STOR 590 HW10 Solution

Taebin Kim

4/22/2020

Page 294 Exercise 1

```
library(faraway)
library(tidyverse)
```

```
## -- Attaching packages ----------------------------------------------------------------------------- t

## v ggplot2 3.2.1     v purrr   0.3.3
## v tibble  2.1.3     v dplyr   0.8.4
## v tidyr   1.0.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0

## -- Conflicts -------------------------------------------------------------------------------- tidyve
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(ggplot2)
```

Part (a)

We check if any of the mothers in the study change their smoking status during the period of observation.

```
data(ohio)



list0=-1
for(i in 0:536){
smokemax=max(ohio$smoke[ohio$id==i])
smokemin=min(ohio$smoke[ohio$id==i])
if(smokemax>smokemin)list0=c(list0,i)
}

list0
```

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

We can see that none of the mothers change their smoking status.

Part (b)

We construct a table that shows proportion of children who wheeze for 0, 1, 2, 3 or 4 years broken down by maternal smoking status.

```r
tab1=matrix(0,ncol=2,nrow=5)
for(i in 0:536){
i1=sum(ohio$resp[ohio$id==i])+1
j1=max(ohio$smoke[ohio$id==i])+1
tab1[i1,j1]=tab1[i1,j1]+1
}
prop.table(tab1, 1)
```

```
##           [,1]      [,2]
## [1,] 0.6676056 0.3323944
## [2,] 0.6701031 0.3298969
## [3,] 0.5681818 0.4318182
## [4,] 0.5217391 0.4782609
## [5,] 0.6111111 0.3888889
```

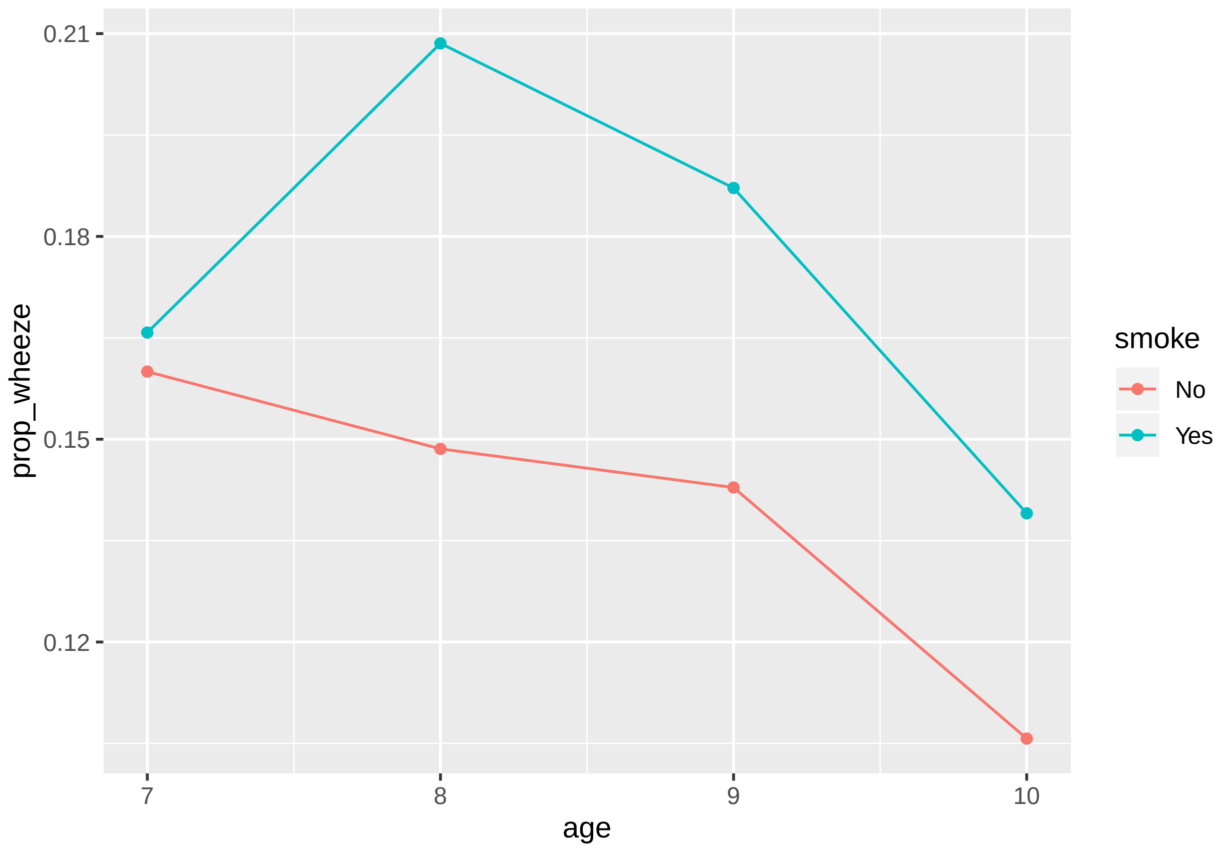
Part (c)

