

Clinical Trials

Galanakis Michalis

11/7/2022

Load the package to read in the data

```
library(readxl)
```

Read in the data

```
data <- read_excel("C:/Users/mihal/OneDrive/Case study Greece dataset_2  
0220303.xlsx")
```

Descriptive Statistics

```
##=====
## Part 1a)
##=====
```

```
head(data)
```

```
## # A tibble: 6 x 5
##   id   trt   y0 time change
##   <dbl> <dbl> <dbl> <dbl>   <dbl>
## 1     1     1 37.2     1    1.88
## 2     1     1 37.2     2    3.77
## 3     1     1 37.2     3   14.4
## 4     2     1 52.0     1    1.23
## 5     2     1 52.0     2    4.16
## 6     2     1 52.0     3    6.05
```

```
# View the first 5 values of the data
```

```
tail(data)
```

```
## # A tibble: 6 x 5
##   id   trt   y0 time change
##   <dbl> <dbl> <dbl> <dbl>   <dbl>
## 1   999     0 48.6     1  0.380
## 2   999     0 48.6     2  0.616
## 3   999     0 48.6     3 -2.08
## 4 1000     0 41.7     1  3.33
## 5 1000     0 41.7     2  4.29
## 6 1000     0 41.7     3  0.867
```

```
# View the last 5 values of the data
```

```
str(data)
```

```
## tibble [3,000 x 5] (S3: tbl_df/tbl/data.frame)
```

```
## $ id      : num [1:3000] 1 1 1 2 2 2 3 3 3 4 ...
```

```
## $ trt     : num [1:3000] 1 1 1 1 1 1 1 1 1 1 ...
```

```
## $ y0      : num [1:3000] 37.2 37.2 37.2 52 52 ...
```

```
## $ time    : num [1:3000] 1 2 3 1 2 3 1 2 3 1 ...
```

```
## $ change: num [1:3000] 1.88 3.77 14.38 1.23 4.16 ...
```

```
# 3000 obs, 5 variables
```

```
dim(data)
```

```
## [1] 3000    5
```

```
# Check that the data were read in correctly
```

```
summary(data)
```

```
##           id           trt           y0           time           chan
ge
## Min.      :  1.0   Min.    :0.0   Min.    :18.92   Min.    :1   Min.    :
-20.213
## 1st Qu.: 250.8   1st Qu.:0.0   1st Qu.:43.24   1st Qu.:1   1st Qu.:
1.374
## Median : 500.5   Median :0.5   Median :49.89   Median :2   Median :
3.677
## Mean    : 500.5   Mean    :0.5   Mean    :49.89   Mean    :2   Mean    :
4.284
## 3rd Qu.: 750.2   3rd Qu.:1.0   3rd Qu.:56.45   3rd Qu.:3   3rd Qu.:
6.583
## Max.    :1000.0   Max.    :1.0   Max.    :76.61   Max.    :3   Max.    :
23.026
```

```
# Useful information regarding our data set
```

```
which(is.na(data))
```

```
## integer(0)
```

```
# There are no missing values
```

```
# Creating our variables
```

```
subject <- data$id
```

```
trt <- factor(data$trt) # 0 refers to SoC, 1 refers to Drug X
```

```
y0 <- data$y0 # Baseline value
```

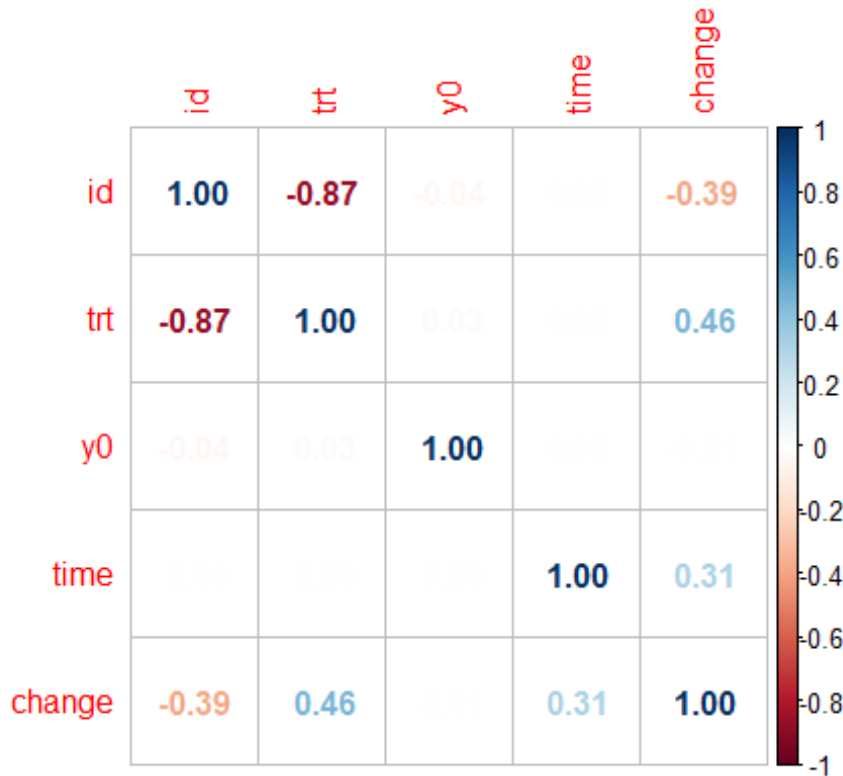
```
time <- data$time # 1 refers to Week 4, 2 refers to Week 8, 3 refers to
Week 12
```

```
change <- data$change # Target variable
```

```
# Heatmap of the correlations of our dataset
library(corrplot)
```

```
## corrplot 0.92 loaded
```

```
new.corrmatrix <- cor(data)
corrplot(new.corrmatrix, method = 'number')
```



```
# The correlations missing in the heatmap can be assumed to be close to 0
```

```
# Descriptive Statistics and Visualizations
```

```
with(data, interaction.plot(time,trt,change,type='o',pch=c(1,16)))
```

```
# We observe that as time goes by patients receiving SoC do NOT progress while patients receiving Drug X do progress!
```

```
# In time 1(4th Week) the difference in change between patients receiving SoC and Drug X is not great
```

```
# In time 2(8th Week) the difference in change between patients receiving SoC and Drug X is considerably high
```

```
# In time 3(12th Week) the difference in change between patients receiving SoC and Drug X is huge!
```

```
ftime <- as.factor(time)
```

