(weeks)4:treatthyroxine    0.6857     4.9039    0.14
## 
## Correlation of Fixed Effects:
##                   (Intr) fct()1 fct()2 fct()3 fct()4 trtthr trtthy
## factr(wks)1       -0.452                                          
## factr(wks)2       -0.452  0.500                                   
## factr(wks)3       -0.452  0.500  0.500                            
## factr(wks)4       -0.452  0.500  0.500  0.500                     
## treatthircl       -0.707  0.320  0.320  0.320  0.320              
## treatthyrxn       -0.642  0.290  0.290  0.290  0.290  0.454       
## fctr(wks)1:trtthr  0.320 -0.707 -0.354 -0.354 -0.354 -0.452 -0.205
## fctr(wks)2:trtthr  0.320 -0.354 -0.707 -0.354 -0.354 -0.452 -0.205
## fctr(wks)3:trtthr  0.320 -0.354 -0.354 -0.707 -0.354 -0.452 -0.205
## fctr(wks)4:trtthr  0.320 -0.354 -0.354 -0.354 -0.707 -0.452 -0.205
## fctr(wks)1:trtthy  0.290 -0.642 -0.321 -0.321 -0.321 -0.205 -0.452
## fctr(wks)2:trtthy  0.290 -0.321 -0.642 -0.321 -0.321 -0.205 -0.452
## fctr(wks)3:trtthy  0.290 -0.321 -0.321 -0.642 -0.321 -0.205 -0.452
## fctr(wks)4:trtthy  0.290 -0.321 -0.321 -0.321 -0.642 -0.205 -0.452
##                   fctr(wks)1:trtthr fctr(wks)2:trtthr fctr(wks)3:trtthr
## factr(wks)1                                                            
## factr(wks)2                                                            
## factr(wks)3                                                            
## factr(wks)4                                                            
## treatthircl                                                            
## treatthyrxn
## fctr(wks)1:trtthr
## fctr(wks)2:trtthr  0.500
## fctr(wks)3:trtthr  0.500             0.500
## fctr(wks)4:trtthr  0.500             0.500             0.500
## fctr(wks)1:trtthy  0.454             0.227             0.227
## fctr(wks)2:trtthy  0.227             0.454             0.227
## fctr(wks)3:trtthy  0.227             0.227             0.454
## fctr(wks)4:trtthy  0.227             0.227             0.227
##                   fctr(wks)4:trtthr fctr(wks)1:trtthy fctr(wks)2:trtthy
## factr(wks)1
## factr(wks)2
## factr(wks)3
## factr(wks)4
## treatthircl
## treatthyrxn
## fctr(wks)1:trtthr
## fctr(wks)2:trtthr
## fctr(wks)3:trtthr
## fctr(wks)4:trtthr
## fctr(wks)1:trtthy  0.227
## fctr(wks)2:trtthy  0.227             0.500
## fctr(wks)3:trtthy  0.227             0.500             0.500
## fctr(wks)4:trtthy  0.454             0.500             0.500
##                   fctr(wks)3:trtthy
## factr(wks)1
## factr(wks)2
## factr(wks)3
## factr(wks)4
## treatthircl
## treatthyrxn
## fctr(wks)1:trtthr
## fctr(wks)2:trtthr
## fctr(wks)3:trtthr
## fctr(wks)4:trtthr
## fctr(wks)1:trtthy
## fctr(wks)2:trtthy
## fctr(wks)3:trtthy
## fctr(wks)4:trtthy  0.500
```

