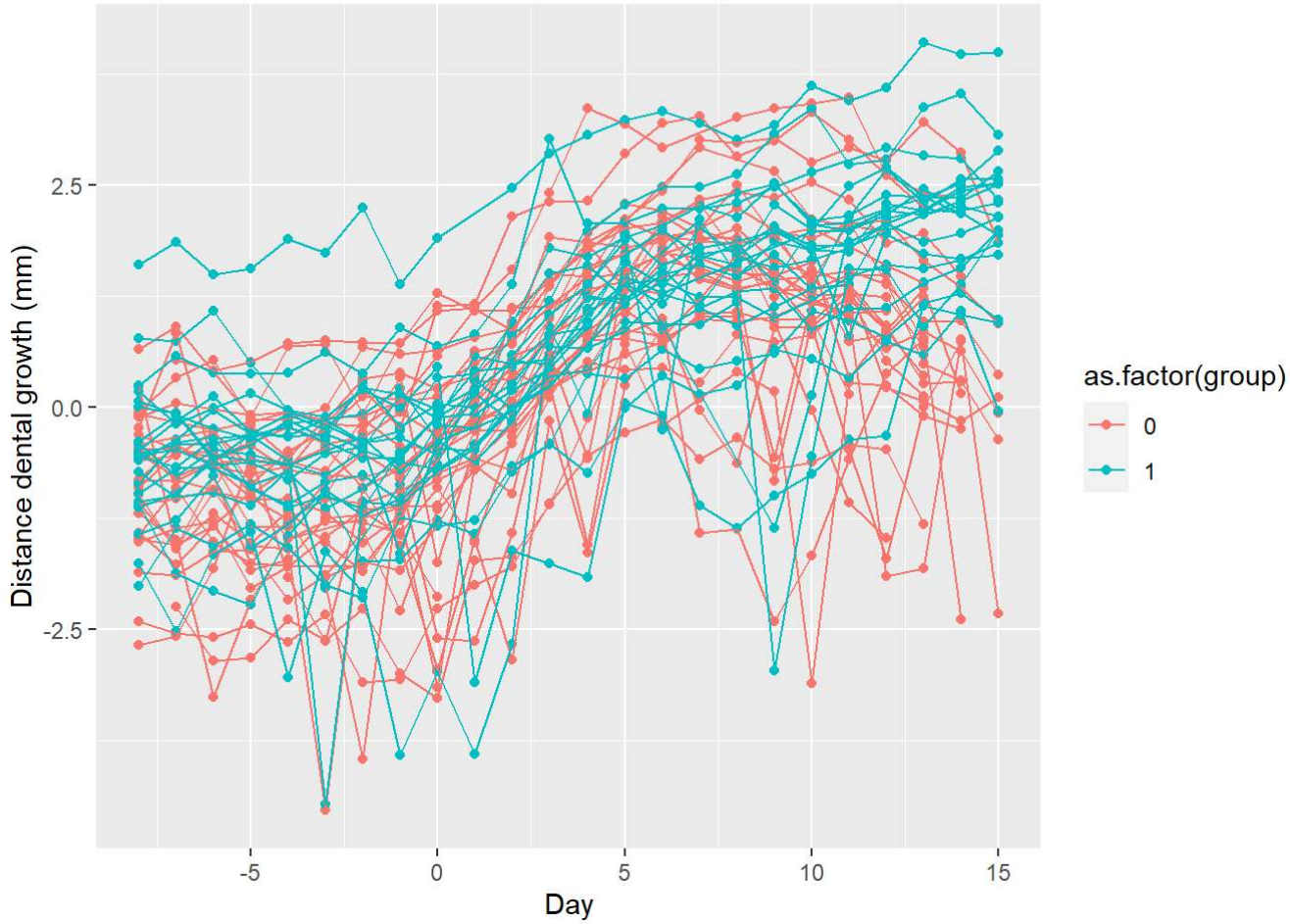
Stats 112 Homework 4

Owen Lin

Problem 1

1a:

```
# setwd("")
prog = read.csv("progesterone.csv", header = TRUE)

prog %>%
  group_by(group) %>%
  ggplot(aes(time, PDG, group = id, color = as.factor(group))) +
  geom_point() +
  geom_line() +
  labs(x = "Day",
       y = "Distance dental growth (mm)")
```



1b:

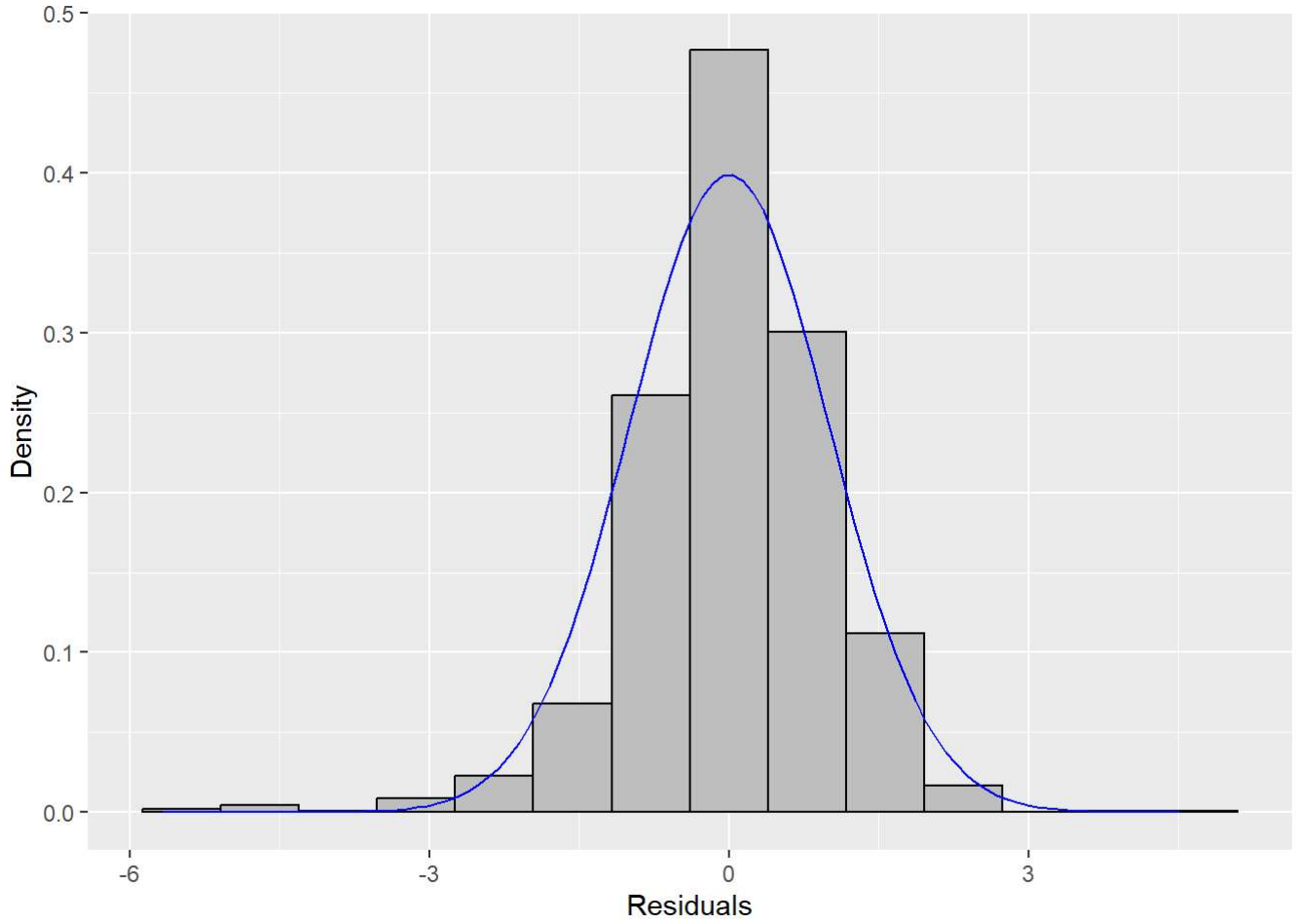
```
prog$group = as.factor(prog$group)
prog = prog %>%
  mutate(timeSqr = time^2, timeCub = time^3)

model1 = lme(PDG ~ time + group : time + timeSqr + group: timeSqr ,
                     data = prog,
                     random = ~ 1 + time + timeSqr| id,
                     method = "REML")

summary(model1)
```

```
## Linear mixed-effects model fit by REML
##   Data: prog
##        AIC      BIC    logLik
##   2623.384 2683.691 -1299.692
##
## Random effects:
##  Formula: ~1 + time + timeSqr | id
##  Structure: General positive-definite, Log-Cholesky parametrization
##             StdDev      Corr
## (Intercept) 0.875009554 (Intr) time
## time        0.048000579  0.451
## timeSqr     0.004289056 -0.526 -0.791
## Residual    0.665482803
##
## Fixed effects:  PDG ~ time + group:time + timeSqr + group:timeSqr
##                       Value  Std.Error   DF   t-value p-value
## (Intercept)     0.02653357 0.12571401 1075  0.211063  0.8329
## time            0.16131529 0.01037028 1075 15.555532  0.0000
## timeSqr        -0.00552998 0.00102274 1075 -5.407044  0.0000
## time:group1    -0.02955356 0.01482118 1075 -1.994009  0.0464
## group1:timeSqr  0.00765360 0.00137907 1075  5.549827  0.0000
##  Correlation:
##                (Intr) time   timSqr tm:gr1
## time            0.307
## timeSqr        -0.376 -0.720
## time:group1    -0.002 -0.634  0.423
## group1:timeSqr  0.012  0.452 -0.641 -0.703
##
## Standardized Within-Group Residuals:
##         Min          Q1         Med          Q3         Max
## -5.26306718 -0.52146641  0.06171999  0.62227311  3.64039251
##
## Number of Observations: 1130
## Number of Groups: 51
```

1c:

We need to transform the residuals because not only do they have different variance, but they are also correlated with each other.