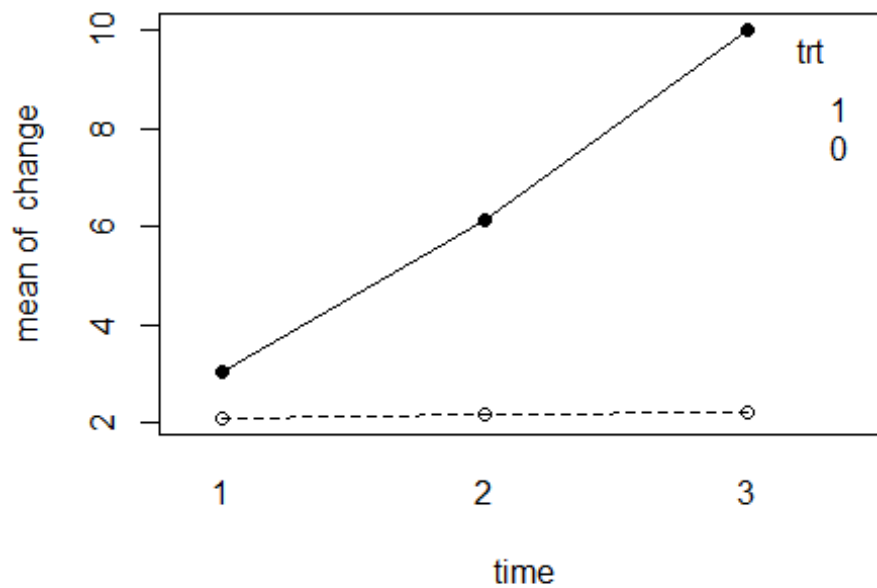
```
with(data,tapply(change,list(trt,ftime),mean))
```

```
##           1           2           3
## 0 2.101552 2.186028 2.226866
## 1 3.042159 6.122384 10.022428
```

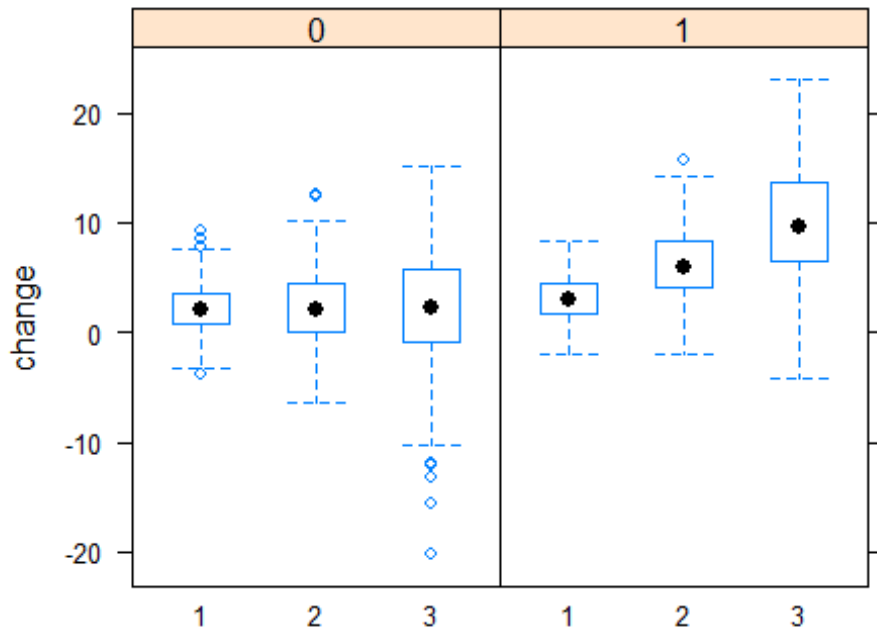
We observe that patients receiving Soc have almost the same (mean) change at each time period (indicating minor to no progress)

