STAT 528 HW5

Problem 1

Do the following with the pulp data set:

Solution 1:

a. Analyze the pulp data in the notes using unrestricted maximum likelihood estimation and comment on the differences that such analysis produces when compared to the REML and ANOVA estimates.

Analyzing the pulp data in the notes using unrestricted maximum likelihood estimation:

```
library(faraway)
library(lme4)
```

```
## Loading required package: Matrix
```

```
library(dplyr)

data(pulp)
mod = lmer(bright ~ 1 + (1|operator), pulp,REML = FALSE)
summary(mod)
```

```
## Linear mixed model fit by maximum likelihood  ['lmerMod']
## Formula: bright ~ 1 + (1 | operator)
##    Data: pulp
##
##      AIC      BIC   logLik deviance df.resid
##     22.5     25.5     -8.3     16.5       17
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.50554 -0.78116 -0.06353  0.65850  1.56232
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  operator (Intercept) 0.04575  0.2139
##  Residual             0.10625  0.3260
## Number of obs: 20, groups:  operator, 4
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)  60.4000     0.1294   466.7
```

Comparing it with REML

```
mmod <- lmer(bright ~ 1 + (1|operator), pulp)
summary(mmod)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: bright ~ 1 + (1 | operator)
##    Data: pulp
##
## REML criterion at convergence: 18.6
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.4666 -0.7595 -0.1244  0.6281  1.6012
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  operator (Intercept) 0.06808  0.2609
##  Residual             0.10625  0.3260
## Number of obs: 20, groups:  operator, 4
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)  60.4000     0.1494   404.2
```

We know that the problem with unrestricted MLE estimation is that the estimates of the variance are biased. Thus as expected the variance and standard errors of the fixed and random effects are different. But the estimates for the residual and the estimate of the fixed effect are the same when calculated using both methods. This leads to different t-values in the two methods.

Comparing it with ANOVA

```
op <- options(contrasts = c("contr.sum", "contr.poly"))
lmod <- aov(bright ~ operator, pulp)
summary(lmod)
```

```
##             Df Sum Sq Mean Sq F value Pr(>F)
## operator     3   1.34  0.4467   4.204 0.0226 *
## Residuals   16   1.70  0.1062
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
coef(lmod)
```

```
## (Intercept)   operator1   operator2   operator3
##       60.40       -0.16       -0.34        0.22
```

The operator variance from this model is

```
## Operator variance - (MSA - MSE) / n
(0.447 - 0.106)/5
```

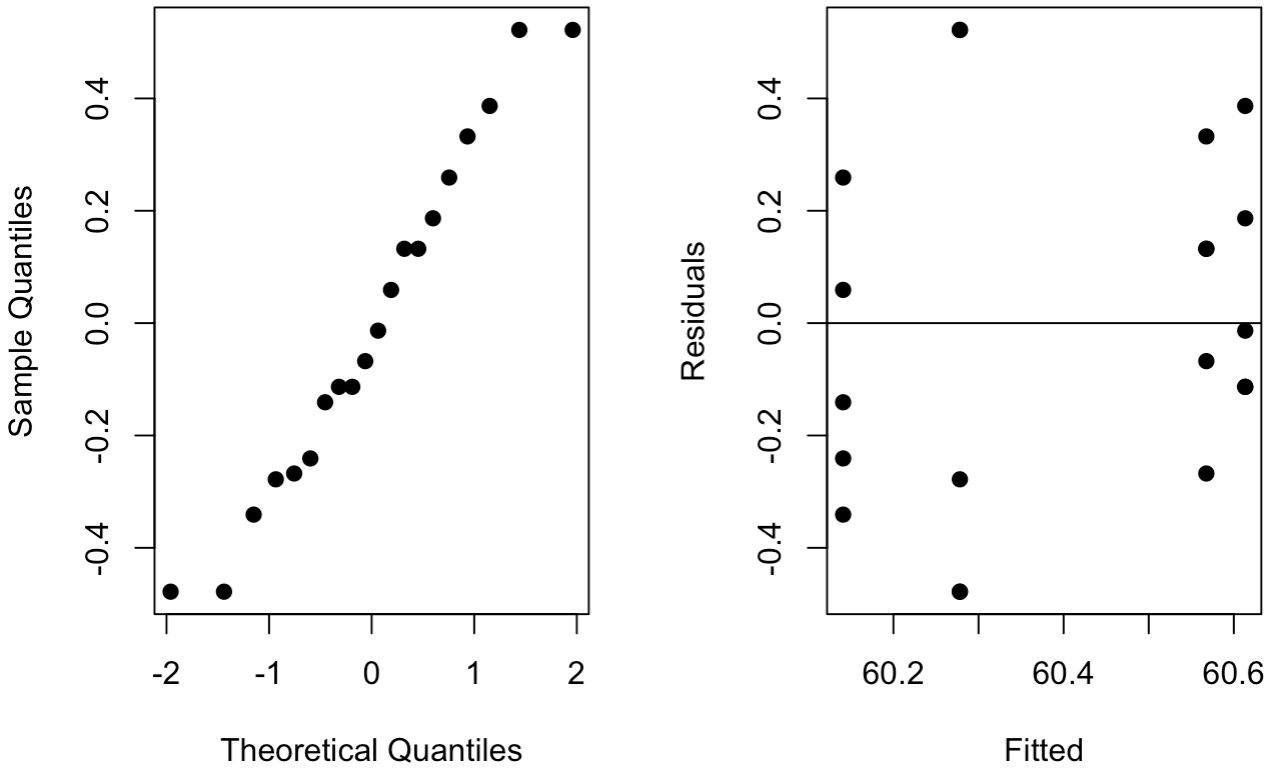
```
## [1] 0.0682
```

Where the residual variance is 0.106

Comparing it with the unrestricted likelihood model we can see that the estimate for the intercept (the fixed effect) is still exactly the same, so is the estimate for the residual variance. The estimates for the operator variance differs. But note that the value of the variance using the anova model is a lot closer to the value that we estimate using unrestricted MLE as opposed to REML, which means this is also biased.

b. Interpret the diagnostic plots in the notes corresponding to the analysis with respect to the pulp data set; conclude that these plots indicate no particular problems.

```
par(mfrow = c(1,2))
qqnorm(resid(mmod), main="", pch = 19)
plot(fitted(mmod),resid(mmod), xlab="Fitted", ylab="Residuals", pch = 19)
abline(0,0)
```



We can see that the plot of the theoretical quantiles against the sample quantiles is a straight line which means that our model follows the distributional assumptions.

From the fitted vs residuals plot we can see that the smoothing line is a straight line. This means that the linearity assumption is reasonable and the data is homoskedastic. Also from the spread of the points around the smoothing line there seems to be no outliers in the data. Thus the plots show that our data has no particular problems.

Problem 2

Do the following:

Solution 2:

a. Compare and contrast the nonparametric bootstrap, residual bootstrap, and parametric bootstraps. Discuss the assumptions that make each bootstrap procedure appropriate.

Parametric Bootstrap: We estimate a model and then simulate from the estimated model. The parametric bootstrap assumes that the model we estimate is perfectly correct for some parameter value. The advantage of this method is that if the parametric model really is correct, we can get more precise results using parametric bootstrap.

Residual Bootstrap: We first estimate the model, and then simulate by resampling the residuals to that estimate and adding them back to the fitted values. This type of bootstrap does not trust the model as much as the parametric bootstrap. It assumes that the shape of the regression function is right but does not make any distributional assumptions on the residuals. This makes it more secure than parametric bootstrap. If we are correct about the shape of the curve then resampling the residuals gives more precise results than the next (non-parametric) bootstrap method.

Nonparametric Bootstrap: We resample whole rows from the dataset. This method does not involve the estimated model in any way. The only assumption that this method makes is that the observations are independent, making it the safest. But the reason we do not always want to use the safest bootstrap is because it gives the widest confidence intervals. The residual bootstrap gives narrower CIs whereas the parametric bootstrap gives the narrowest.

b. In the notes we developed a parametric bootstrap procedure to approximate the distribution of the LRT corresponding to a test between mixed-effects models. Write your own parametric bootstrap procedure with B = 1e4 samples and make it as fast as possible using parallel programming with either mclapply and/or foreach and whatever accompanying software packages are needed.

```
## null model fit
null_mod <- lm(bright ~ 1, pulp)
```

Doing the parametric bootstrap procedure sequentially

```
set.seed(2609)
B <- 1e4
lrtstat <- numeric(B)
system.time(for(b in 1:B){
        y <- unlist(simulate(null_mod))
        beta_null <- lm(y ~ 1)
        beta_alt <- lmer(y ~ 1 + (1|operator), pulp, REML = FALSE,control = lmerControl
(check.conv.singular = .makeCC(action = "ignore",tol = 1e-4)))
        lrtstat[b] <- as.numeric(2*(logLik(beta_alt) - logLik(beta_null)))
})
```

```
##    user  system elapsed
## 115.963   0.388 116.941
```

The proportion of LRTs close to zer0

```
mean(lrtstat < 1e-5)
```

```
## [1] 0.7041
```

The estimated p-value is

```
## p-value
pval <- mean(lrtstat > 2.568371)
pval
```

```
## [1] 0.0224
```

Doing the parametric bootstrap procedure using parallel computing.

```
library(foreach)
library(doParallel)
```

```
## Loading required package: iterators
```

```
## Loading required package: parallel
```

```
cores=detectCores()
registerDoParallel(cores)


set.seed(2609)
B <- 1e4

system.time(lrtstat <- foreach(b=1:B,.combine = c) %dopar%{
        y <- unlist(simulate(null_mod))
        beta_null <- lm(y ~ 1)
        beta_alt <- lme4::lmer(y ~ 1 + (1|operator), pulp, REML = FALSE,control = lme4::
lmerControl(check.conv.singular = lme4::.makeCC(action = "ignore",tol = 1e-4)))
        as.numeric(2*(logLik(beta_alt) - logLik(beta_null)))
})
```

```
##    user  system elapsed
## 233.201   6.493  23.897
```

```
stopImplicitCluster()
```

The proportion of LRTs close to zero

```
mean(lrtstat < 1e-5)
```

```
## [1] 0.7024
```

The estimated p-value is

```
## p-value
pval <- mean(lrtstat > 2.568371)
pval
```

```
## [1] 0.021
```

Clearly the results are the same (the small difference is due to the randomization) but the process is a lot faster!

c. Explain how the testing procedure using the exactRLRT function works. See the analysis of the irrigation data in the LMM course notes for context.