```
  anova(fit)
```

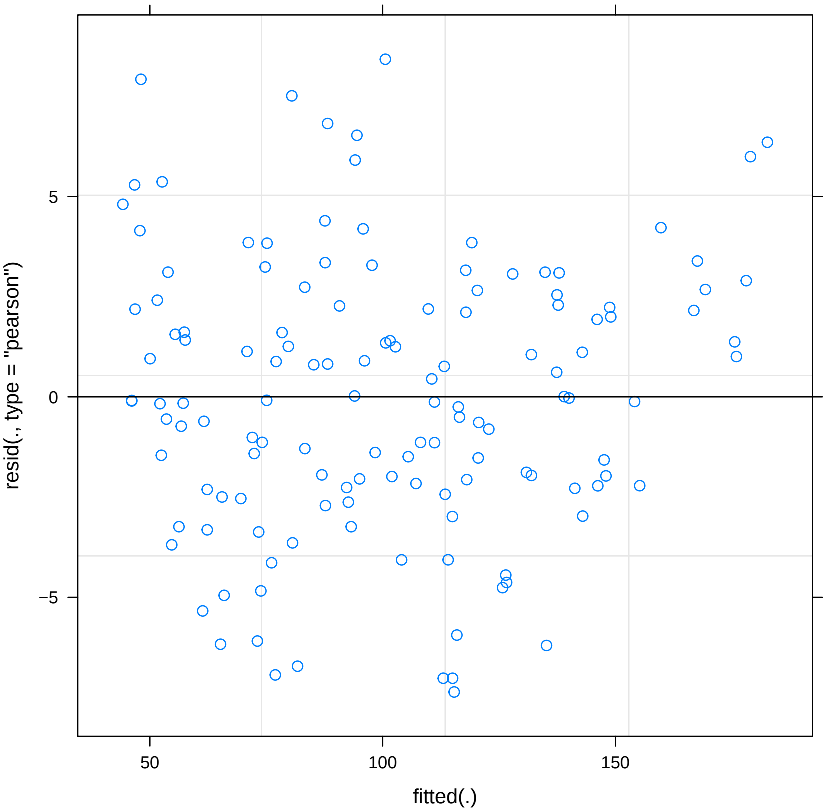
```
## Analysis of Variance Table
##                      Df Sum Sq Mean Sq  F value
## factor(weeks)         4 145188   36297 733.0981
## treat                  2    770     385   7.7774
## factor(weeks):treat  8   6403     800  16.1641

  fit1 = lm(weight ~ factor(weeks) + treat + factor(weeks):treat, data = dat )
  fit2 = lmer(weight ~ treat + factor(weeks):treat + (1|subject), data = dat )
  fit3 = lmer(weight ~ factor(weeks) + factor(weeks):treat + (1|subject), data = dat )
  fit4 = lmer(weight ~ factor(weeks) + treat + (1|subject), data = dat )
  AIC(fit)

## [1] 926.1398

  AIC(fit1)

## [1] 1046.712

  AIC(fit2)

## [1] 926.1398

  AIC(fit3)

## [1] 926.1398

  AIC(fit4)

## [1] 1034.661
```

The AIC of the full model is 926.1398, which means it's the smallest AIC of all. So there's no need to drop the terms.

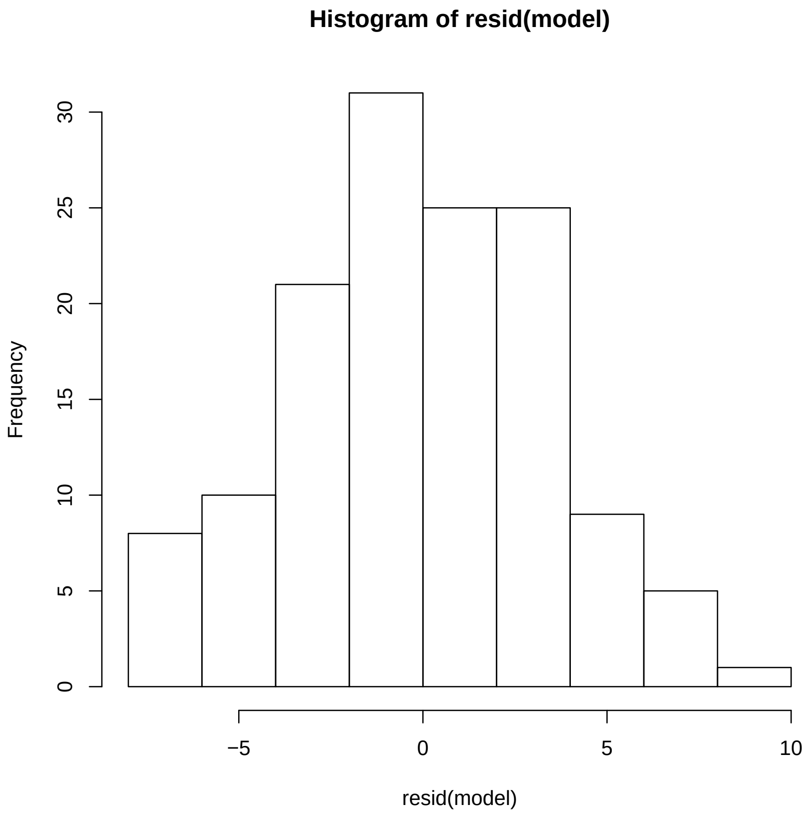
3 3

(a)

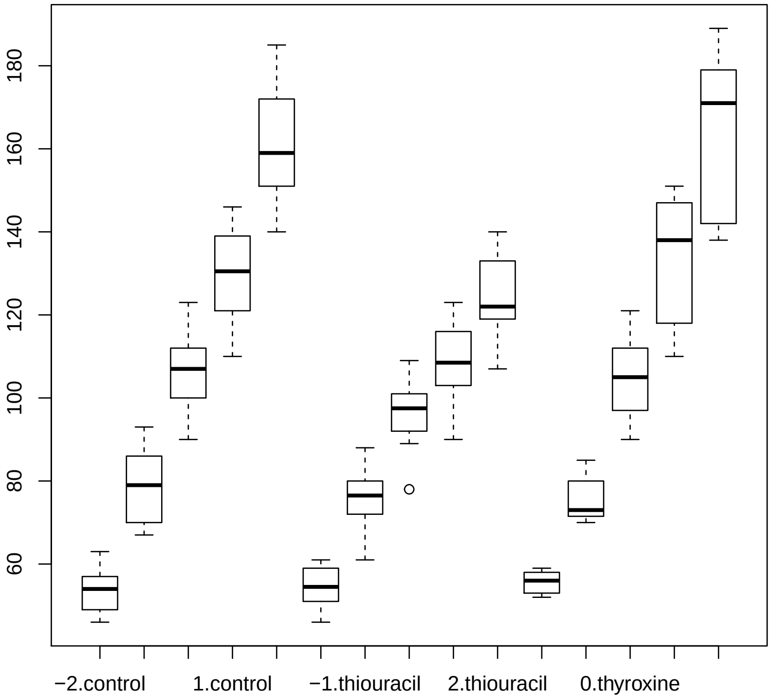
```
  model = lmer(weight ~ wk + treat  + (1|subject) + (0 + wk|subject), data = dat)
  par( mfrow = c(1, 1))
  plot(model, which = 1)
```