However patients receiving Drug X treatment seem to have (mean) change considerably higher at each time period, thus indicating that the drug X is more efficient!

```
library(lattice)
```



```
bwplot(change~ftime|trt)
```



We observe that as regards the boxplots for the first time period (which refer to Week 4) there doesn't seem to be a great difference. Some outliers (on the patients receiving the SoC) noted.

We observe that as regards the boxplots for the second time period (which refers to Week 8) they seem a bit different.

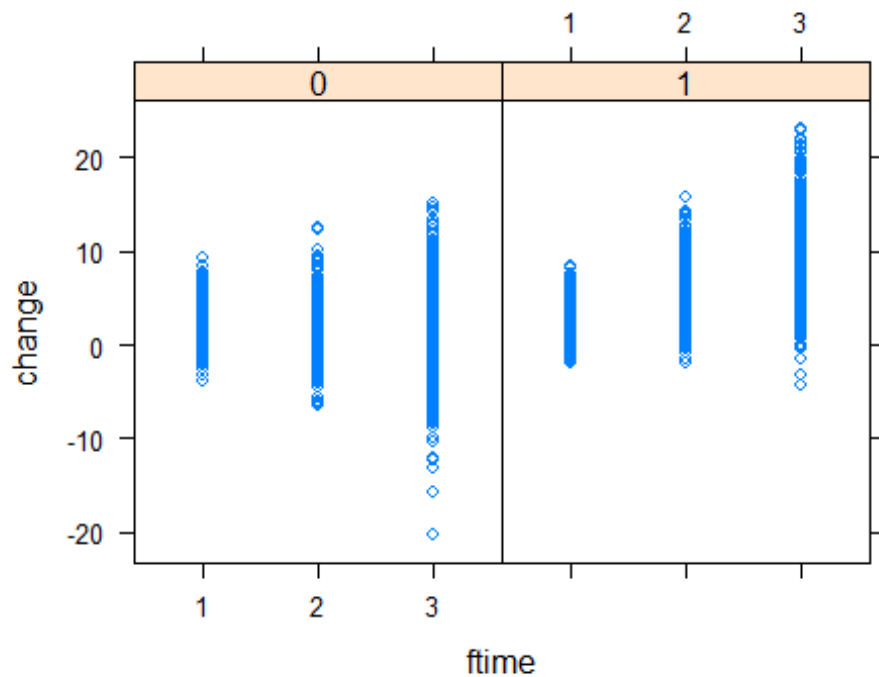
Patients receiving Drug X seem to score higher, thus having a better quality of life, in comparison to patients receiving SoC.

The variability seems to be the same between the 2 treatments.

we also observe that regarding the boxplots for the third time period (which refers to Week 12) there seems to be an overwhelming difference between the 2 treatments.

Patients receiving Drug X seem to have a considerably higher change, thus indicating that they have a much better quality of life in comparison to patients receiving SoC.

```
xyplot(change~ftime | trt)
```



Same conclusions as above

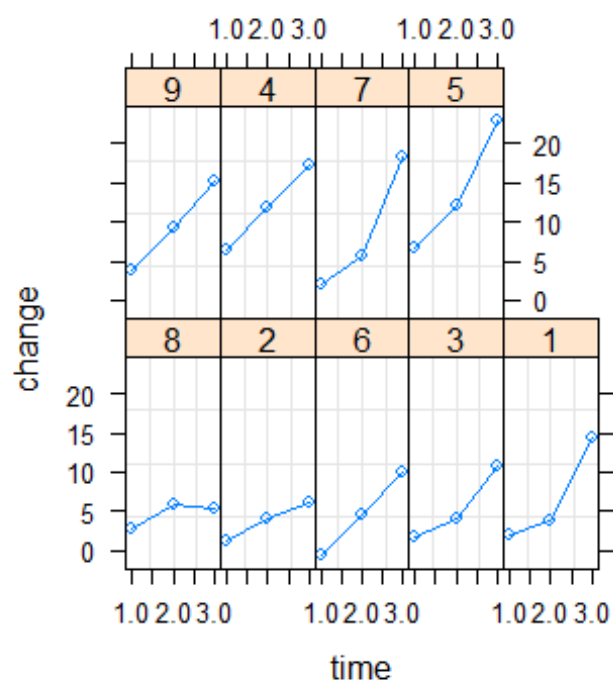
More descriptive statistics and visualizations

```
library(nlme)

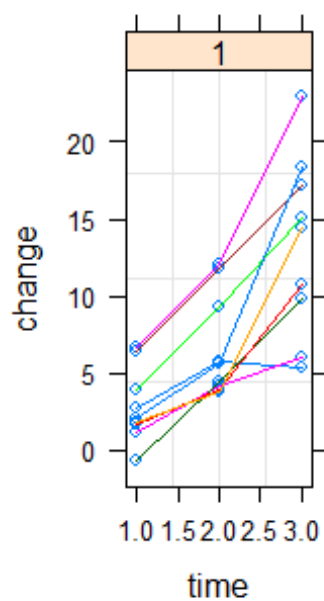
## Warning: package 'nlme' was built under R version 4.1.3

# dat <- groupedData(change~ftime|subject,outer=~as.factor(trt),data=data)
# plot(dat)
# plot(dat,outer=T)

dat2 <- groupedData(change~time|subject,outer=~as.factor(trt),data=data)
plot(dat2[1:27,]) # Profile plot per subject
```



```
plot(dat2[1:27,],outer=T) # Profile plot per drug
```