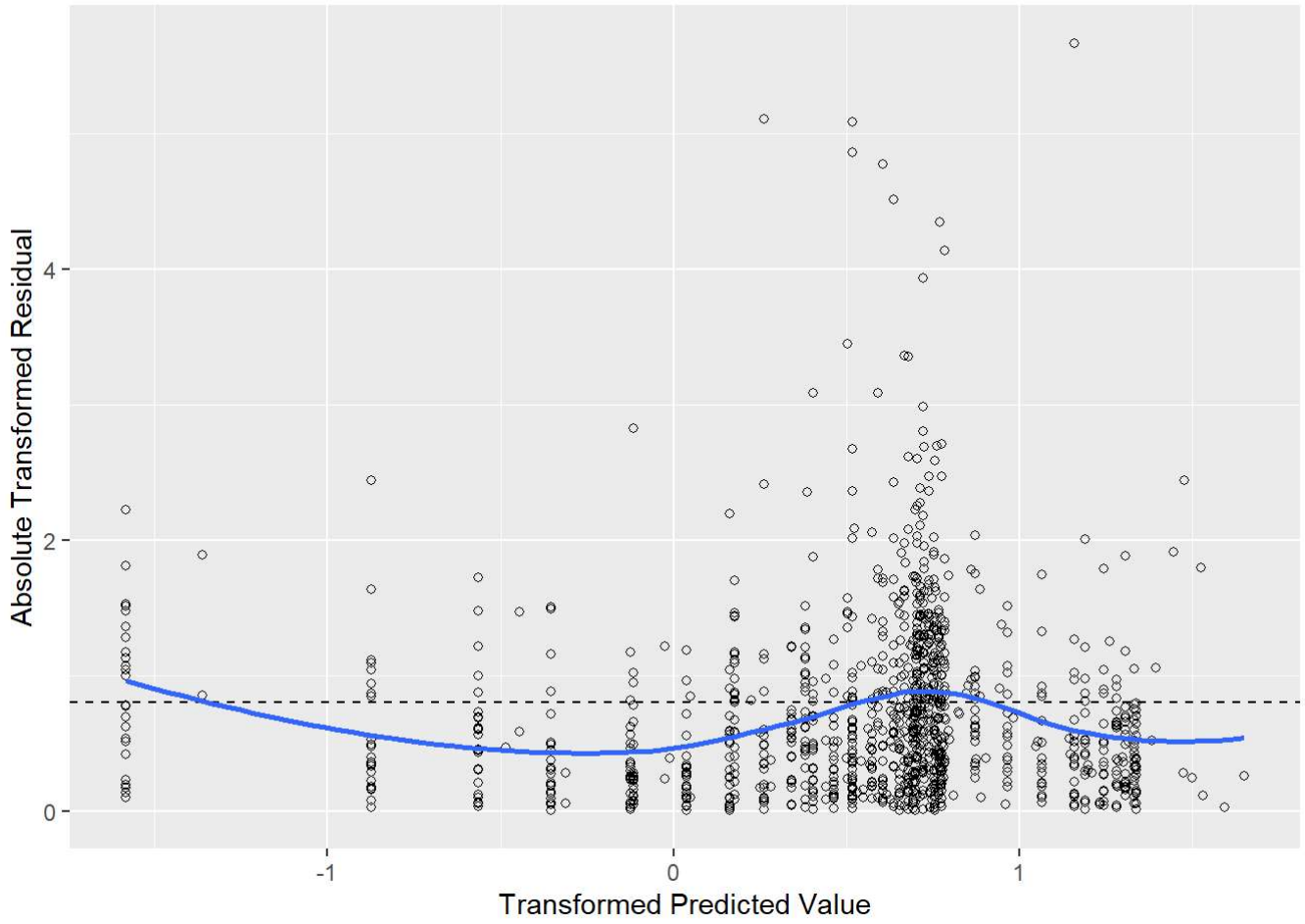
```
res_population = residuals(model1, type = "response", level = 0)

Sigma_i = extract.lme.cov(model1, prog)
L_i = t(chol(Sigma_i)) #block matrix of lower triangular Cholesky factors
res_transformed <- solve(L_i) %*% res_population
tibble(r_star = res_transformed) %>%
  ggplot(aes(x = r_star)) +
  geom_histogram(aes(y = stat(density)), bins = 14, color = "black", fill = "gray") +
  geom_function(fun = dnorm, color = "blue") +
  labs(x = "Residuals", y = "Density")
```



1d:

The smooth line is around 1, but at the predicted range of 0.7-0.8, there is a lot of outlier in transformed residuals.