Our goal is to test whether a random effect is different from zero. Since according to the distributional assumption we make the expectation of these random effects = 0, this is equivalent to testing if the variance of the random effect is different from zero. Thus our test is:

$$H_0 : \sigma_\alpha^2 = 0 \quad \text{vs.} \quad H_1 : \sigma_\alpha^2 \neq 0$$

The problem with using the typical LRT test is that the null hypothesis lies on the boundary of the parameter space and the observations are not iid due to the particular covariance structure of the grouped data. This violates the regularity conditions of the LRT and the the LRT statistics does not have an asymptotic χ^2 distribution.

Thus we consider the RLRT statistic, which is the test statistic based on Restricted Maximum Likelihood (REML) estimation of the variance components.

Consider a test of the given structure for an LMM with only one random effect (vector) and i.i.d. errors. In this case, the exact RLRT distribution under H_0 can be expressed (with $\lambda = \sigma_\alpha^2/\sigma_\varepsilon^2$)

$$RLRT_n \overset{d}{=} \sup_{\lambda \geq 0} \left\{ (n-p)\log\left[1 + \frac{N_n(\lambda)}{D_n(\lambda)}\right] - \sum_{l=1}^{K} \log(1+\lambda_{\mu_{l,n}}) \right\}$$

where

$$N_n(\lambda) = \sum_{l=1}^{K} \frac{\lambda_{\mu_{l,n}}}{1+\lambda_{\mu_{l,n}}} \omega_l^2, \quad D_n(\lambda) = \sum_{l=1}^{K} \frac{\omega_l^2}{1+\lambda_{\mu_{l,n}}} + \sum_{l=K+1}^{n-p} \omega_l^2$$

Here,$\mu_{l,n}$, $l = 1, \ldots, K_s$, are the eigen- values of the matrix $\Sigma_s^{\frac{1}{2}} Z_s'(I_n - X(X'X)^{-1}X')Z_s\Sigma_s^{\frac{1}{2}}$ and $\omega_l \overset{iid}{\sim} N(0,1)$, $l = 1, \ldots, n-p$.

This distribution only depends on the design matrices of the fixed and random effects, X and Z_s, and on the correlation structure within the random effects vector, Σ_s.

Critical values or p-values of the distribution of $RLRT_n$ can be determined efficiently by simulation, which is implemented in the RLRsim package in R

Problem 3

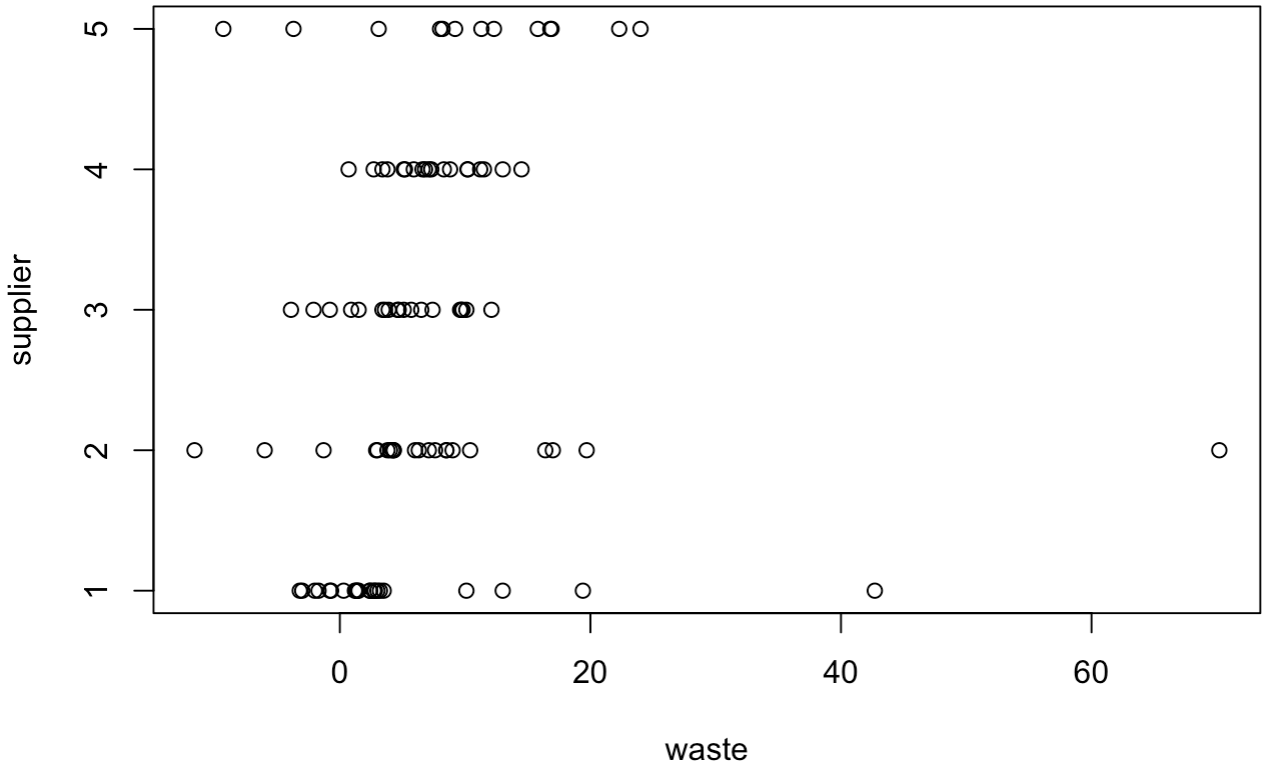
Analyzing the `denim` dataset

Solution 3:

```
data(denim)
```

a. Plot the data and comment.

```
plot(denim)
```



From the plot there seems to be two outliers. Leaving those out the variability of the waste due to supplier 1,3 and 4 seem to be lesser than that due to supplier 2,5

b. Fit the linear fixed effects model. Is the operator significant?

```
fixed_mod = lm(waste ~ ., data = denim)
summary(fixed_mod)
```

```
##
## Call:
## lm(formula = waste ~ ., data = denim)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -20.432  -4.377  -1.323   2.639  61.368
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    7.210      1.030   6.997  4.5e-10 ***
## supplier1     -2.688      1.927  -1.395    0.167
## supplier2      1.621      1.927   0.841    0.402
## supplier3     -2.379      2.033  -1.170    0.245
## supplier4      0.279      2.033   0.137    0.891
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.86 on 90 degrees of freedom
## Multiple R-squared:  0.04901,    Adjusted R-squared:  0.006747
## F-statistic:  1.16 on 4 and 90 DF,  p-value: 0.334
```

The suppliers do not seem to be significant. We can also compare the null model to this model to check its significance.

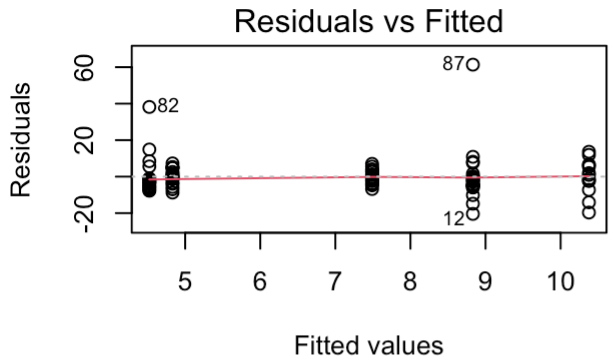
```
null_mod = lm(waste~1, data = denim)
anova(null_mod,fixed_mod)
```

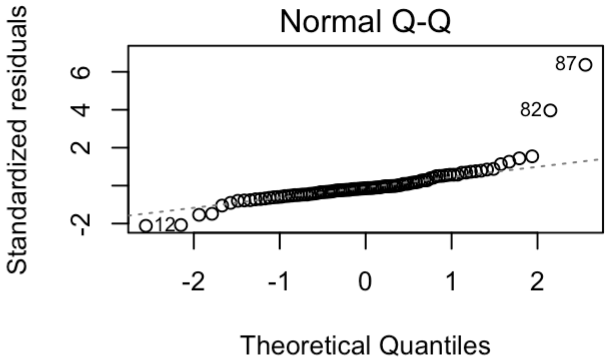
```
## Analysis of Variance Table
##
## Model 1: waste ~ 1
## Model 2: waste ~ supplier
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1     94 9200.0
## 2     90 8749.1  4    450.92 1.1596  0.334
```

The null model is preffered which gives further evidence that the suppliers are not significant.

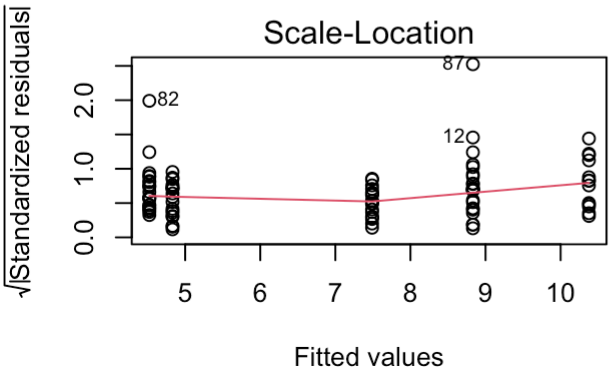
c. Make a useful diagnostic plot for this model and comment.