```r
hist(resid(model))
```



```
boxplot(weight ~ wk*treat, data = dat)
```



(b)

```
  model2 = lmer(weight ~ wk + wk2 + treat  + (1|subject)+ (0 + wk|subject)+ (0 + wk2|sub
  summary(model2)

## Linear mixed model fit by REML ['lmerMod']
## Formula: weight ~ wk + wk2 + treat + (1 | subject) + (0 + wk | subject) +
##     (0 + wk2 | subject)
##    Data: dat
##
## REML criterion at convergence: 908.3
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.4904 -0.4034  0.0611  0.3895  1.7395
##
## Random effects:
##  Groups    Name        Variance Std.Dev.
##  subject   (Intercept) 87.138   9.335
##  subject.1 wk          36.444   6.037
##  subject.2 wk2          2.042   1.429
##  Residual               9.402   3.066
## Number of obs: 135, groups:  subject, 27
##
## Fixed effects:
##                   Estimate Std. Error t value
## (Intercept)     104.87487    3.02136   34.71
## wk               23.18148    1.17669   19.70
## wk2               0.08201    0.31704    0.26
## treatthiouracil -11.04663    4.26709   -2.59
## treatthyroxine   -0.54055    4.70210   -0.11
##
## Correlation of Fixed Effects:
##             (Intr) wk     wk2    trtthr
## wk           0.000
## wk2         -0.052  0.000
## treatthircl -0.706  0.000  0.000
## treatthyrxn -0.641  0.000  0.000  0.454
```

No, they seems to be different, the second model has term $I(weeks^2)$.

(c)

```
  AIC(model)

## [1] 939.0791

  AIC(model2)

## [1] 926.3378
```

The AIC of the first model is 939.0791, which is bigger than 926.3378(second model). So that, we should choose the first model with wk2.

```
  summary(model2)

## Linear mixed model fit by REML ['lmerMod']
## Formula: weight ~ wk + wk2 + treat + (1 | subject) + (0 + wk | subject) +
##     (0 + wk2 | subject)
##    Data: dat
##
## REML criterion at convergence: 908.3
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.4904 -0.4034  0.0611  0.3895  1.7395
##
## Random effects:
##  Groups    Name        Variance Std.Dev.
##  subject   (Intercept) 87.138   9.335
##  subject.1 wk          36.444   6.037
##  subject.2 wk2          2.042   1.429
##  Residual               9.402   3.066
## Number of obs: 135, groups:  subject, 27
##
## Fixed effects:
##                  Estimate Std. Error t value
## (Intercept)     104.87487    3.02136   34.71
## wk               23.18148    1.17669   19.70
## wk2               0.08201    0.31704    0.26
## treatthiouracil -11.04663    4.26709   -2.59
## treatthyroxine   -0.54055    4.70210   -0.11
##
## Correlation of Fixed Effects:
##             (Intr) wk     wk2    trtthr
## wk           0.000
## wk2         -0.052  0.000
## treatthircl -0.706  0.000  0.000
## treatthyrxn -0.641  0.000  0.000  0.454
```

(d)