Profile plots are quite informative since we can have indications regarding the heterogeneity of each subject, the general trend of the variance!

Useful plot for tracking purposes

```
head(data,10)
```

```
## # A tibble: 10 x 5
##       id    trt    y0  time change
##   <dbl> <dbl> <dbl> <dbl> <dbl>
## 1     1     1   37.2     1   1.88
## 2     1     1   37.2     2   3.77
## 3     1     1   37.2     3  14.4
## 4     2     1   52.0     1   1.23
## 5     2     1   52.0     2   4.16
## 6     2     1   52.0     3   6.05
## 7     3     1   50.4     1   1.72
## 8     3     1   50.4     2   3.98
## 9     3     1   50.4     3  10.8
## 10    4     1   63.5     1   6.47
```

Data manipulation (long to wide format)

```
data21 <- data[data[,4]==1,5]
head(data21)
```

```
## # A tibble: 6 x 1
##   change
##   <dbl>
## 1  1.88
## 2  1.23
## 3  1.72
## 4  6.47
## 5  6.63
## 6 -0.683
```

```
data22 <- data[data[,4]==2,5]
head(data22)
```

```
## # A tibble: 6 x 1
##   change
##   <dbl>
## 1  3.77
## 2  4.16
## 3  3.98
## 4 11.7
## 5 12.1
## 6  4.52
```

```
data23 <- data[data[,4]==3,5]
head(data23)
```



```

## # A tibble: 6 x 1
##   change
##   <dbl>
## 1  14.4
## 2   6.05
## 3  10.8
## 4  17.2
## 5  23.0
## 6   9.87

treat <- data[data[,4]==1,2]
head(treat)

## # A tibble: 6 x 1
##   trt
##   <dbl>
## 1     1
## 2     1
## 3     1
## 4     1
## 5     1
## 6     1

vector_trt <- treat$trt
length(vector_trt)

## [1] 1000

vector_trt <- as.factor(vector_trt)

subject2 <- data[data[,4]==1,1]
vec_subject <- subject2$id
length(vec_subject)

## [1] 1000

y0 <- data[data[,4]==1,3]
y0 <- y0$y0
length(y0)

## [1] 1000

data2 <- cbind(data21,data22,data23,vector_trt,vec_subject,y0)
colnames(data2) <- c('Change in Week 4', 'Change in Week 8', 'Change in W
eek 12', 'trt', 'subject', 'y0')
head(data2)

##   Change in Week 4 Change in Week 8 Change in Week 12 trt subject
y0
## 1          1.8842358          3.767594          14.381247   1      1 37
.22891
## 2          1.2310059          4.157039          6.050582   1      2 52

```

```
.00300
## 3      1.7246401      3.981285      10.783979      1      3 50
.39272
## 4      6.4721330      11.730751      17.179892      1      4 63
.48535
## 5      6.6308687      12.064772      22.968183      1      5 42
.58701
## 6     -0.6829066      4.516765      9.873356      1      6 47
.10957

dim(data2)

## [1] 1000      6
```