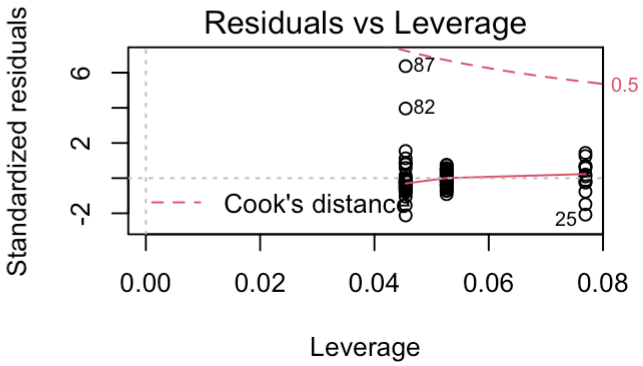
```
par(mfrow = c(2,2))
plot(fixed_mod)
```





The





residuals vs fitted plots form a straight smoothing line which means that the linearity assumption is fair. But we can see that the 82nd and 87th observations are outliers. From the QQ plot we can see that the normality assumptions are satisfied and the scale location plot is almost straight, indicating that the homoskedastic assumption is also a fair one (the slight curve can be attributed to the presence of the outliers). The last plot again points to the two outlying observations.

d. Analyze the data with supplier as a random effect. What are the estimated standard deviations of the effects?

```
library(lme4)
mixed_effects = lmer(waste ~ 1+(1|supplier),denim)
summary(mixed_effects)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: waste ~ 1 + (1 | supplier)
##    Data: denim
##
## REML criterion at convergence: 702.1
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.9095 -0.4363 -0.1669  0.3142  6.3817
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  supplier (Intercept)  0.6711  0.8192
##  Residual             97.3350  9.8658
## Number of obs: 95, groups:  supplier, 5
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)    6.997      1.078    6.49
```

Thus $\sigma_\alpha = 0.8192$ and $\sigma_\varepsilon = 9.8658$

e. Test the significance of the supplier term.

```
library(RLRsim)
exactRLRT(mixed_effects)
```

```
##
##  simulated finite sample distribution of RLRT.
##
##  (p-value based on 10000 simulated values)
##
## data:
## RLRT = 0.029383, p-value = 0.3439
```

It still does not seem to be significant.

f. Compute confidence intervals for the random effect SDs

```
confint(mixed_effects,method = "boot")
```

```
## Computing bootstrap confidence intervals ...
```

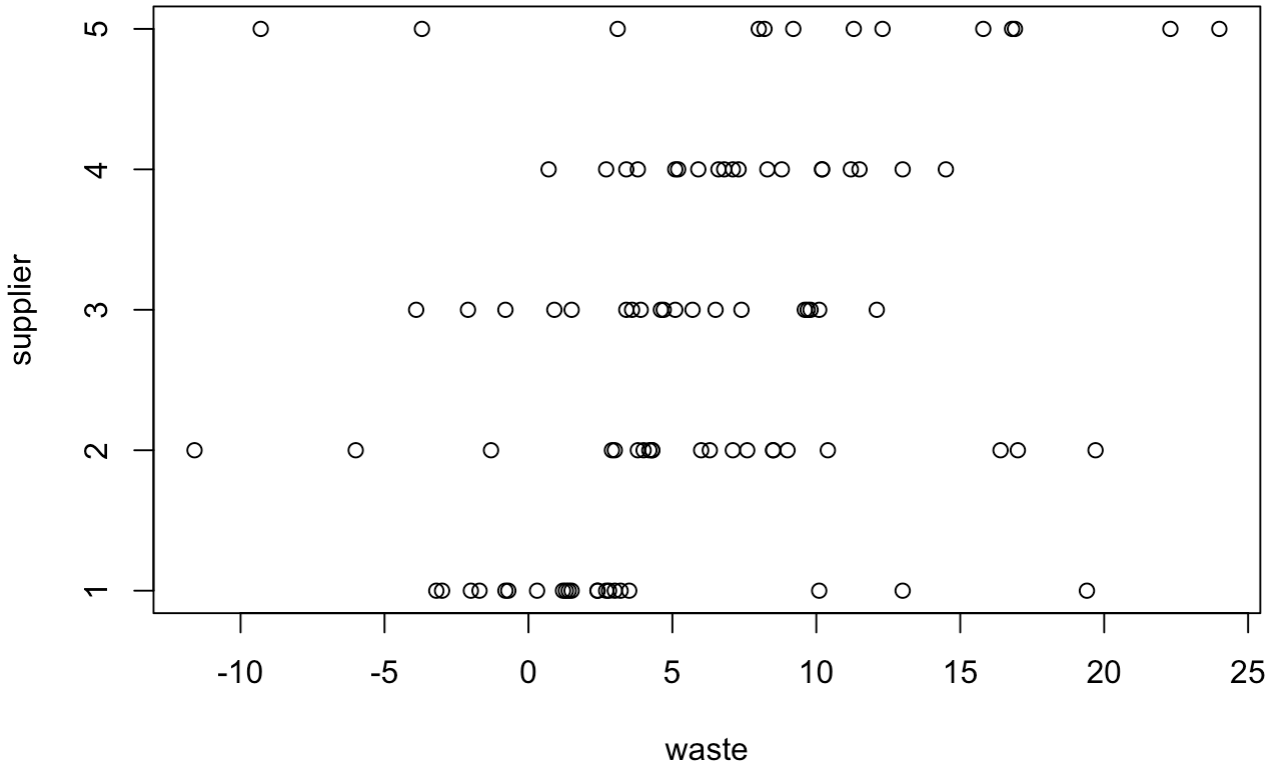
```
##
## 289 message(s): boundary (singular) fit: see ?isSingular
```

```
##                 2.5 %    97.5 %
## .sig01      0.000000  3.081810
## .sigma      8.433525 11.257427
## (Intercept) 4.836143  9.121049
```

g. Locate two outliers and remove them from the data. Repeat the fitting, testing and computation of the confidence intervals, commenting on the differences you see from the complete data.

We have seen from the above analysis that the 82nd and 87th observations are outliers. Thus we remove them from our dataset

```
denim_new = denim[-c(82,87),]
plot(denim_new)
```



The outliers seem to be removed. Let us repeat the above analysis.

First let us create a linear fixed effects model.

```
fixed_mod_new = lm(waste ~ ., data = denim_new)
summary(fixed_mod_new)
```

```
##
## Call:
## lm(formula = waste ~ ., data = denim_new)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -19.6769  -2.4048  -0.3048   2.7105  16.6952
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   6.2625     0.6428   9.742 1.22e-15 ***
## supplier1    -3.5577     1.2154  -2.927  0.00435 **
## supplier2    -0.3529     1.2154  -0.290  0.77220
## supplier3    -1.4309     1.2606  -1.135  0.25942
## supplier4     1.2270     1.2606   0.973  0.33304
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.102 on 88 degrees of freedom
## Multiple R-squared:  0.1425, Adjusted R-squared:  0.1035
## F-statistic: 3.657 on 4 and 88 DF,  p-value: 0.008383
```

Supplier 5 now seems to be significant.

Let us next apply fit the mixed effects model.

```
mixed_effects_new = lmer(waste ~ 1+(1|supplier),denim_new)
summary(mixed_effects_new)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: waste ~ 1 + (1 | supplier)
##    Data: denim_new
##
## REML criterion at convergence: 603.9
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.99119 -0.48597 -0.08981  0.49970  2.60002
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  supplier (Intercept)  5.718   2.391
##  Residual             37.292   6.107
## Number of obs: 93, groups:  supplier, 5
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)    6.155      1.246   4.938
```

The standard deviation of the suppliers effect has increased and the residual standard error has now decreased. This implies that a greater portion of the variability is now explained by the suppliers.

Let us test the significance of this mixed effect.

```
exactRLRT(mixed_effects_new)
```

```
##
##  simulated finite sample distribution of RLRT.
##
##  (p-value based on 10000 simulated values)
##
## data:
## RLRT = 4.5674, p-value = 0.0101
```

Clearly now the suppliers effect is significant.

We next compute the cofidence intervals.

```
confint(mixed_effects_new,method = "boot")
```

```
## Computing bootstrap confidence intervals ...
```

```
##
## 51 message(s): boundary (singular) fit: see ?isSingular
```

```
##                2.5 %   97.5 %
## .sig01      0.000000 4.363196
## .sigma      5.197841 6.987394
## (Intercept) 3.501615 8.570003
```

The interval still has 0 despite the test being significant.

h. Estimate the effect of each supplier. If only one supplier will be used, choose the best.

```
ranef(mixed_effects_new)$supplier
```

```
##   (Intercept)
## 1  -2.6325749
## 2  -0.1872530
## 3  -0.9851799
## 4   0.9935099
## 5   2.8114979
```

Since the first supplier has the smallest estimated waste, I would choose it as the best.

Problem 4

Load in the dataset and analyze the response variable (labeled mtqM). Refer to the soybean analysis in the course notes, and recall that the researchers are interested in determining which ID variables differ from the RC reference level. Consider response transformations if modeling assumptions are violated. Do attempts to rectify departures from modeling assumptions affect the conclusions?

Solution 4:

```
dat <- read.csv("/Users/diptarka/Documents/GitHub/stat528resources/homework/HW5/soybean_
full.csv")
dat$ID <- as.factor(dat$ID)
head(dat)
```

```
##         Date    ID plot_number disk       AqI      AqE     tqE      AqM     tqM
## 1 2021-06-24 NAM10         123   30 0.2624030 2.154646 1.27040 1.432582 12.9961
## 2 2021-06-24 NAM10          70   68 0.3043593 2.545618 0.88499 1.719984 14.4052
## 3 2021-06-24 NAM10         136   21 0.3896498 2.205999 0.94881 1.644281 14.5132
## 4 2021-06-24 NAM10         136   15 0.2554035 2.042392 1.00720 1.812088 15.1677
## 5 2021-06-24 NAM10          70   59 0.3862977 2.229652 0.90931 1.624016 15.5010
## 6 2021-06-24 NAM10          70   52 0.2733171 2.255601 1.13010 1.610873 15.6934
##   maxNPQ   protocol       Ta      VPD Precip      Fsd  Ta_7day  VPD_7day
## 1   3.85 Protocol_3 24.31523 1.134132  31.75 381.4252 21.62982 0.9447707
## 2   4.57 Protocol_3 24.31523 1.134132  31.75 381.4252 21.62982 0.9447707
## 3   4.24 Protocol_3 24.31523 1.134132  31.75 381.4252 21.62982 0.9447707
## 4   4.11 Protocol_3 24.31523 1.134132  31.75 381.4252 21.62982 0.9447707
## 5   4.24 Protocol_3 24.31523 1.134132  31.75 381.4252 21.62982 0.9447707
## 6   4.14 Protocol_3 24.31523 1.134132  31.75 381.4252 21.62982 0.9447707
##   Precip_7day Fsd_7day Precip_7day_sum Precip_cum year
## 1    4.966305 511.1265        34.76413   896.4501 2021
## 2    4.966305 511.1265        34.76413   896.4501 2021
## 3    4.966305 511.1265        34.76413   896.4501 2021
## 4    4.966305 511.1265        34.76413   896.4501 2021
## 5    4.966305 511.1265        34.76413   896.4501 2021
## 6    4.966305 511.1265        34.76413   896.4501 2021
```