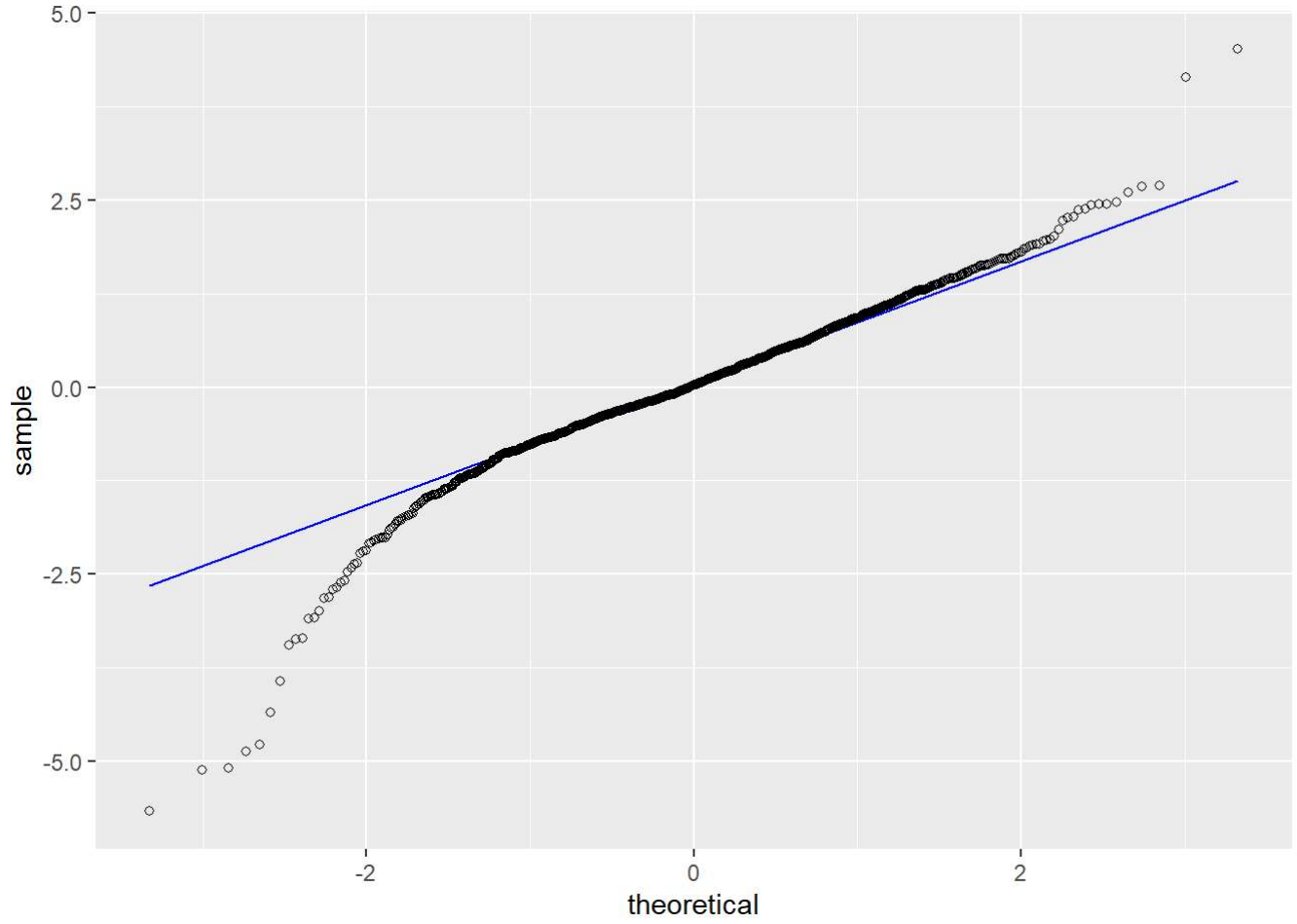
```
mu_hat = fitted(model1, level = 0)
mu_hat_transformed = solve(L_i) %*% mu_hat
abs_res_transformed = abs(res_transformed)

tibble(x = mu_hat_transformed, y = abs_res_transformed) %>%
  ggplot(aes(x = x, y = y)) +
  geom_hline(yintercept = 0.8, linetype = "dashed") +
  geom_point(shape = 1) +
  geom_smooth(method = "loess", se = FALSE) +
  labs(x = "Transformed Predicted Value", y = "Absolute Transformed Residual")
```



1e:

The qq plot didn't fit well for the end points: there are a lot of outliers.

```
tibble(r_star = res_transformed) %>%
  ggplot(aes(sample = r_star)) +
  geom_qq_line(color = "blue") +
  geom_qq(shape = 1)
```




```
  labs(x = "Quantiles of Standard Normal", y = "Quantiles of Transformed Residuals")
```

```
## $x
## [1] "Quantiles of Standard Normal"
##
## $y
## [1] "Quantiles of Transformed Residuals"
##
## attr(,"class")
## [1] "labels"
```