Summary measures analysis

```
mean_4_8_12 <- (data2[,1]+data2[,2]+data2[,3])/3 # Create a summary mea
sure

# Post analysis
mod1 <- glm(mean_4_8_12~vector_trt,gaussian)
summary(mod1)

##
## Call:
## glm(formula = mean_4_8_12 ~ vector_trt, family = gaussian)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -10.2712   -1.9112   -0.0115    1.9506   10.1780
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    2.1715     0.1267   17.14  <2e-16 ***
## vector_trt1    4.2242     0.1791   23.58  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 8.021502)
##
##      Null deviance: 12466.4  on 999  degrees of freedom
## Residual deviance:  8005.5  on 998  degrees of freedom
## AIC: 4924
##
## Number of Fisher Scoring iterations: 2

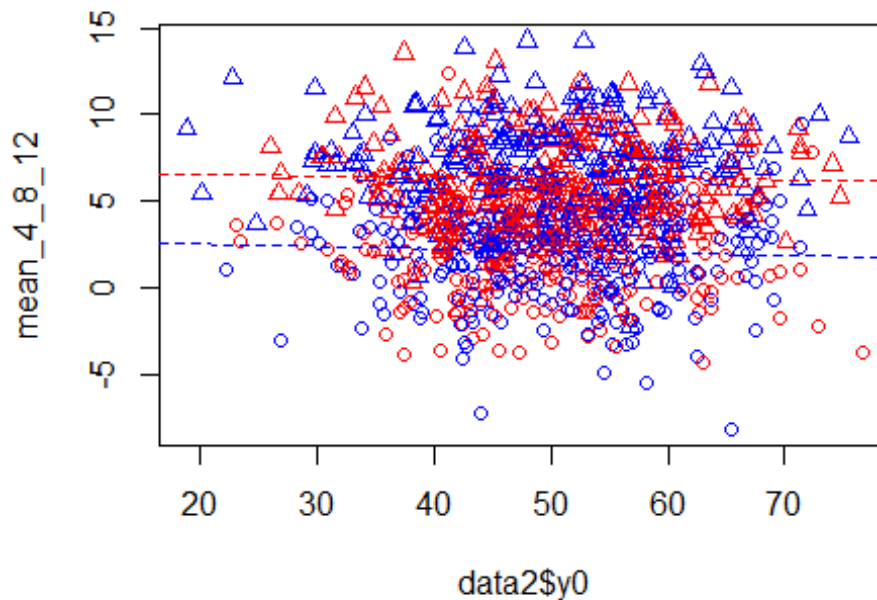
# We observe that the effect of the factor treatment is statistically s
ignificant!

plot(data2$y0,mean_4_8_12,pch=0+as.numeric(vector_trt),col=c('blue','re
d'))
```

We observe that the scores in Weeks 4,8,12 (mean scores) are highly correlated with the baseline scores

```
abline(lm(mean_4_8_12~data2$y0,subset=(vector_trt==0)),lty=2,col='blue'
)
```

```
abline(lm(mean_4_8_12~data2$y0,subset=(vector_trt==1)),lty=2,col='red')
```



*# We observe that the slopes are parallel!
That means that the interaction term (with the baseline value) is expected to be statistically insignificant!
Let us see:*

Ancova analysis (GLM with gaussian family would be the same)

```
model1 <- lm(mean_4_8_12~vector_trt*y0)
summary(model1)
```

```
##
```

```
## Call:
```

```
## lm(formula = mean_4_8_12 ~ vector_trt * y0)
```

```
##
```

```
## Residuals:
```

```
##      Min       1Q   Median       3Q      Max
## -10.0619  -1.9255  -0.0228   1.9812  10.0669
```

```
##
```

```
## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.832794    0.670846   4.223 2.63e-05 ***
## vector_trt1   3.885075    0.939445   4.136 3.84e-05 ***
## y0            -0.013321    0.013270  -1.004   0.316
## vector_trt1:y0  0.006895    0.018486   0.373   0.709
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.833 on 996 degrees of freedom
## Multiple R-squared:  0.3586, Adjusted R-squared:  0.3567
## F-statistic: 185.7 on 3 and 996 DF,  p-value: < 2.2e-16

# We observe that baseline and the interaction term are not statistical
ly significant

model2 <- lm(mean_4_8_12~vector_trt) # Model without the interaction te
rm and the baseline value
summary(model2)

##
## Call:
## lm(formula = mean_4_8_12 ~ vector_trt)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10.2712  -1.9112  -0.0115   1.9506  10.1780
##
## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.1715     0.1267  17.14  <2e-16 ***
## vector_trt1   4.2242     0.1791  23.58  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.832 on 998 degrees of freedom
## Multiple R-squared:  0.3578, Adjusted R-squared:  0.3572
## F-statistic: 556.1 on 1 and 998 DF,  p-value: < 2.2e-16

# We observe that there's a difference between the 2 treatments! The es
timated difference is 4.22 (statistically significant)

# Another simple analysis (change analysis)
model3 <- lm(mean_4_8_12~vector_trt,offset=y0)
summary(model3)

##
## Call:
```

```
## lm(formula = mean_4_8_12 ~ vector_trt, offset = y0)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -32.865  -7.056   0.138   6.627  34.051
##
## Coefficients:
##              Estimate Std. Error  t value Pr(>|t|)
## (Intercept) -47.4727     0.4563 -104.040  < 2e-16 ***
## vector_trt1  3.7260     0.6453   5.774 1.03e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.2 on 998 degrees of freedom
## Multiple R-squared:  0.4867, Adjusted R-squared:  0.4862
## F-statistic: 946.3 on 1 and 998 DF,  p-value: < 2.2e-16

# Once again the treatment factor is statistically significant
```