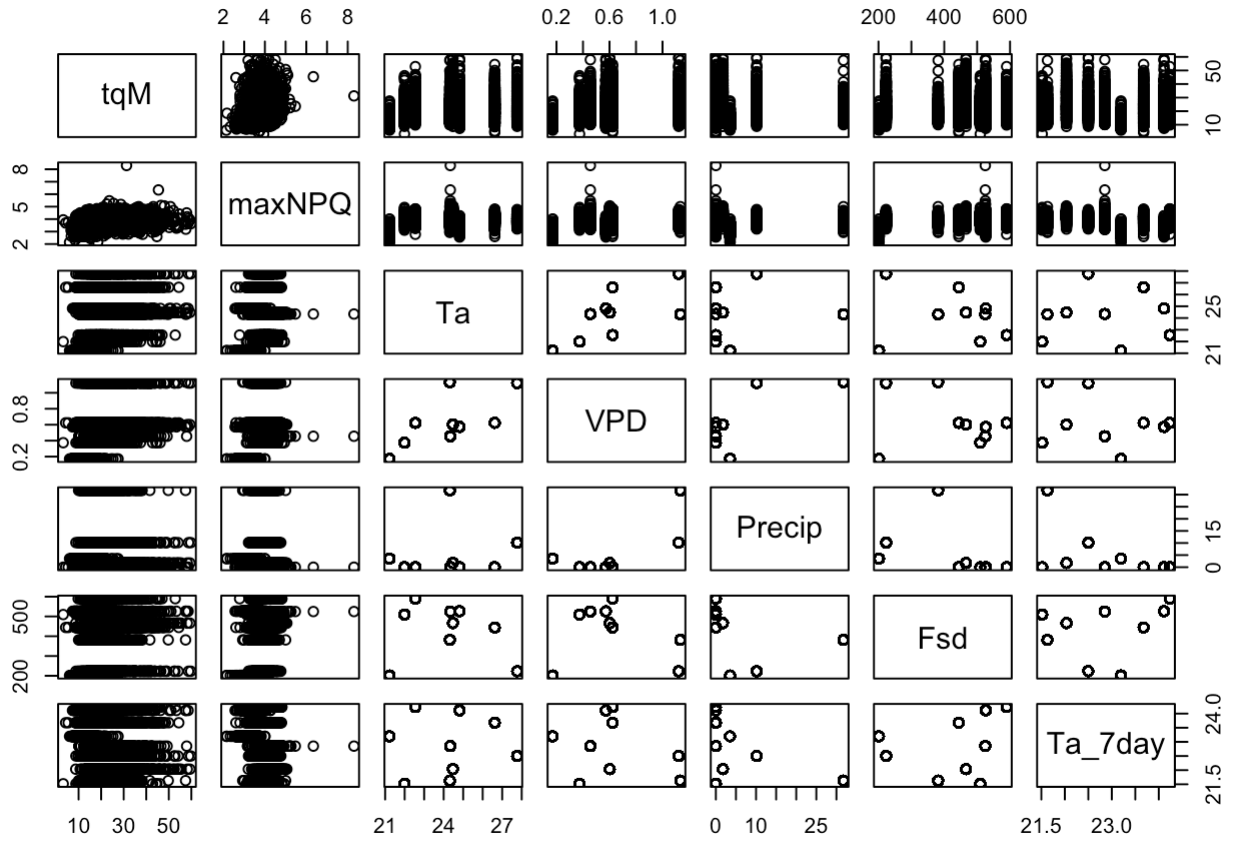
EDA

```
range(dat$tqM)
```

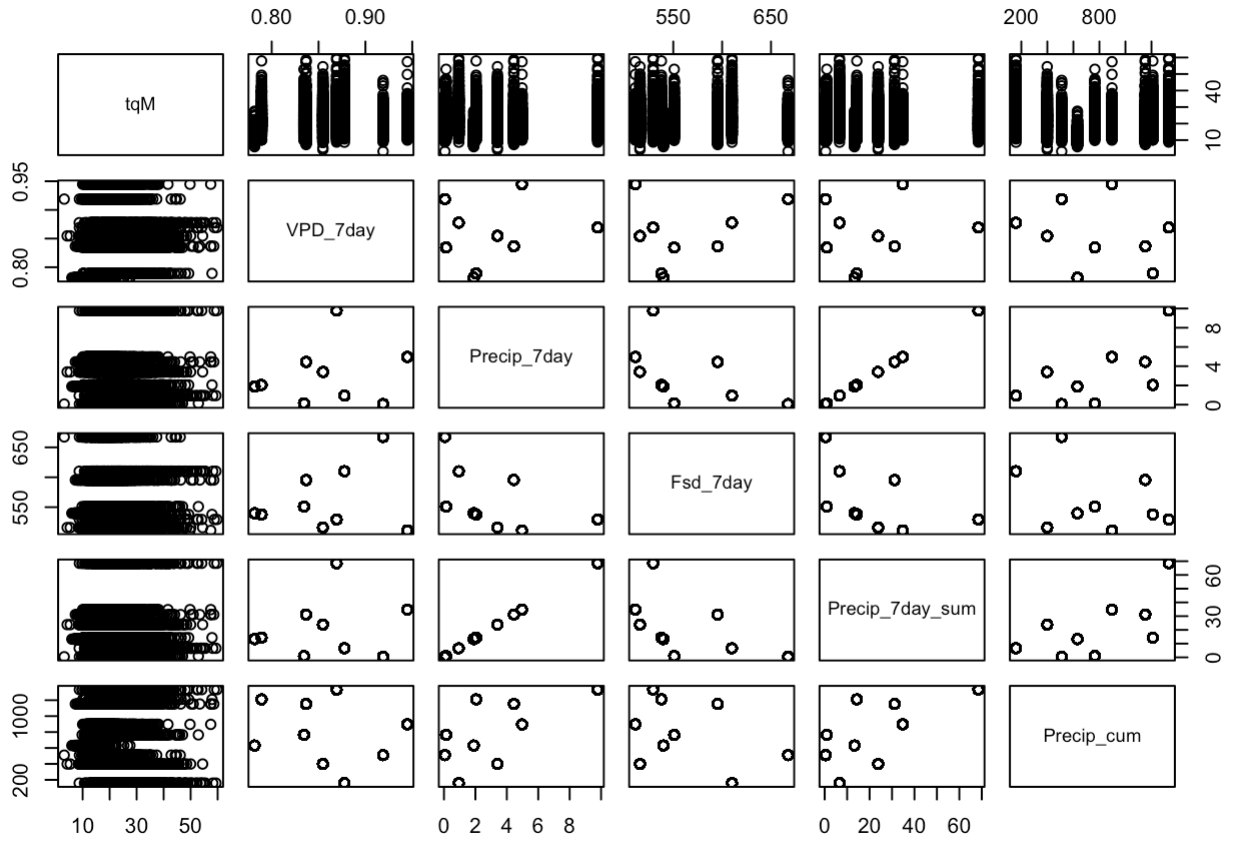
```
## [1]  3.1739 59.7599
```

Let's see the correlation of variables.

```
pairs(dat[,c(9:10, 12:16)])
```

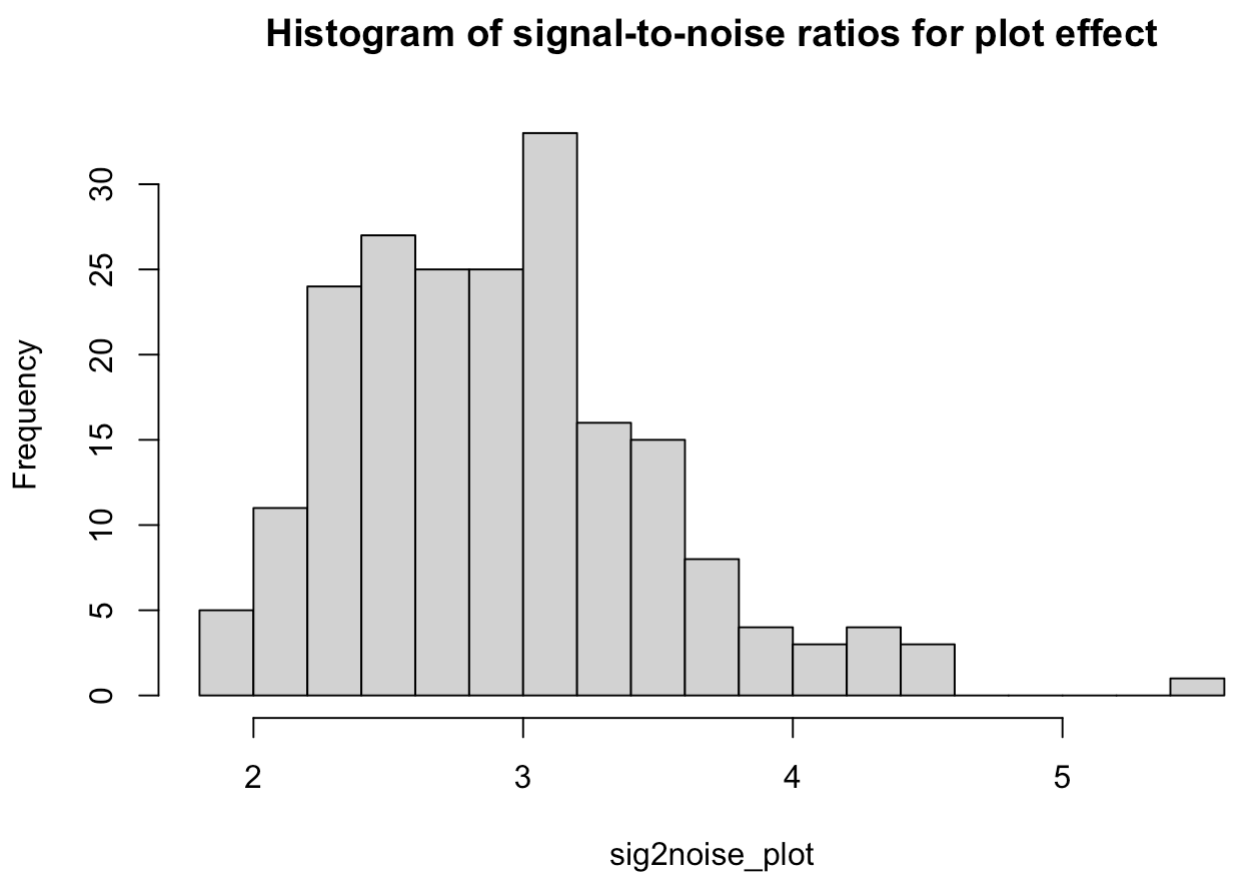



```
pairs(dat[,c(9, 17:21)])
```

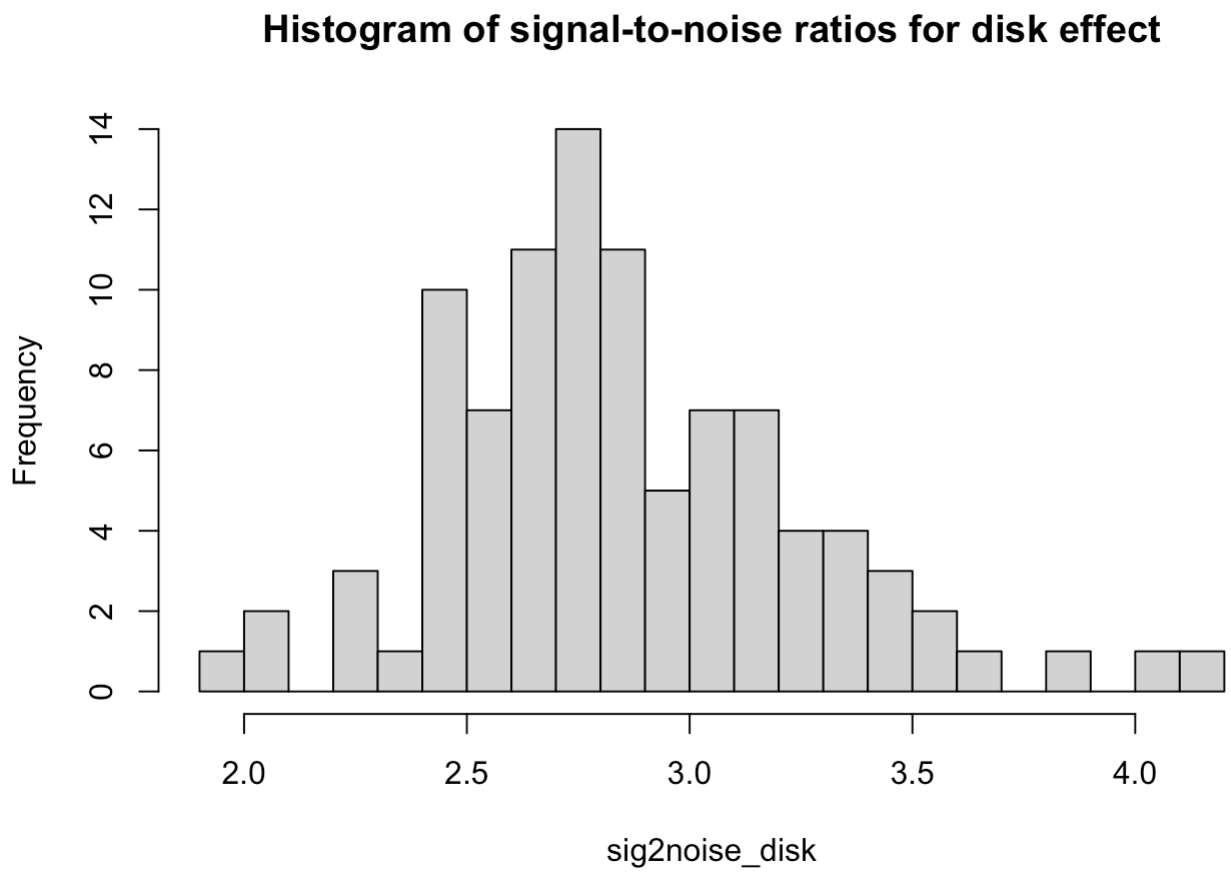


And the signal-to-noise ratio histogram. Looks like the variation caused by plot and disk are not that significant for `tqM`.

```
sig2noise_plot <- sapply(unique(dat$plot_number), function(x) mean(dat[dat$plot_number ==
= x, ]$tqM) /
                                    sd(dat[dat$plot_number == x, ]$tqM))
hist(sig2noise_plot, breaks = 20, main = "Histogram of signal-to-noise ratios for plot e
ffect")
```




```
disk_num <- sort(unique(dat$disk))
sig2noise_disk <- sapply(disk_num, function(x) mean(dat[dat$disk == x, ]$tqM) /
                                 sd(dat[dat$disk == x, ]$tqM))
hist(sig2noise_disk, breaks = 20, main = "Histogram of signal-to-noise ratios for disk e
ffect")
```



According to the correlation plots, select and scale covariates before model fitting.

```
selected <- c('Ta', 'VPD', 'Fsd', 'Ta_7day', 'Precip', 'Precip_7day', 'VPD_7day', 'Fsd_7
day')
dat_sd <- dat %>% mutate_at(selected, ~(scale(.) %>% as.vector))
```

Model fitting and selection

Three models.

```
library(lme4)

m <- lm(tqM ~ ID + Ta + VPD + Fsd + Ta_7day + Precip +
                Precip_7day + VPD_7day + Fsd_7day,
        data = dat_sd)

m_re_plot <- lmer(tqM ~ ID + Ta + VPD + Fsd + Ta_7day + Precip +
                          Precip_7day + VPD_7day + Fsd_7day + (1|plot_number), data = da
t_sd, REML = FALSE)

m_re_full <- lmer(tqM ~ ID + Ta + VPD + Fsd + Ta_7day + Precip +
                          Precip_7day + VPD_7day + Fsd_7day + (1|plot_number) + (1|dis
k), data = dat_sd, REML = FALSE)
```

```
B <- 1e3

library(parallel)
library(doParallel)

myCluster <- makeCluster(detectCores(), type='PSOCK')
registerDoParallel(myCluster)