We make a plot which shows how the proportion of children wheezing changes by age with a separate line for smoking and nonsmoking mothers.

```r
df <- ohio %>% group_by(age, smoke) %>% summarise(prop_wheeze = mean(resp))
df$smoke <- ifelse(df$smoke == 1, "Yes", "No")
df$age <- df$age + 9
head(df)
```

```
## # A tibble: 6 x 3
## # Groups:   age [3]
##     age smoke prop_wheeze
##   <dbl> <chr>       <dbl>
## 1     7 No          0.16
## 2     7 Yes         0.166
## 3     8 No          0.149
## 4     8 Yes         0.209
## 5     9 No          0.143
## 6     9 Yes         0.187
```

```r
ggplot(df, aes(x = age, y = prop_wheeze, color = smoke)) + geom_point() + geom_line()
```



Part (d)

```
df2 <- ohio %>% group_by(id) %>% summarise(sum_wheeze = sum(resp), smoke = mean(smoke)) #min(smoke) or
head(df2)
```

```
## # A tibble: 6 x 3
##      id sum_wheeze smoke
##   <int>      <int> <dbl>
## 1     0          0     0
## 2     1          0     0
## 3     2          0     0
## 4     3          0     0
## 5     4          0     0
## 6     5          0     0
```

```
glm1 <- glm(cbind(sum_wheeze, 4 - sum_wheeze) ~ smoke, family = binomial, df2)
summary(glm1)
```

```
##
## Call:
## glm(formula = cbind(sum_wheeze, 4 - sum_wheeze) ~ smoke, family = binomial,
##     data = df2)
##
```

```
## Deviance Residuals: 
##     Min       1Q   Median       3Q      Max  
## -1.2411  -1.0954  -1.0954   0.5863   3.9711  
## 
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)    
## (Intercept) -1.82124    0.07719 -23.595   <2e-16 ***
## smoke        0.27156    0.12334   2.202   0.0277 *  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for binomial family taken to be 1)
## 
##     Null deviance: 1045.3  on 536  degrees of freedom
## Residual deviance: 1040.5  on 535  degrees of freedom
## AIC: 1337.9
## 
## Number of Fisher Scoring iterations: 4
```

The p-value indicates that the smoke effect is significant. However, we have to check if it is still significant after considering random effects.