1f:

There are 9 potential outlying individuals with a p-value less than 0.05.

```
mahalanobis_distance = function(x){
  x <- as.matrix(x)
  t(x) %*% x
}

mahalanobis_data <- tibble(id = prog$id, r_star = res_transformed) %>%
  group_by(id) %>%
  nest() %>%
  mutate(df = map_dbl(data, ~nrow(.x)))%>%
  mutate(d = map_dbl(data, ~mahalanobis_distance(.x)))%>%
  mutate(p_value = pchisq(d, df, lower.tail = FALSE))


mahalanobis_data %>%
  arrange(p_value)
```

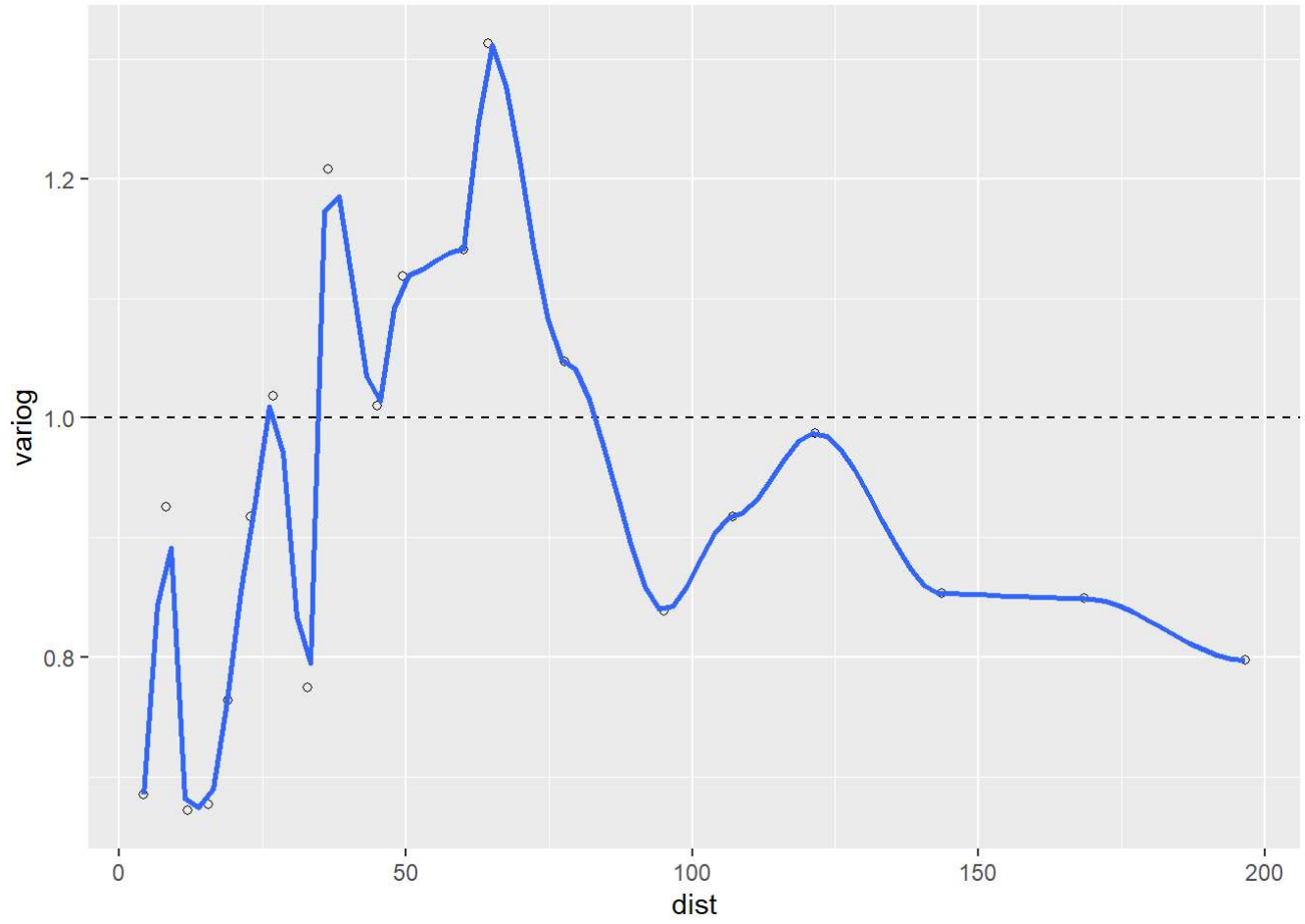
```
## # A tibble: 51 × 5
## # Groups:   id [51]
##       id data                 df     d  p_value
##    <int> <list>            <dbl> <dbl>    <dbl>
##  1    10 <tibble [23 × 1]>    23  98.5 2.54e-11
##  2    42 <tibble [21 × 1]>    21  82.1 3.57e- 9
##  3    43 <tibble [24 × 1]>    24  72.6 8.74e- 7
##  4    23 <tibble [23 × 1]>    23  61.0 2.71e- 5
##  5    15 <tibble [9 × 1]>      9  32.7 1.53e- 4
##  6     8 <tibble [22 × 1]>    22  47.3 1.32e- 3
##  7    48 <tibble [24 × 1]>    24  48.6 2.10e- 3
##  8    26 <tibble [24 × 1]>    24  47.2 3.16e- 3
##  9    27 <tibble [23 × 1]>    23  44.3 4.82e- 3
## 10     7 <tibble [21 × 1]>    21  31.2 7.10e- 2
## # … with 41 more rows
```

```
sum(mahalanobis_data$p_value<0.05)
```

```
## [1] 9
```