system.time(lrtstat <- foreach(1:B) %dopar% {
        y <- unlist(simulate(m))
        bnull <- lm(y ~ ID + Ta + VPD + Fsd + Ta_7day + Precip +
                             Precip_7day + VPD_7day + Fsd_7day,
                    data = dat_sd)
        balt <- lme4::lmer(y ~ ID + Ta + VPD + Fsd + Ta_7day + Precip +
                                    Precip_7day + VPD_7day + Fsd_7day + (1|plot_number),
                           data = dat_sd, REML = FALSE)
        as.numeric(2*(logLik(balt) - logLik(bnull)))
})
```

```
##    user  system elapsed
##   0.613   0.341  50.610
```

```
## p-value
pval <- mean(lrtstat > 2.568371)
pval
```

```
## [1] 0
```

```
## simple standard error of the above
sqrt(pval*(1-pval)/B)
```

```
## [1] 0
```

```
B <- 1e3

myCluster <- makeCluster(detectCores(), type='PSOCK')
registerDoParallel(myCluster)

system.time(lrtstat <- foreach(1:B) %dopar% {
        y <- unlist(simulate(m_re_plot))
        bnull <- lme4::lmer(y ~ ID + Ta + VPD + Fsd + Ta_7day + Precip +
                                    Precip_7day + VPD_7day + Fsd_7day + (1|plot_number),
                            data = dat_sd, REML = FALSE)
        balt <- lme4::lmer(y ~ ID + Ta + VPD + Fsd + Ta_7day + Precip +
                                   Precip_7day + VPD_7day + Fsd_7day + (1|plot_number) +
(1|disk),
                           data = dat_sd, REML = FALSE)

        as.numeric(2*(logLik(balt) - logLik(bnull)))
})
```

```
##    user  system elapsed
##   0.835   0.616 137.464
```

```
## p-value
pval <- mean(lrtstat > 2.568371)
pval
```

```
## [1] 0.051
```

```
## simple standard error of the above
sqrt(pval*(1-pval)/B)
```

```
## [1] 0.006956939
```

Bootstrapping choose the model with only random effect from plot(at significance level of 5%). Let's see if AIC and BIC agree with this.

```
AIC(m)
```

```
## [1] 29004.12
```

```
AIC(m_re_plot)
```

```
## [1] 28882.22
```

```
AIC(m_re_full)
```

```
## [1] 28883.1
```

```
BIC(m)
```

```
## [1] 29366.73
```

```
BIC(m_re_plot)
```