Part (e)

We fit a model for each individual response using a GLMM fit using penalized quasi-likelihood.

```r
library(MASS)
```

```
## 
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
## 
##     select
```

```r
pql1 <- glmmPQL(resp ~ age + smoke, random = ~1|id, family = binomial, ohio)
```

```
## iteration 1

## iteration 2

## iteration 3

## iteration 4

## iteration 5

## iteration 6

## iteration 7

## iteration 8
```

```r
summary(pql1)
```

```
## Linear mixed-effects model fit by maximum likelihood
##  Data: ohio
##   AIC BIC logLik
##    NA  NA     NA
##
## Random effects:
##  Formula: ~1 | id
##         (Intercept)  Residual
## StdDev:    2.057175 0.6355563
##
## Variance function:
##  Structure: fixed weights
##  Formula: ~invwt
## Fixed effects: resp ~ age + smoke
##                   Value  Std.Error   DF    t-value p-value
## (Intercept) -2.7658365 0.14218299 1610 -19.452654  0.0000
## age         -0.1815756 0.04365164 1610  -4.159652  0.0000
## smoke        0.3251839 0.23131699  535   1.405793  0.1604
##  Correlation:
##       (Intr) age
## age    0.197
## smoke -0.591 -0.003
##
## Standardized Within-Group Residuals:
##        Min         Q1        Med         Q3        Max
## -2.6145143 -0.2829352 -0.2583719 -0.2154580  3.4443795
##
## Number of Observations: 2148
## Number of Groups: 537
```

The odds of wheezing decreases over time, and the odds of wheezing increases if the mother is a smoker.

Part (f)

We fit the same model but using adaptive Gaussian-Hermit quadrature.

```r
library(lme4)
```

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

```
##
## Attaching package: 'Matrix'
```

```
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
```

```
glmer1 <- glmer(resp ~ age + smoke + (1|id), family = binomial, ohio)
summary(glmer1)
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
##   Approximation) [glmerMod]
##  Family: binomial  ( logit )
## Formula: resp ~ age + smoke + (1 | id)
##    Data: ohio
##
##      AIC      BIC   logLik deviance df.resid
##   1597.9   1620.6   -794.9   1589.9     2144
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.4027 -0.1802 -0.1577 -0.1321  2.5176
##
## Random effects:
##  Groups Name        Variance Std.Dev.
##  id     (Intercept) 5.49     2.343
## Number of obs: 2148, groups:  id, 537
##
## Fixed effects:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -3.37395    0.27498 -12.270   <2e-16 ***
## age         -0.17676    0.06797  -2.601   0.0093 **
## smoke        0.41478    0.28704   1.445   0.1485
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##       (Intr) age
## age    0.227
## smoke -0.419 -0.010
```

The values are similar to those from the previous model. For both models, `age` is the only significant predictor.

Part (g)

We use INLA to fit the same model.

```
library(INLA)
```

```
## Loading required package: sp
```

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

```
## Loading required package: foreach
```

```
##
## Attaching package: 'foreach'
## The following objects are masked from 'package:purrr':
##
##     accumulate, when

## This is INLA_20.03.17 built 2020-03-17 07:56:40 UTC.
## See www.r-inla.org/contact-us for how to get help.
```

```
formula <- resp ~ age + smoke + f(id, model = 'iid')
result <- inla(formula, family = 'binomial', data = ohio)
summary(result)
```

```
##
## Call:
##    "inla(formula = formula, family = \"binomial\", data = ohio)"
## Time used:
##     Pre = 0.352, Running = 1.33, Post = 0.107, Total = 1.79
## Fixed effects:
##               mean    sd 0.025quant 0.5quant 0.975quant   mode kld
## (Intercept) -2.991 0.202     -3.410   -2.983     -2.617 -2.967   0
## age         -0.167 0.063     -0.290   -0.166     -0.044 -0.166   0
## smoke        0.392 0.239     -0.078    0.391      0.863  0.390   0
##
## Random effects:
##   Name     Model
##     id IID model
##
## Model hyperparameters:
##                   mean    sd 0.025quant 0.5quant 0.975quant  mode
## Precision for id 0.276 0.047      0.196    0.272      0.378 0.265
##
## Expected number of effective parameters(stdev): 281.76(12.71)
## Number of equivalent replicates : 7.62
##
## Marginal log-Likelihood:  -834.77
```

The result show that the odds of wheezing decreases over time, and the odds of wheezing increases if the mother is a smoker. smoke is not significant since 0 is included in the 95% CI.