Different summary measure (Slopes)

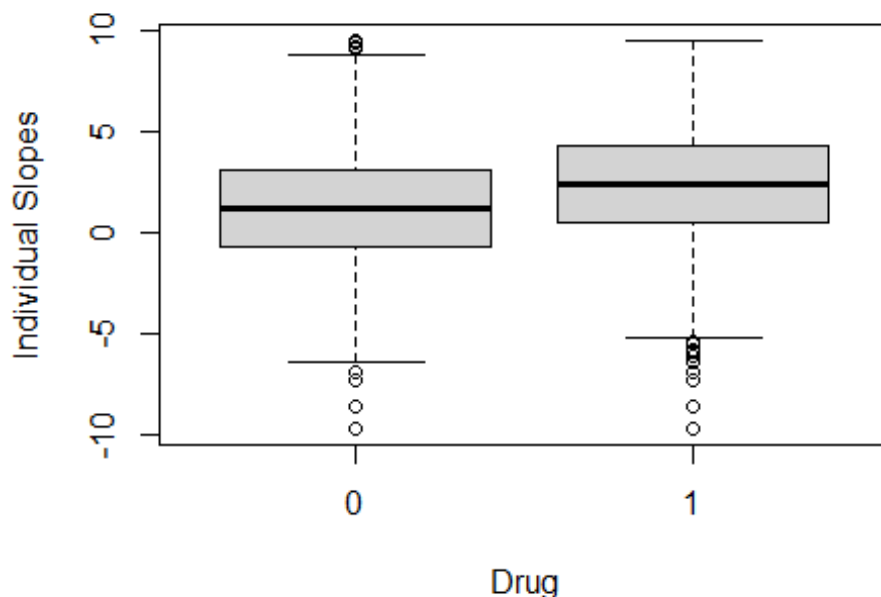
```
library(nlme)
id <- data$id
sepcoef <- lmList(change~time|id,data) # Fit a linear model on time for
each subject
coef.emp <- coef(sepcoef) # ALL coefficients per subject
drug <- data$trt
subj <- data$id
data3 <- cbind(coef.emp,subj,drug)

## Warning in data.frame(..., check.names = FALSE): row names were found from a
## short variable and have been discarded

head(data3)

##      (Intercept)      time subj drug
## 1  -5.819319  6.248506    1    1
## 2  -1.006700  2.409788    1    1
## 3  -3.562704  4.529669    1    1
## 4   1.086499  5.353880    2    1
## 5  -2.449373  8.168657    2    1
## 6  -5.987191  5.278131    2    1

boxplot(data3[,2]~data3[,4],xlab = 'Drug', ylab = 'Individual Slopes')
```



We observe that the median of the slopes of the patients receiving Drug X (1 refers to Drug X) is higher than the median of the slopes of the patients receiving SoC treatment
Variability is approximately the same between the 2 treatment groups
There are some outliers observed in both treatment groups

We can check that also by fitting a glm as follows:

```
model3 <- lm(time~as.factor(drug),data=data3)
summary(model3)
```

```
##
## Call:
## lm(formula = time ~ as.factor(drug), data = data3)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.0449  -1.9081   0.0236   1.9403   8.2916
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    1.20515    0.07416  16.25  <2e-16 ***
## as.factor(drug)1  1.14249    0.10488  10.89  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.872 on 2998 degrees of freedom
```

```
## Multiple R-squared:  0.03807,    Adjusted R-squared:  0.03775
## F-statistic: 118.7 on 1 and 2998 DF,  p-value: < 2.2e-16

# We observe that the estimated difference between patients receiving Drug X and those receiving SoC treatment, is 1.14 (statistically significant!)
# That is to mean that patients receiving Drug X have higher mean slopes!
# We can also infer that the progress over time is NOT the same for the 2 treatment groups!

# We also observe that:
var(coef(sepcoef)[1])

##                (Intercept)
## (Intercept)      16.06361

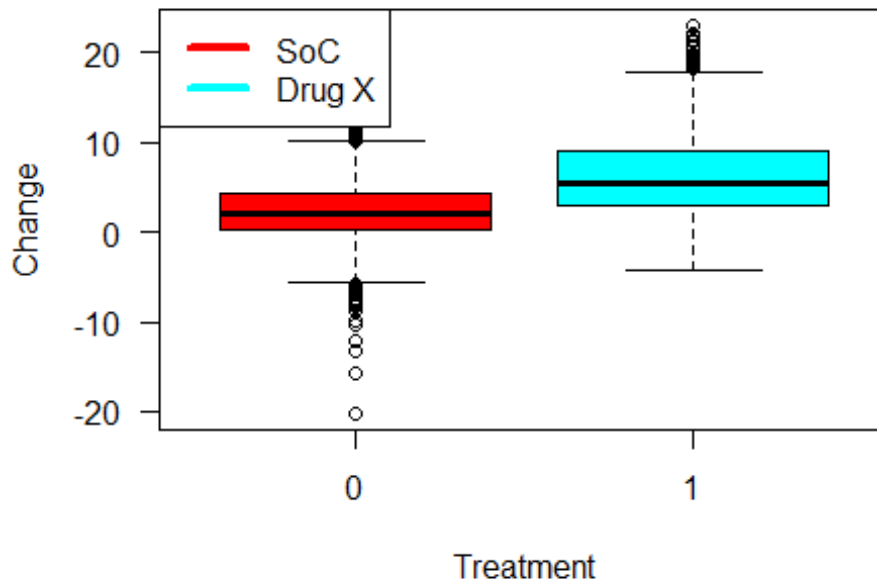
var(coef(sepcoef)[2])

##                time
## time 8.579126
```