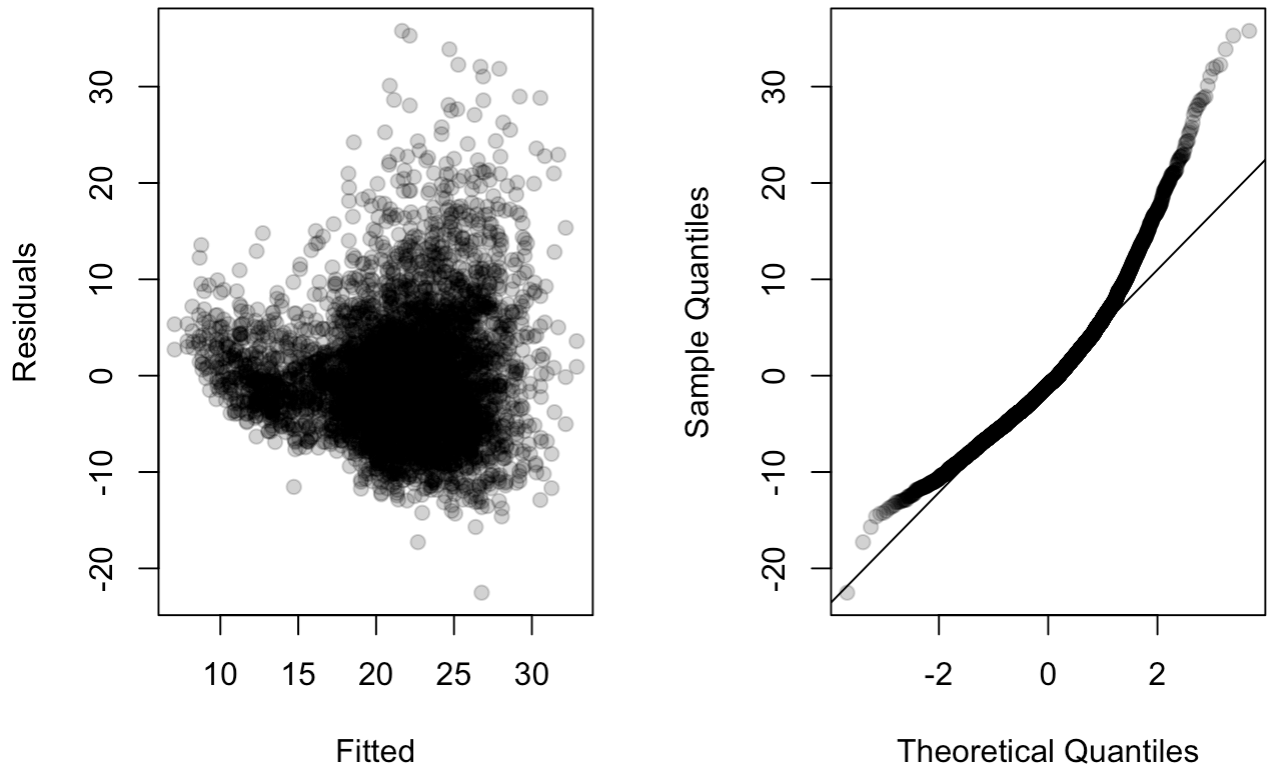
```
## [1] 29251.19
```

```
BIC(m_re_full)
```

```
## [1] 29258.42
```

They do. So let's just draw diagnostic plots for the model.

```
par(mfrow = c(1,2))
plot(fitted(m_re_plot), residuals(m_re_plot), xlab="Fitted", ylab="Residuals",
     pch = 19, col = rgb(0,0,0,alpha=0.2))
a <- qqnorm(residuals(m_re_plot), main="", pch = 19, col = rgb(0,0,0,alpha=0.2))
qqline(residuals(m_re_plot))
```



Looks like modeling assumptions are violated. Let try log transformation on the response.

```
dat_sd <- dat_sd %>% mutate(tqM_log = log(tqM))
```

```
m_log <- lm(tqM_log ~ ID + Ta + VPD + Fsd + Ta_7day + Precip +
                    Precip_7day + VPD_7day + Fsd_7day,
            data = dat_sd)

m_re_plot_log <- lmer(tqM_log ~ ID + Ta + VPD + Fsd + Ta_7day + Precip +
                              Precip_7day + VPD_7day + Fsd_7day + (1|plot_number), data
= dat_sd, REML = FALSE)

m_re_full_log <- lmer(tqM_log ~ ID + Ta + VPD + Fsd + Ta_7day + Precip +
                              Precip_7day + VPD_7day + Fsd_7day + (1|plot_number) + (1|d
isk), data = dat_sd, REML = FALSE)
```

```
AIC(m_log)
```

```
## [1] 2201.405
```

```
AIC(m_re_plot_log)
```

```
## [1] 2030.788
```

```
AIC(m_re_full_log)
```

```
## [1] 2029.283
```

```
BIC(m_log)
```

```
## [1] 2564.009
```

```
BIC(m_re_plot_log)
```

```
## [1] 2399.753
```

```
BIC(m_re_full_log)
```

```
## [1] 2404.61
```

This time AIC and BIC disagrees. Let's see what bootstrapping say.

```
myCluster <- makeCluster(detectCores(), type='PSOCK')
registerDoParallel(myCluster)


system.time(lrtstat <- foreach(1:B) %dopar% {
        y <- unlist(simulate(m_re_plot_log))
        bnull <- lme4::lmer(y ~ ID + Ta + VPD + Fsd + Ta_7day + Precip +
                                    Precip_7day + VPD_7day + Fsd_7day + (1|plot_number),
data = dat_sd, REML = FALSE)
        balt <- lme4::lmer(y ~ ID + Ta + VPD + Fsd + Ta_7day + Precip +
                                   Precip_7day + VPD_7day + Fsd_7day + (1|plot_number) +
(1|disk), data = dat_sd, REML = FALSE)

        as.numeric(2*(logLik(balt) - logLik(bnull)))
})
```

```
##    user  system elapsed
##   0.917   1.010 136.868
```

```
## p-value
pval <- mean(lrtstat > 2.568371)
pval
```

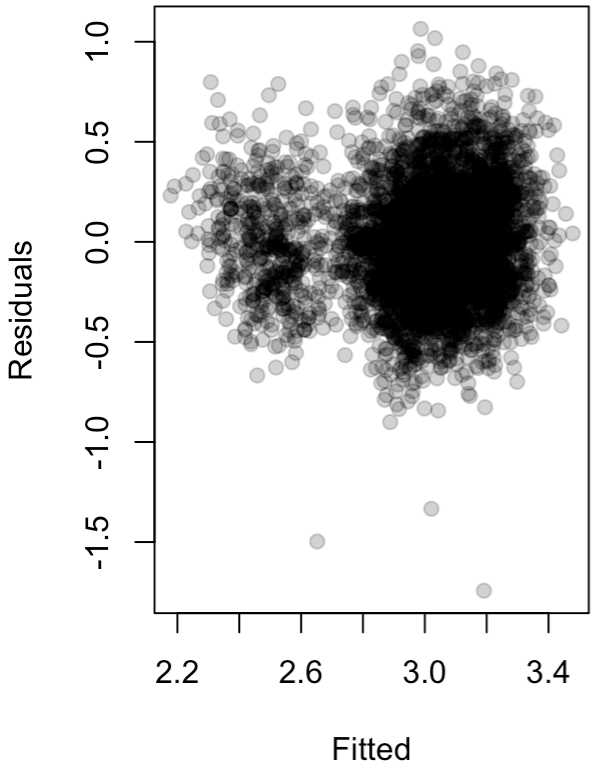
```
## [1] 0.05
```

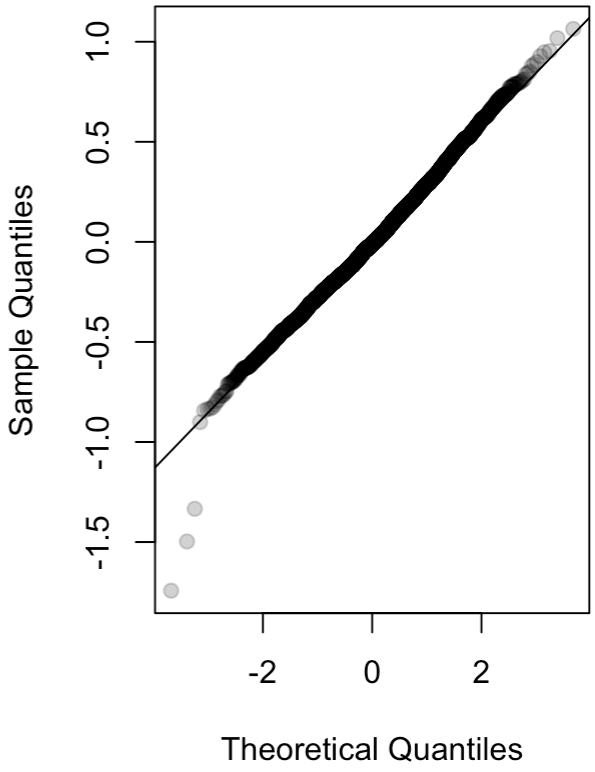
```
## simple standard error of the above
sqrt(pval*(1-pval)/B)
```

```
## [1] 0.006892024
```

Bootstrapping choose the full model, so we will go with it.

```
par(mfrow = c(1,2))
plot(fitted(m_re_full_log), residuals(m_re_full_log), xlab="Fitted", ylab="Residuals",
     pch = 19, col = rgb(0,0,0,alpha=0.2))
a <- qqnorm(residuals(m_re_full_log), main="", pch = 19, col = rgb(0,0,0,alpha=0.2))
qqline(residuals(m_re_full_log))
```





The log-transformation does make the model fit better. Modelling assumption are satisfied now.

Results

```
## AIC for each ID variable from full AqE fixed-effects model
M <- model.matrix(tqM_log ~ ID + Ta + VPD + Fsd + Ta_7day + Precip +
                          Precip_7day + VPD_7day + Fsd_7day,
                  data = dat_sd)
# Note that likelihood ratios are asymptotic, i.e. don't account for
# uncertainty in the estimate of the residual variance
system.time(AIC_IDs <- foreach(j = grep("ID", colnames(M))) %dopar% {
        M1 <- M[, -j]
        foo <- lme4::lmer(tqM_log ~ -1 + M1 + (1|plot_number) + (1|disk),
                          data = dat_sd, REML = FALSE)
        AIC(m_re_full_log) - AIC(foo)
})
```

```
##    user  system elapsed
##   0.390   0.094   5.799
```

```
AIC_IDs <- data.frame(unlist(AIC_IDs))
```

```
rownames(AIC_IDs) <- colnames(M)[grep("ID", colnames(M))]
colnames(AIC_IDs) <- c("tqM")
cbind(round(AIC_IDs,2), ifelse(AIC_IDs < 0, 1, 0))
```