Part (i)

We fit the model using GEE.

```
library(geepack)
```

```
##
## Attaching package: 'geepack'

## The following object is masked _by_ '.GlobalEnv':
##
##     ohio
## The following object is masked from 'package:faraway':
##
##     ohio
```

```
gee1 <- geeglm(resp ~ age + smoke, id = id, corstr = 'ar1', ohio, family = binomial)
summary(gee1)
```

```
##
## Call:
## geeglm(formula = resp ~ age + smoke, family = binomial, data = ohio,
##     id = id, corstr = "ar1")
##
##  Coefficients:
##             Estimate  Std.err    Wald Pr(>|W|)
## (Intercept) -1.90218  0.11525 272.409   <2e-16 ***
## age         -0.11489  0.04539   6.407   0.0114 *
## smoke        0.23448  0.18119   1.675   0.1956
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation structure = ar1
## Estimated Scale Parameters:
##
##             Estimate Std.err
## (Intercept)    1.021  0.1232
##   Link = identity
##
## Estimated Correlation Parameters:
##       Estimate Std.err
## alpha    0.491 0.06733
## Number of clusters:   537  Maximum cluster size: 4
```

Since the estimate of the correlation is significantly larger than the standard error, we conclude that a wheezing child is likely to wheeze again next year.

Part (j)

The overall conclusion is that `age` is a significant predictor while `smoke` is not. The odds of wheezing decreases over time. The GLMM model is preferrable to consider random effects.