1g:
The Semi Variogram is not constant around 1. It goes up past 1 and then went below 1.

```
Variogram(model1,
          data = prog,
          form = ~ 1 + time + timeSqr| id ,
          resType = "normalized") %>%
  as_tibble() %>%
  ggplot(aes(x = dist, y = variog)) +
  geom_hline(yintercept = 1, linetype = "dashed") +
  geom_point(shape = 1) +
  geom_smooth(method = "loess", se = FALSE, span = 0.1)
```

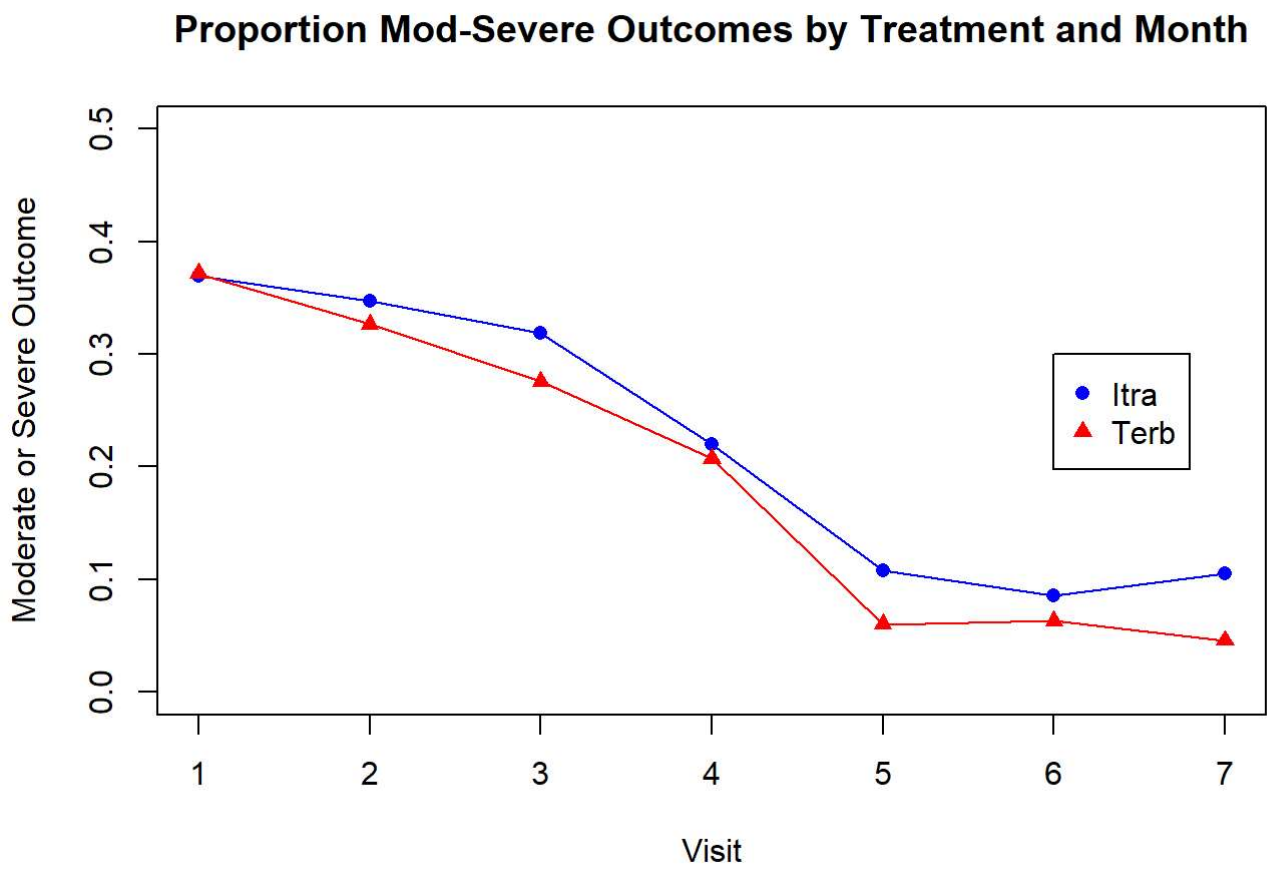


Problem 2

2a:

As month increases, the proportion of infection generally goes down for both treatments.

```
toes = read.table("./toenail-data.txt", header=FALSE)
names(toes) = c("ID","Y","Trt","Month","Visit")
toes$Trt = factor(toes$Trt, levels=c(0,1), labels=c("Itra","Terb"))
toes$ID = factor(toes$ID)

visits = c(1,2,3,4,5,6,7)
plot(visits, unlist(by(toes[toes$Trt=="Itra",]$Y, toes[toes$Trt=="Itra",5] , mean)), type="o",
   pch=16, col="blue",xlab="Visit", ylab="Moderate or Severe Outcome",
      main="Proportion Mod-Severe Outcomes by Treatment and Month", ylim=c(0,0.5))

points(visits, unlist(by(toes[toes$Trt=="Terb",]$Y, toes[toes$Trt=="Terb",5] , mean)), type="o",
   pch=17, col="red")

legend(6,.3,c("Itra","Terb"), col=c("blue","red"), pch=c(16,17))
```



2b:

$$\ln \frac{p}{1-p} = \beta_0 + \beta_1 * month_{ij} + \beta_2 * trt_i + \beta_3 * month * trt_i$$

2c:

```
mod1gee= geeglm(Y ~ 1+Month*Trt , family=binomial, id=ID, corstr="exchangeable", data=toes)

# mod2gee= geeglm(Y ~ 1+Month+Trt , family=binomial, id=ID, corstr="exchangeable", data=toes)

# mod3gee = geeglm(Y ~ 1+Trt , family=binomial, id=ID, corstr="exchangeable", data=toes)
summary(mod1gee)
```

```
##
## Call:
## geeglm(formula = Y ~ 1 + Month * Trt, family = binomial, data = toes,
##     id = ID, corstr = "exchangeable")
##
##  Coefficients:
##               Estimate  Std.err   Wald Pr(>|W|)
## (Intercept)   -0.58192  0.17206 11.439 0.000719 ***
## Month         -0.17128  0.03000 32.596 1.13e-08 ***
## TrtTerb        0.00718  0.25949  0.001 0.977924
## Month:TrtTerb -0.07773  0.05411  2.064 0.150862
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation structure = exchangeable
## Estimated Scale Parameters:
##
##             Estimate Std.err
## (Intercept)    1.088  0.5013
##   Link = identity
##
## Estimated Correlation Parameters:
##       Estimate Std.err
## alpha   0.4218  0.2119
## Number of clusters:   294  Maximum cluster size: 7
```

2d:
For the existing treatment group (Itraconazole, trt = 0):
One unit increase in month result in $e^{-0.17128}$ relative change in the odd of getting moderate/severe infection. For the new treatment group (Terbinafine, trt = 1):
One unit increase in month result in $e^{-0.17128-0.07773}$ relative change in the odd of getting moderate/severe infection.

2e:
Wald test concludes that Month should be in the model

```
V=mod1gee$geese$vbeta

beta.hat = coef(mod1gee)
L= matrix(c(0,1,0,0,0,0,0,1),2,4, byrow=TRUE)

# (Matrix multiplication in R -->  %*% )
# L %*% beta.hat
# Wald statistic to test for interaction:
# (Transpose in R --> t()
#  Matrix inversion in R --> solve() )
W2 = t(L%*%beta.hat) %*% solve(L%*%V%*%t(L)) %*% L%*%beta.hat
# approximate p-value:
pchisq(W2, df=1, lower.tail=FALSE)
```

```
##           [,1]
## [1,] 1.897e-15
```

```
# anova(mod1gee, mod3gee)
```

2f:
Because in GEE, no likelihood function is assumed for the model.

2g:
$\ln \frac{p}{1-p} = \beta_0 + \beta_1 * month_{ij} + \beta_2 * trt_i + \beta_3 * month_{ij} * trt_i + b_{0i}$

2h:

```
mod = glmer(Y ~ 1+Month*Trt  + (1 | ID), family=binomial, data=toes, nAGQ =  5)
summary(mod)
```

```
## Generalized linear mixed model fit by maximum likelihood (Adaptive
##   Gauss-Hermite Quadrature, nAGQ = 5) [glmerMod]
##  Family: binomial  ( logit )
## Formula: Y ~ 1 + Month * Trt + (1 | ID)
##    Data: toes
##
##      AIC      BIC   logLik deviance df.resid
##     1270     1298     -630     1260     1903
##
## Scaled residuals:
##    Min     1Q Median     3Q    Max
##  -3.10  -0.20  -0.10  -0.01  40.64
##
## Random effects:
##  Groups Name        Variance Std.Dev.
##  ID     (Intercept) 13.6     3.69
## Number of obs: 1908, groups:  ID, 294
##
## Fixed effects:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -1.4576     0.3947   -3.69  0.00022 ***
## Month          -0.3821     0.0434   -8.81  < 2e-16 ***
## TrtTerb        -0.1298     0.5378   -0.24  0.80925
## Month:TrtTerb  -0.1336     0.0662   -2.02  0.04343 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) Month  TrtTrb
## Month       -0.194
## TrtTerb     -0.665  0.220
## Mnth:TrtTrb  0.207 -0.565 -0.312
```

2i:
Fit another nested model with only treatment and random intercept. Then compare the AIC of the two.