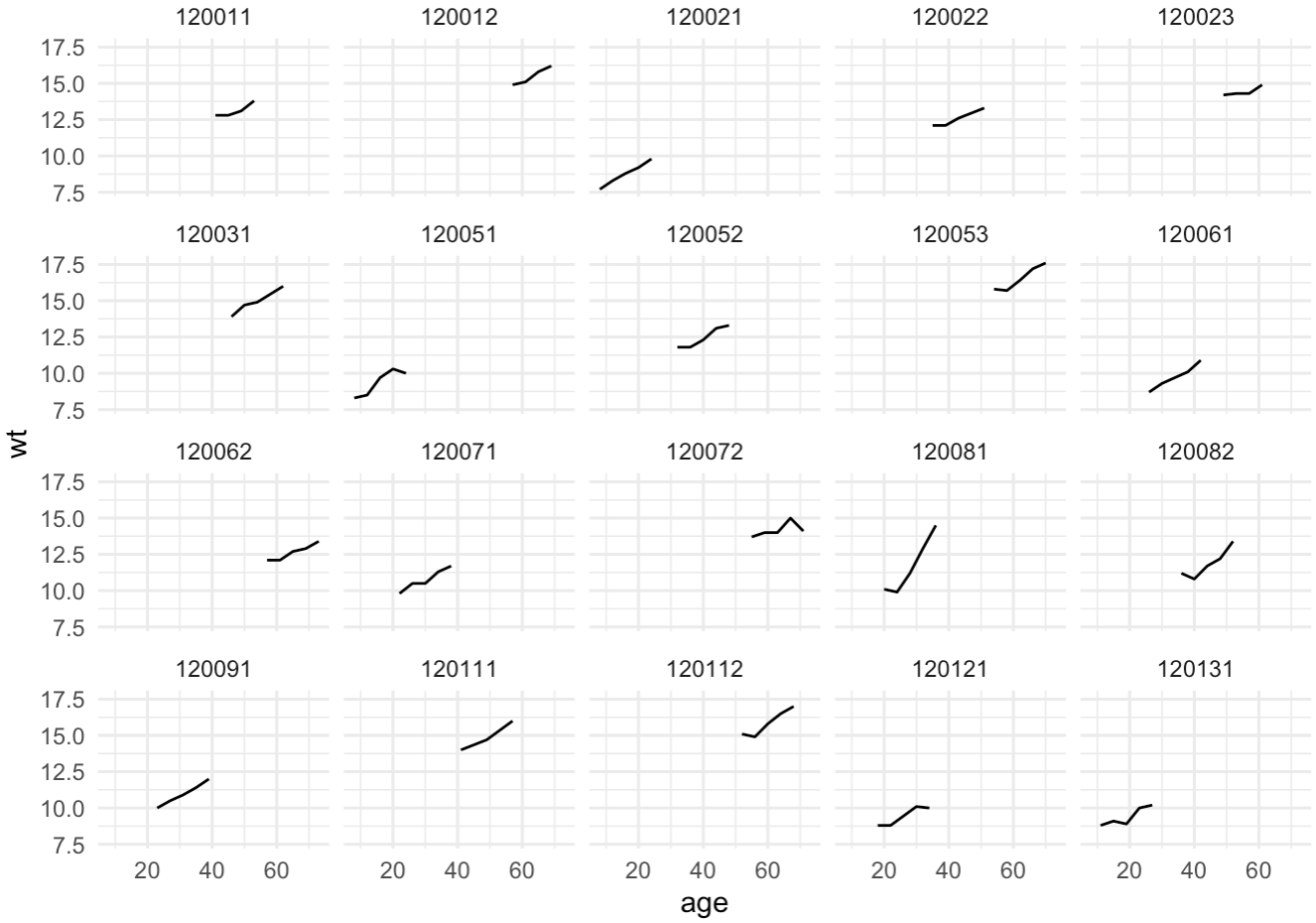
```
##        tqM tqM
## ID1   0.83   0
## ID2   1.64   0
## ID3   1.61   0
## ID4   1.78   0
## ID5   1.78   0
## ID6   0.32   0
## ID7  -0.11   1
## ID8   1.49   0
## ID9   1.99   0
## ID10  1.93   0
## ID11  1.09   0
## ID12  1.99   0
## ID13  1.88   0
## ID14 -2.60   1
## ID15 -1.04   1
## ID16  1.89   0
## ID17  0.31   0
## ID18  1.98   0
## ID19 -2.06   1
## ID20  1.95   0
## ID21  0.16   0
## ID22  1.76   0
## ID23 -0.73   1
## ID24  1.99   0
## ID25  1.99   0
## ID26  1.99   0
## ID27  0.95   0
## ID28  1.99   0
## ID29 -1.03   1
## ID30  1.98   0
## ID31  0.89   0
## ID32 -2.22   1
## ID33 -0.40   1
## ID34  1.51   0
## ID35  1.92   0
## ID36  1.92   0
## ID37  1.94   0
## ID38 -0.96   1
## ID39  0.82   0
## ID40 -2.33   1
## ID41 -4.17   1
## ID42 -1.37   1
## ID43 -2.62   1
## ID44 -2.73   1
## ID45  1.10   0
## ID46 -6.02   1
## ID47 -3.91   1
```

Problem 5 [20 points]:

The data in the package is a subset of a data from a public health study on Nepalese children. Develop a model for the weight of a child as he or she ages. You may use mage, lit, died, gender, and alive (but not ht) as covariates. Show how you developed your model and interpret your final model.

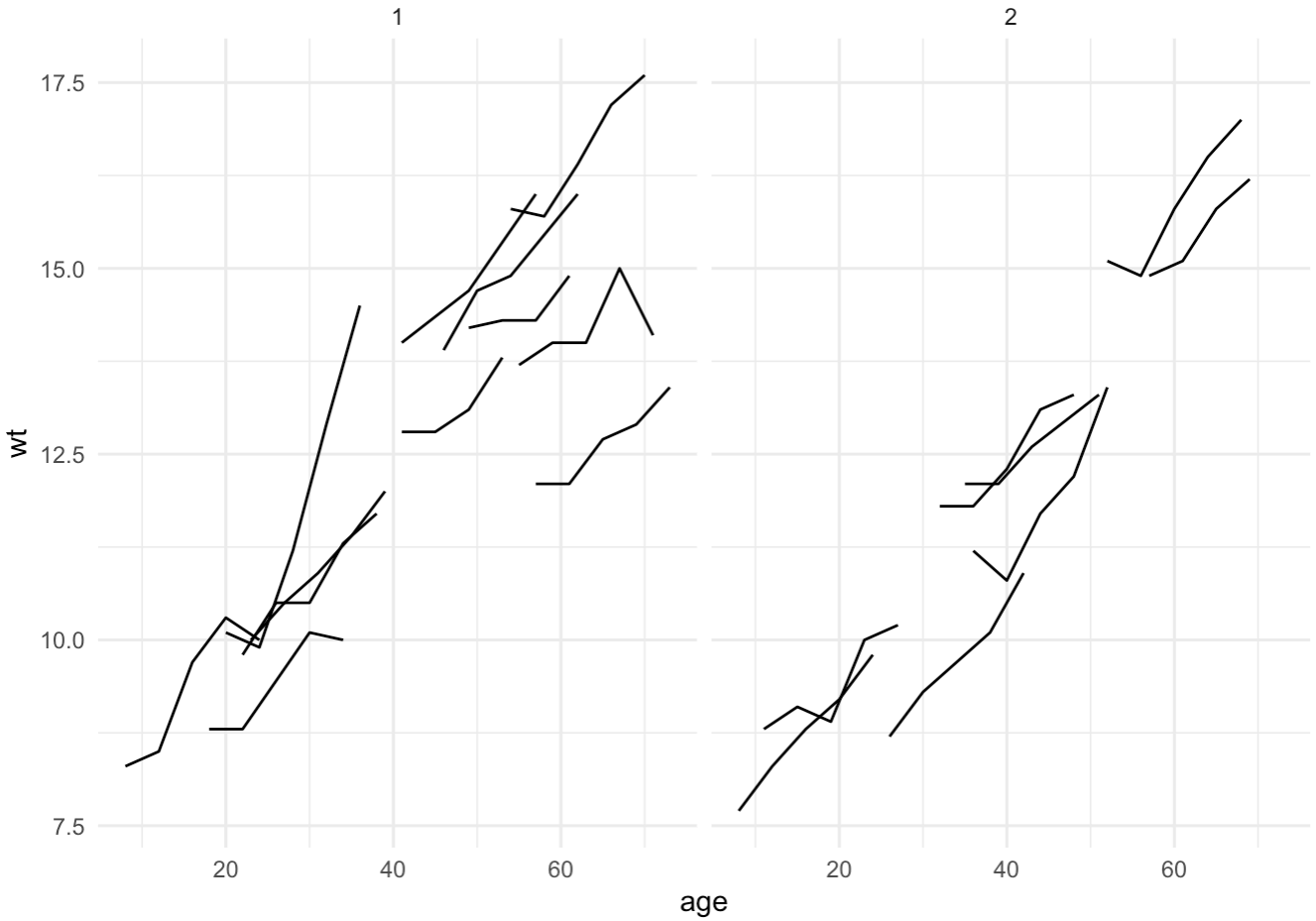
Solution 5:

```
library(faraway)
data(nepali)
nepali = na.omit(nepali)
nepali = data.frame(nepali)
dat20 = nepali[nepali$id <= unique(nepali$id)[20], ]
ggplot(dat20, aes(x = age, y = wt)) +
  geom_line() +
  facet_wrap(~id) +
  theme_minimal()
```



From the first 20 Nepalese children we can see an obvious positive effect of age on weight. But the rate is not the same for everyone. We will try to see how the weights vary by sex.

```
ggplot(dat20, aes(x=age, y=wt, group=id)) +
geom_line() +
facet_wrap(~ sex) +
theme_minimal()
```