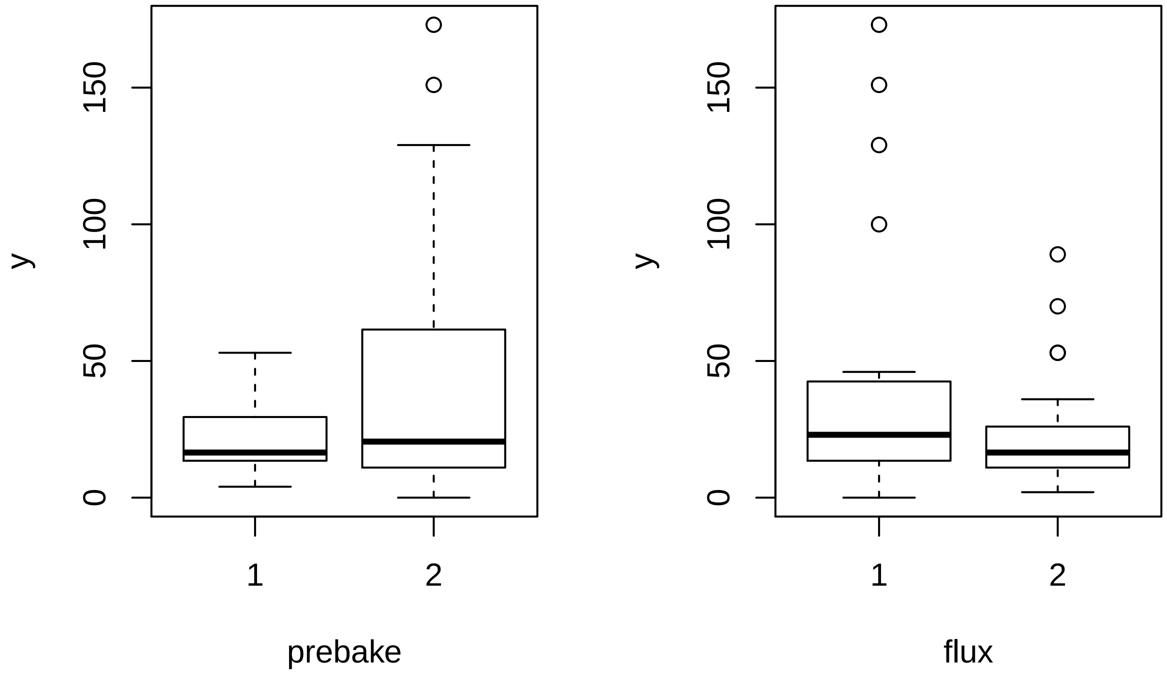
Page 295 Exercise 3

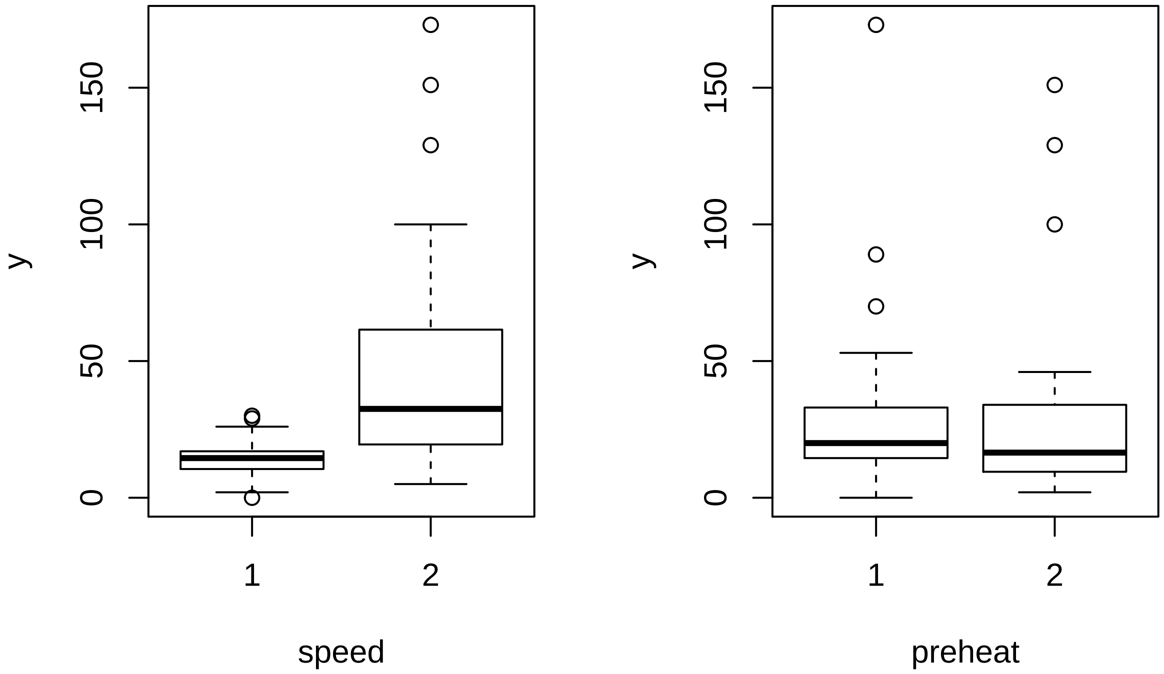
Part (a)