2j:
For the average subject with old treatment, 1 unit increase in Month result in a relative odd change of $e^{-0.3821}$
For the average subject with new treatment, 1 unit increase in Month result in a relative odd change of $e^{-0.3821-0.1336}$

2k:
$\ln \frac{p}{1-p} = 2.017 - 0.3821 * Month_{ij} - 0.1298 * trt_i - 0.1336 * month_{ij} * trt_i$

```
coef(mod)$ID[1,]
```

```
##   (Intercept)   Month TrtTerb Month:TrtTerb
## 1       2.017 -0.3821 -0.1298       -0.1336
```

2l:
Model in part b (GEE) addresses the marginal model, and model in part g (GLMM) addresses the conditional model.

Problem 3

3a:
$\ln p = \beta_0 + \beta_1 * year_{ij} + \beta_2 * trt_i + \beta_3 * year_{ij} * trt_i$

3b:
We are looking at count per row (given an id AND a year), so each count is relating to a year of observation and we shouldn't put a offset term.

3c:

```
skin = read.csv("skin.csv")

skin$trt_num = skin$trt
skin$trt = factor(skin$trt, levels=c('0','1'),labels=c('Placebo','beta carotene'))

gee_2 = geeglm(y ~ year + trt + year*trt,data = skin,family = poisson(link = "log"),id = id,  co
rstr = "ar1")
summary(gee_2)
```

```
##
## Call:
## geeglm(formula = y ~ year + trt + year * trt, family = poisson(link = "log"),
##     data = skin, id = id, corstr = "ar1")
##
##  Coefficients:
##                       Estimate Std.err   Wald Pr(>|W|)
## (Intercept)            -1.3289  0.1234 115.93   <2e-16 ***
## year                   -0.0116  0.0329   0.12     0.73
## trtbeta carotene        0.0657  0.1644   0.16     0.69
## year:trtbeta carotene   0.0327  0.0484   0.46     0.50
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation structure = ar1
## Estimated Scale Parameters:
##
##             Estimate Std.err
## (Intercept)     2.62   0.377
##   Link = identity
##
## Estimated Correlation Parameters:
##       Estimate Std.err
## alpha    0.545   0.111
## Number of clusters:   1683  Maximum cluster size: 5
```

3d:
Going from no treatment to having a treatment, the main effect results in a relative count change of new skin cancers per year by $e^{0.0657}$

3e:
For the no treatment group, one year increase results in a relative count change of $e^{-0.0116}$

3f:
For the treatment group, one year increase will results in an additional $e^{0.0327}$ relative count change of new skin cancers per year.

3g:
$\ln p = \beta_0 + \beta_1 * year_{ij} + \beta_2 * trt_i + \beta_3 * year_{ij} * trt_i + b_{0i} + b_{1i} * year_{ij}$

3h:

```
glmm_3 = glmer(y ~ year+trt+trt*year + (1+year | id), family=poisson, data=skin , nAGQ=0)
summary(glmm_3)
```

```
## Generalized linear mixed model fit by maximum likelihood (Adaptive
##   Gauss-Hermite Quadrature, nAGQ = 0) [glmerMod]
##  Family: poisson  ( log )
## Formula: y ~ year + trt + trt * year + (1 + year | id)
##    Data: skin
##
##      AIC      BIC   logLik deviance df.resid
##     8429     8477    -4208     8415     7074
##
## Scaled residuals:
##    Min     1Q Median     3Q    Max
## -2.535 -0.359 -0.283 -0.265  3.602
##
## Random effects:
##  Groups Name        Variance Std.Dev. Corr
##  id     (Intercept) 2.416    1.555
##         year        0.101    0.317    -0.46
## Number of obs: 7081, groups:  id, 1683
##
## Fixed effects:
##                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)            -1.8569     0.1066  -17.41   <2e-16 ***
## year                   -0.0365     0.0325   -1.12     0.26
## trtbeta carotene        0.0897     0.1469    0.61     0.54
## year:trtbeta carotene   0.0209     0.0447    0.47     0.64
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) year   trtbtc
## year        -0.775
## trtbetcartn -0.726  0.563
## yr:trtbtcrt  0.563 -0.726 -0.772
```

3i:
For an average subject in the no treatment group, one year increase results in a relative count change of $e^{-0.0365}$

3j:
For an average subject in the treatment group, one year increase will results in an additional $e^{0.0209}$ relative count change of new skin cancers per year.