```
  model3 = lmer(weight ~ weeks + I(weeks^2) + treat  + (1|subject) + (0 + weeks|subject)
## Warning in Ops.factor(weeks, 2):  ^ not meaningful for factors
## Error:  Invalid grouping factor specification, subject
```

It seems that the slopes may depend on the treatment.

(e)

```
  summary(model3)
## Error in summary(model3):  error in evaluating the argument ’object’
in selecting a method for function ’summary’:  Error:  object ’model3’
not found

  drop1(model3, )

## Error in drop1(model3, ):  object ’model3’ not found
```

4 4

(a)

$$\begin{aligned}
E(Y_{ij}) &= E(\mu + \rho_i + \beta_1 x_i + \gamma_1 t_j + \epsilon_{ij}) \\
&= \mu + \beta_1 x_i + \gamma_1 t_j \\
Var(Y_{ij}) &= var(\mu + \rho_i + \beta_1 x_i + \gamma_1 t_j + \epsilon_{ij}) \\
&= var(\rho_i + \epsilon_{ij}) \\
&= \sigma_\rho^2 + \sigma^2 (\text{since } \rho_i \text{ and } \epsilon_{ij} \text{ are independent}) \\
Cov(Y_{ij}, Y_{ij'}) &= E((Y_{ij} - E(Y_{ij}))(Y_{ij'} - E(Y_{ij'})) \\
&= E((\rho_i + \epsilon_{ij})(\rho_i + \epsilon_{ij'})) \\
&= E(\rho_i^2)(\text{since } \rho_i \text{ and } \epsilon_{ij} \text{ and } \epsilon_{ij'} \text{ are independent}) \\
&= Var(\rho_i) + (E(\rho_i))^2 \\
&= \sigma_\rho^2 \\
Corr(Y_{ij}, Y_{ij'}) &= \frac{Cov(Y_{ij}, Y_{ij'})}{\sqrt{Var(Y_{ij}) * Var(Y_{ij'})}} \\
&= \frac{\sigma_\rho^2}{\sigma_\rho^2 + \sigma^2}
\end{aligned}$$

(b)

$$\begin{aligned}
E(Y_{ij}) &= E(\mu + \rho_i + \beta_1 x_i + \gamma_1 t_j + \gamma_{i1} t_j + \epsilon_{ij}) \\
&= \mu + \beta_1 x_i + \gamma_1 t_j \\
Var(Y_{ij}) &= var(\mu + \rho_i + \beta_1 x_i + \gamma_1 t_j + \gamma_{i1} t_j + \epsilon_{ij}) \\
&= var(\rho_i + \gamma_{i1} t_j + \epsilon_{ij}) \\
&= \sigma_\rho^2 + t_j^2 \sigma_{\gamma 1}^2 + \sigma^2 (\text{since } \rho_i,\ \gamma_{i1} \text{ and } \epsilon_{ij} \text{ are independent}) \\
Cov(Y_{ij}, Y_{ij'}) &= E((Y_{ij} - E(Y_{ij}))(Y_{ij'} - E(Y_{ij'})) \\
&= E((\rho_i + \gamma_{i1} t_j + \epsilon_{ij})(\rho_i + \gamma_{i1} t_{j'} + \epsilon_{ij'})) \\
&= E(\rho_i^2 + \gamma_{i1}^2 t_j * t_{j'})(\text{since } \rho_i,\ \gamma_{i1},\ \epsilon_{ij} \text{ and } \epsilon_{ij'} \text{ are independent}) \\
&= E(\rho_i^2) + (t_j * t_{j'}) * E(\gamma_{i1}^2) \\
&= \sigma_\rho^2 + (t_j * t_{j'}) \sigma_{\gamma 1}^2 \\
Corr(Y_{ij}, Y_{ij'}) &= \frac{Cov(Y_{ij}, Y_{ij'})}{\sqrt{Var(Y_{ij}) * Var(Y_{ij'})}} \\
&= \frac{\sigma_\rho^2 + (t_j * t_{j'}) \sigma_{\gamma 1}^2}{\sqrt{(\sigma_\rho^2 + t_j^2 \sigma_{\gamma 1}^2 + \sigma^2)(\sigma_\rho^2 + t_{j'}^2 \sigma_{\gamma 1}^2 + \sigma^2)}}
\end{aligned}$$