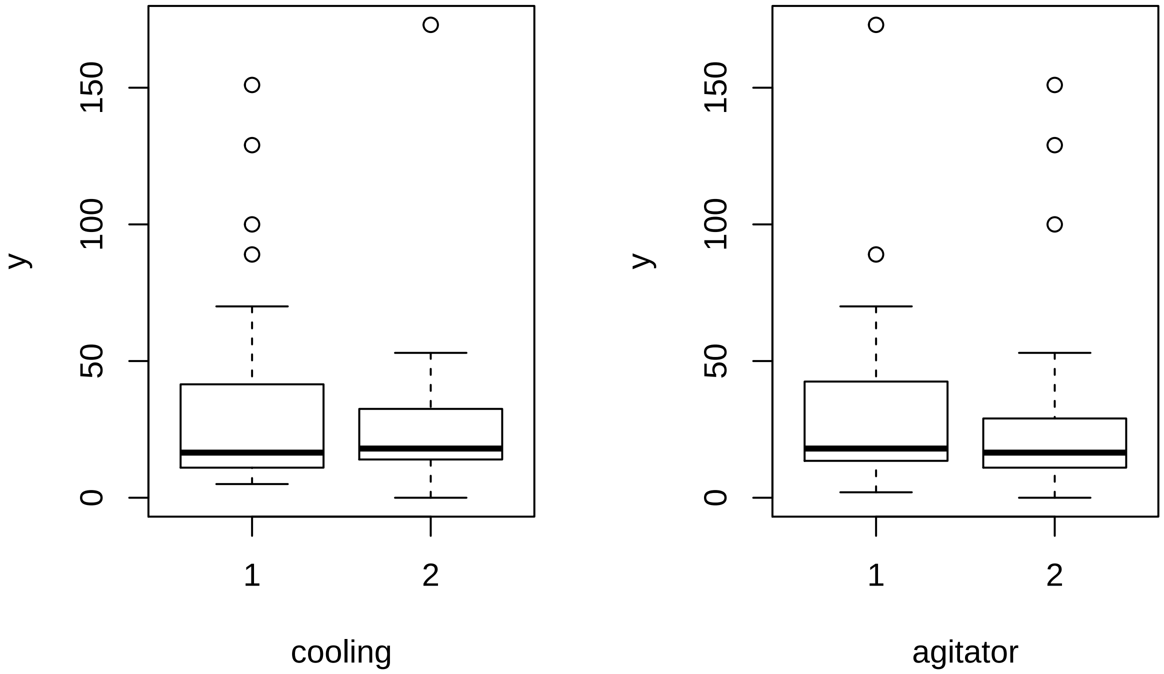
We plot the data to show how the number of defects varies with the predictors after separating y_is.

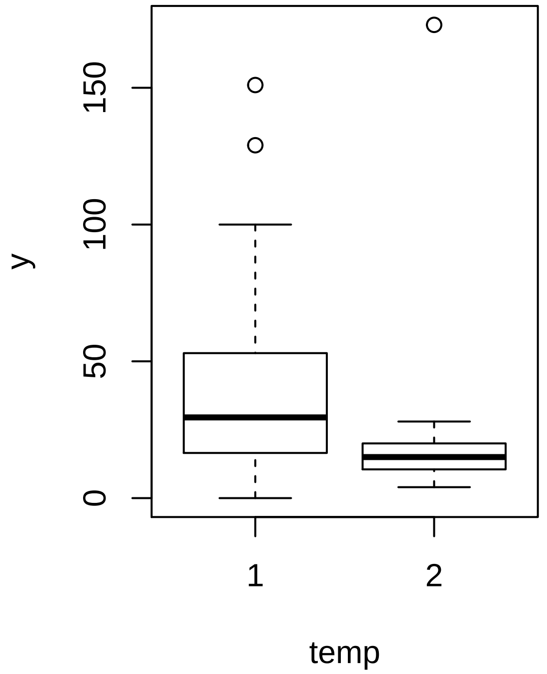
```
data(wavesolder)

df <- wavesolder %>%
  mutate(id = row_number()) %>%
```

```r
  pivot_longer(cols = c("y1", "y2", "y3"),
               values_to = "y")
par(mfrow = c(1,2))
plot(y ~ prebake + flux + speed + preheat + cooling + agitator + temp, df)
```




y
0
50
100
150
1
2
speed
y
0
50
100
150
1
2
preheat


y
0
50
100
150
1
2
cooling
y
0
50
100
150
1
2
agitator



Part (b)