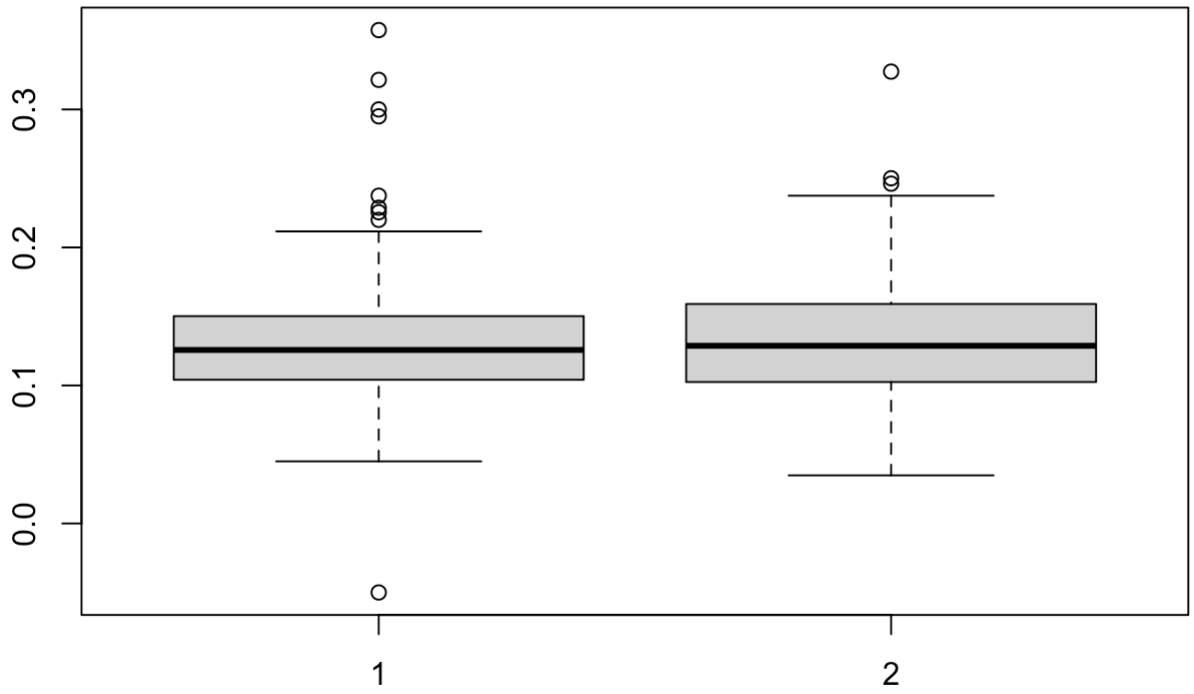


From this graph the weight growth rate of males (represented by 1) seem to be larger than for females.

We would check the differences in slope by a test of hypothesis now.

```
ml = lmList(wt ~ I(age) | id, nepali)
intercepts = sapply(ml,coef)[1,]
slopes = sapply(ml,coef)[2,]

slope_sex = nepali$sex[match(unique(nepali$id),nepali$id)]
boxplot(split(slopes,slope_sex))
```




```
t.test(slopes[slope_sex=="1"],slopes[slope_sex=="2"])
```

```
##
##  Welch Two Sample t-test
##
## data:  slopes[slope_sex == "1"] and slopes[slope_sex == "2"]
## t = 0.04012, df = 187.92, p-value = 0.968
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.01529955  0.01593479
## sample estimates:
## mean of x mean of y
## 0.1336587 0.1333411
```

The t-test also confimrs that sex should be included in our analysis.

We now fit our model.

Model 1: (i:id, j:age)

$$\text{wt}_{ij} = \mu + \beta_a * \text{age}_j + \beta_s * \text{sex}_i + \beta_i \text{age}_j * \text{sex}_i + \beta_m \text{mage}_i + \beta_l \text{lit}_i + \beta_d \text{died}_i + \beta_{al} \text{alive}_i + \gamma_i^0 + \gamma_i^1 \text{age}_j + \epsilon_{ij}$$

Model 2: (i:id, j:age)

$$\text{wt}_{ij} = \mu + \beta_a * \text{age}_j + \beta_s * \text{sex}_i + \beta_i \text{age}_j * \text{sex}_i + \gamma_i^0 + \gamma_i^1 \text{age}_j + \epsilon_{ij}$$

where

$$\begin{pmatrix} \gamma_i^0 \\ \gamma_i^1 \end{pmatrix} \sim N(0, \sigma^2 D)$$

```
mod1 = suppressWarnings({lmer(wt ~ age*sex + mage + lit + died + alive + (age|id), nepal
i)})
mod2 = suppressWarnings({lmer(wt ~ age*sex + (age|id), nepali)})
c(AIC(mod1),AIC(mod2))
```

```
## [1] 1740.223 1729.694
```

```
c(BIC(mod1),BIC(mod2))
```

```
## [1] 1797.541 1767.906
```

Model 2 is preferred by both AIC and BIC. We print the summary of model 2.

```
sumary(mod2, digits=3)
```

```
## Fixed Effects:
##             coef.est coef.se
## (Intercept)  6.722    0.411
## age          0.139    0.011
## sex         -0.394    0.262
## age:sex     -0.001    0.007
##
## Random Effects:
##  Groups   Name        Std.Dev. Corr
##  id       (Intercept) 1.416
##           age         0.037    -0.578
##  Residual             0.375
## ---
## number of obs: 877, groups: id, 197
## AIC = 1729.7, DIC = 1669.3
## deviance = 1691.5
```

Age has around 14% effect on the weight and females generally have lesser weight compared to males. Now for random effects of age for every individual, the standard deviation for the intercept and slope are 1.416 and 0.037 ($\sigma\sqrt{D_{11}}$ and $\sigma\sqrt{D_{22}}$), respectively. These have a negative correlation of -0.578 (cor(γ^0, γ^1)). Finally, there is some additional variation in the measurement not so far accounted for having standard deviation of 0.375 (sd(ε_{ij})). The variation of increase of weight among each individual is relatively small but the variation in overall weight among individuals is pretty high. Furthermore, given the large residual variation, there is a large age-to-age variation in weight.

Problem 6:

Prove that the penalized weighted residual sum-of-squares problem can be cast as

$$r^2(\theta, \beta, u) = r^2(\theta) + \|L_\theta^T(u - \mu_{U|Y=y_{obs}}) + R_{ZX}(\beta - \hat{\beta}_\theta)\|^2 + \|R_X(\beta - \hat{\beta}_\theta)\|^2$$

Solution 6:

The PLS problem is to minimize

$$r^2(\theta, \beta, u) = \left\| \begin{bmatrix} y_{obs} \\ 0 \end{bmatrix} - \begin{bmatrix} Z\Lambda_\theta & X \\ I_q & 0 \end{bmatrix} \begin{bmatrix} u \\ \beta \end{bmatrix} \right\|^2$$

This PLS problem may be thought of as a standard least squares problem for an extended response vector, which implies that the minimizing value $(\mu_{U|Y=yobs}^T \ \hat{\beta}_\theta^T)^T$ satisfies the normal equations,

$$\begin{bmatrix} \Lambda_\theta^T Z^T y_{obs} \\ X^T y_{obs} \end{bmatrix} = \begin{bmatrix} \Lambda_\theta^T Z^T Z \lambda_\theta + I & \Lambda_\theta^T Z^T X \\ X^T Z \Lambda_\theta & X^T X \end{bmatrix} \begin{bmatrix} \mu_{U|Y=yobs} \\ \hat{\beta}_\theta \end{bmatrix}$$

We can perform a Cholesky decomposition on the above cross-product matrix, so that

$$\begin{bmatrix} \Lambda_\theta^T Z^T Z \lambda_\theta + I & \Lambda_\theta^T Z^T X \\ X^T Z \Lambda_\theta & X^T X \end{bmatrix} = \begin{bmatrix} L_\theta & 0 \\ R_{ZX}^T & R_X^T \end{bmatrix} \begin{bmatrix} L_\theta^T & R_{ZX} \\ 0 & R_X \end{bmatrix}$$

To simplify this notation we define

$$Y = \begin{bmatrix} y_{obs} \\ 0 \end{bmatrix}, \quad A = \begin{bmatrix} Z\Lambda_\theta & X \\ I_q & 0 \end{bmatrix}, \quad \gamma = \begin{bmatrix} u \\ \beta \end{bmatrix}, \quad \hat{\gamma} = \begin{bmatrix} \mu_{U|Y=yobs} \\ \hat{\beta}_\theta \end{bmatrix} \quad \text{and } Q = \begin{bmatrix} L_\theta^T & R_{ZX} \\ 0 & R_X \end{bmatrix}$$

Using this notation we get

$$r^2(\theta, \beta, u) = \|Y - A\gamma\|^2$$

with the normal equations

$$A^T Y = A^T A \hat{\gamma}$$

and the cholesky decomposition of $A^T A$ is

$$A^T A = Q^T Q$$

Also note that we define

$$r^2(\theta) = \|Y - A\hat{\gamma}\|^2$$

Thus simplyfying $r^2(\theta, \beta, u)$ we get

$$
\begin{aligned}
r^2(\theta, \beta, u) &= \|Y - A\gamma\|^2 \\
&= (Y - A\gamma)^T (Y - A\gamma) \\
&= (Y - A\hat{\gamma} + A\hat{\gamma} - A\gamma)^T (Y - A\hat{\gamma} + A\hat{\gamma} - A\gamma) \\
&= (Y - A\hat{\gamma})^T (Y - A\hat{\gamma}) - (A\hat{\gamma} - A\gamma)^T (Y - A\hat{\gamma}) - (Y - A\hat{\gamma})^T (A\hat{\gamma} - A\gamma) + (A\hat{\gamma} - A\gamma)^T (A\hat{\gamma} - A\gamma) \\
&= \|Y - A\hat{\gamma}\|^2 + (\hat{\gamma} - \gamma)^T (A^T Y - A^T A\hat{\gamma}) - (A^T Y - A^T A\hat{\gamma})^T (\hat{\gamma} - \gamma) + (\hat{\gamma} - \gamma)^T A^T A (\hat{\gamma} - \gamma) \\
&= r^2(\theta) - 0 - 0 + (\hat{\gamma} - \gamma)^T Q^T Q (\hat{\gamma} - \gamma) \\
&= r^2(\theta) + \|Q(\hat{\gamma} - \gamma)\|^2 \\
&= r^2(\theta) + \left\| \begin{bmatrix} L_\theta^T & R_{ZX} \\ 0 & R_X \end{bmatrix} \left(\begin{bmatrix} u \\ \beta \end{bmatrix} - \begin{bmatrix} \mu_{U|Y=yobs} \\ \hat{\beta}_\theta \end{bmatrix} \right) \right\|^2 \\
&= r^2(\theta) + \|L_\theta^T (u - \mu_{U|Y=y_{obs}}) + R_{ZX}(\beta - \hat{\beta}_\theta)\|^2 + \|R_X(\beta - \hat{\beta}_\theta)\|^2
\end{aligned}
$$

Hence proved.