More visualizations

```
# Let us visualize the impact of different treatment groups on change variable across all time points
boxplot(change~trt, col=rainbow(2), las=1, main='Boxplots of both trt arms across all time points', ylab = 'Change', xlab='Treatment')
legend("topleft", c('SoC', 'Drug X'), lwd=4, col=c('red', 'cyan'))
```

Boxplots of both trt arms across all time points



A boxplot is a standardized way of displaying the distribution of data based on a five number summary
(“minimum”, first quartile (Q1), median, third quartile (Q3), and “maximum”). It can tell you about your outliers
and what their values are. It can also tell you if your data is symmetrical, how tightly your data is grouped, and
if and how your data is skewed.
We notice that the data are equally dispersed above and below the median.
We observe that there are outliers in both groups!
However, in treatment arm 1 (which refers to patients receiving Drug X) we notice there are outliers only in the higher extreme, which is positive as these values refer to better quality of life!
We observe that there is no visible difference between the 2 groups of variable treatment as regards their impact on change!
Moreover, we observe that variability seems to be approximately the same between the 2 groups.
Plus, note that the scores of patients receiving Drug X seem to be higher in the 0-100 scale of the QoL scores in comparison to patients receiving SoC treatment!

Create a new data frame, one for each treatment


```

new_data_drug_x <- data[data$trt==1,]
table(new_data_drug_x$trt)

##
##      1
## 1500

# Confirmation that we selected the patients receiving Drug X

new_data_soc <- data[data$trt==0,]
table(new_data_soc$trt)

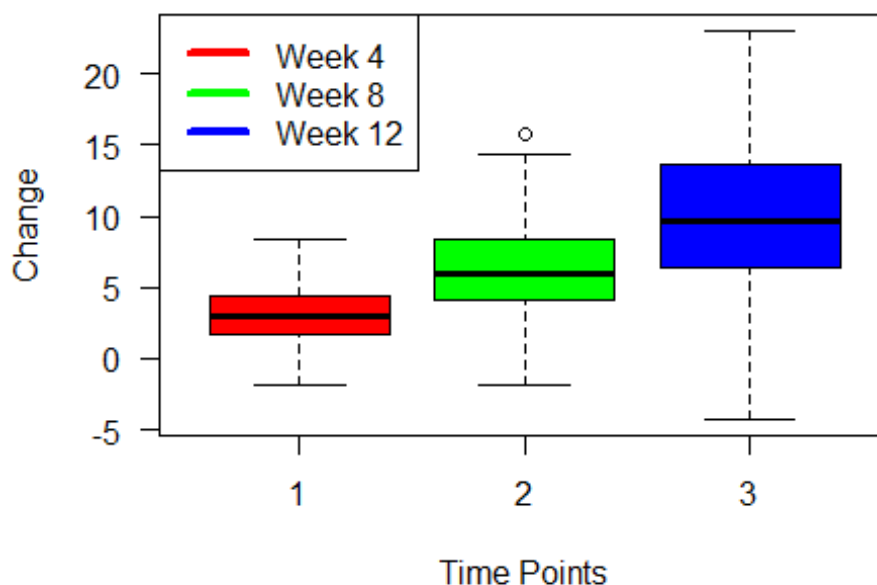
##
##      0
## 1500

# Confirmation that we selected the patients receiving SoC

# Let us visualize the impact of different time points on change variable for patients receiving treatment Drug X
boxplot(new_data_drug_x$change~new_data_drug_x$time, col=rainbow(3), las=1, main='Boxplots of all Time Points for patients receiving Drug X', ylab = 'Change', xlab='Time Points')
legend("topleft", c('Week 4', 'Week 8', 'Week 12'), lwd=4, col=c('red', 'green', 'blue'))