We fit a Poisson GLM to the individual runs with the number of defects as the response and main effects for all the predictors. Then we fit a comparable quasi-Poisson GLM.

```
glm1 <- glm(y ~  prebake + flux + speed + preheat + cooling + agitator + temp, family = poisson, df)
summary(glm1)
```

```
##
## Call:
## glm(formula = y ~ prebake + flux + speed + preheat + cooling +
##     agitator + temp, family = poisson, data = df)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -7.907  -2.120  -0.412   1.550  12.149
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   3.0428     0.0856   35.55   <2e-16 ***
## prebake2      0.6512     0.0544   11.97   <2e-16 ***
## flux2        -0.5767     0.0544  -10.60   <2e-16 ***
## speed2        1.2287     0.0611   20.10   <2e-16 ***
## preheat2     -0.1665     0.0534   -3.12   0.0018 **
## cooling2     -0.1619     0.0531   -3.05   0.0023 **
## agitator2    -0.0897     0.0526   -1.71   0.0882 .
## temp2        -0.6982     0.0554  -12.61   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 1450.52  on 47  degrees of freedom
## Residual deviance:  491.78  on 40  degrees of freedom
## AIC: 738.5
##
## Number of Fisher Scoring iterations: 5
```

```
glm2 <- glm(y ~  prebake + flux + speed + preheat + cooling + agitator + temp, family = quasipoisson, d
summary(glm2)
```

```
##
## Call:
## glm(formula = y ~ prebake + flux + speed + preheat + cooling +
##     agitator + temp, family = quasipoisson, data = df)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -7.907  -2.120  -0.412   1.550  12.149
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.0428     0.3135    9.70  4.5e-12 ***
## prebake2      0.6512     0.1993    3.27   0.0022 **
## flux2        -0.5767     0.1992   -2.89   0.0061 **
## speed2        1.2287     0.2239    5.49  2.5e-06 ***
## preheat2     -0.1665     0.1957   -0.85   0.4001
## cooling2     -0.1619     0.1946   -0.83   0.4105
## agitator2    -0.0897     0.1928   -0.47   0.6441
## temp2        -0.6982     0.2028   -3.44   0.0014 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for quasipoisson family taken to be 13.42)
##
##     Null deviance: 1450.52  on 47  degrees of freedom
## Residual deviance:  491.78  on 40  degrees of freedom
## AIC: NA
##
## Number of Fisher Scoring iterations: 5
```

The Possion GLM model is inadequate since the residual deviance of the model is significantly larger than the degrees of freedom. The quasi-Poission GLM model does not change any coefficient estimates but increases the p-values of the predictors.

Part (c)

We sum the defects within each replicate group of three and fit a quasi-Poisson GLM to these sums.

```
df2 <- wavesolder %>%
  mutate(y = y1 + y2 + y3)
glm3 <- glm(y ~  prebake + flux + speed + preheat + cooling + agitator + temp, family = quasipoisson, d
summary(glm3)
```

```
##
## Call:
## glm(formula = y ~ prebake + flux + speed + preheat + cooling +
##     agitator + temp, family = quasipoisson, data = df2)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -5.875  -2.197  -0.037   2.486   6.427
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   4.1414     0.4081   10.15  7.6e-06 ***
## prebake2      0.6512     0.2594    2.51   0.0364 *
## flux2        -0.5767     0.2593   -2.22   0.0568 .
## speed2        1.2287     0.2915    4.22   0.0029 **
## preheat2     -0.1665     0.2548   -0.65   0.5318
## cooling2     -0.1619     0.2533   -0.64   0.5407
## agitator2    -0.0897     0.2510   -0.36   0.7299
## temp2        -0.6982     0.2640   -2.64   0.0295 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for quasipoisson family taken to be 22.73)
##
##     Null deviance: 1126.79  on 15  degrees of freedom
## Residual deviance:  168.05  on  8  degrees of freedom
## AIC: NA
##
## Number of Fisher Scoring iterations: 4
```

The new model does not change any coefficient estimates but increases the dispersion parameter and the p-values of the predictors.

Part (d)

We fit a GEE model to the individual defect responses with a fixed scale that allows for an autoregressive correlation structure within the groups.

```
gee1 <- geeglm(y ~ prebake + flux + speed + preheat + cooling + agitator + temp, id = id, corstr = "ar1
summary(gee1)
```

```
##
## Call:
```

```
## geeglm(formula = y ~ prebake + flux + speed + preheat + cooling +
##     agitator + temp, family = "poisson", data = df, id = id,
##     corstr = "ar1", scale.fix = T)
##
##  Coefficients:
##             Estimate Std.err   Wald Pr(>|W|)
## (Intercept)    3.041   0.222 187.39  < 2e-16 ***
## prebake2       0.669   0.179  13.96  0.00019 ***
## flux2         -0.600   0.164  13.33  0.00026 ***
## speed2         1.240   0.201  37.90  7.4e-10 ***
## preheat2      -0.174   0.167   1.09  0.29730
## cooling2      -0.155   0.169   0.85  0.35630
## agitator2     -0.111   0.160   0.48  0.48816
## temp2         -0.665   0.178  13.98  0.00019 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation structure = ar1
## Scale is fixed.
##
##   Link = identity
##
## Estimated Correlation Parameters:
##       Estimate Std.err
## alpha   -0.163   0.386
## Number of clusters:   16  Maximum cluster size: 3
```

Since there was an overdispersion in the Poisson GLM model, it is unreasonable to fix the scale.

Part (e)

We refit without a fixed scale.

```
gee2 <- geeglm(y ~ prebake + flux + speed + preheat + cooling + agitator + temp, id = id, corstr = "ar1
summary(gee2)
```

```
##
## Call:
## geeglm(formula = y ~ prebake + flux + speed + preheat + cooling +
##     agitator + temp, family = "poisson", data = df, id = id,
##     corstr = "ar1", scale.fix = F)
##
##  Coefficients:
##             Estimate Std.err   Wald Pr(>|W|)
## (Intercept)    3.041   0.222 186.81  < 2e-16 ***
## prebake2       0.669   0.179  13.95  0.00019 ***
## flux2         -0.601   0.165  13.34  0.00026 ***
## speed2         1.240   0.202  37.80  7.8e-10 ***
## preheat2      -0.175   0.168   1.09  0.29711
## cooling2      -0.155   0.169   0.85  0.35779
## agitator2     -0.112   0.161   0.49  0.48546
## temp2         -0.663   0.178  13.89  0.00019 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation structure = ar1
## Estimated Scale Parameters:
##
##             Estimate Std.err
## (Intercept)     10.8    4.62
##   Link = identity
##
## Estimated Correlation Parameters:
##       Estimate Std.err
## alpha    -0.17   0.337
## Number of clusters:   16  Maximum cluster size: 3
```

Since the standard error of the correlation is larger than the absolute value of the estimate, we conclude that it is not significant.

Part (f)

We fit a GEE model with an independent correlation structure within the replicates.

```
gee3 <- geeglm(y ~ prebake + flux + speed + preheat + cooling + agitator + temp, id = id, corstr = "inde
summary(gee3)
```

```
##
## Call:
## geeglm(formula = y ~ prebake + flux + speed + preheat + cooling +
##     agitator + temp, family = "poisson", data = df, id = id,
##     corstr = "independence")
##
##  Coefficients:
##             Estimate Std.err   Wald Pr(>|W|)
## (Intercept)   3.0428  0.2141 201.98  < 2e-16 ***
## prebake2      0.6512  0.1718  14.37  0.00015 ***
## flux2        -0.5767  0.1591  13.14  0.00029 ***
## speed2        1.2287  0.1933  40.39  2.1e-10 ***
## preheat2     -0.1665  0.1614   1.06  0.30243
## cooling2     -0.1619  0.1624   0.99  0.31898
## agitator2    -0.0897  0.1534   0.34  0.55857
## temp2        -0.6982  0.1735  16.19  5.7e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation structure = independence
## Estimated Scale Parameters:
##
##             Estimate Std.err
## (Intercept)     11.2    5.04
## Number of clusters:   16  Maximum cluster size: 3
```

Since this model produces similar result to the quasi-Poisson model, we conclude that the correlation among y_is for the same wafer is not significant.