```

Boxplots of all Time Points for patients receiving Dri

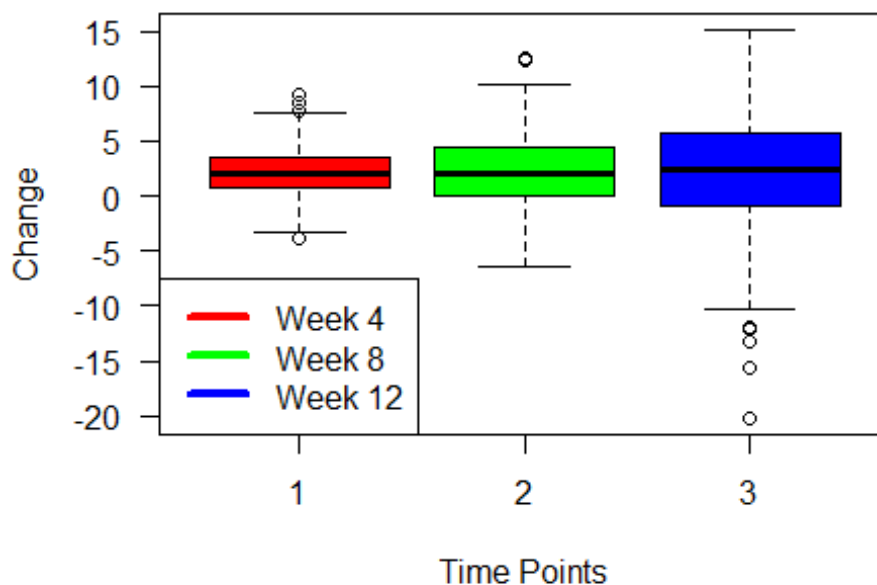


It is clear that time points 1 and 3 differ.
 # As time goes by, the variability of change increases.
 # The majority of patients receiving Drug X at week 12 seem to have positive change scores thus indicating a better quality of life

Let us visualize the impact of different time points on change variable for patients receiving treatment SoC

```
boxplot(new_data_soc$change~new_data_soc$time, col=rainbow(3), las=1, main='Boxplots of all Time Points for patients receiving SoC', ylab = 'Change', xlab='Time Points')
legend("bottomleft", c('Week 4', 'Week 8', 'Week 12'), lwd=4, col=c('red', 'green', 'blue'))
```

Boxplots of all Time Points for patients receiving S



It is clear that time points do not differ in this case
 # As time goes by, the variability of change increases.
 # In this case, we observe that the majority of patients receiving SoC do not have significantly higher change values
 # hence they do not have a better quality of life, which is the overall goal
 # Therefore, this is another proof that Drug X is more effective than SoC overall

Fitting marginal models

```
##=====
## Part 1b)
##=====
head(data)

## # A tibble: 6 x 5
##   id   trt   y0 time change
##   <dbl> <dbl> <dbl> <dbl> <dbl>
## 1     1     1 37.2     1  1.88
## 2     1     1 37.2     2  3.77
## 3     1     1 37.2     3 14.4
## 4     2     1 52.0     1  1.23
## 5     2     1 52.0     2  4.16
## 6     2     1 52.0     3  6.05

m0.gls <- gls(change~trt,weights=NULL,method="ML",data=data) # Independent, homoscedastic data
summary(m0.gls) # AIC = 17019.9

## Generalized least squares fit by maximum likelihood
##   Model: change ~ trt
##   Data: data
##       AIC       BIC    logLik
## 17019.9 17037.92 -8506.951
##
## Coefficients:
##              Value Std.Error   t-value p-value
## (Intercept) 2.171482 0.1065048 20.38858      0
## trt          4.224175 0.1506206 28.04514      0
##
## Correlation:
##   (Intr)
## trt -0.707
##
## Standardized residuals:
##       Min       Q1       Med       Q3       Max
## -5.4285660 -0.6594284 -0.1025674  0.5516608  4.0330874
##
## Residual standard error: 4.123539
## Degrees of freedom: 3000 total; 2998 residual

# Same as linear model or glm with gaussian family
AIC_MARGINAL_MODEL_1 <- extractAIC(m0.gls)[2]

m1.gls <- gls(change~trt*ftime + y0,correlation=corCompSymm(form=~1|subject),method="ML",data=data) # Compound symmetry correlation matrix
summary(m1.gls) # R = 42.87% , AIC = 15705.87
```

```

## Generalized least squares fit by maximum likelihood
##   Model: change ~ trt * ftime + y0
##   Data: data
##           AIC      BIC    logLik
##   15705.87 15759.93 -7843.936
##
## Correlation Structure: Compound symmetry
## Formula: ~1 | subject
## Parameter estimate(s):
##      Rho
## 0.4287999
##
## Coefficients:
##              Value Std.Error   t-value p-value
## (Intercept)  2.586481 0.4857190   5.325056  0.0000
## trt          0.945473 0.2275936   4.154217  0.0000
## ftime2       0.084476 0.1719751   0.491213  0.6233
## ftime3       0.125314 0.1719751   0.728674  0.4663
## y0          -0.009768 0.0092316  -1.058116  0.2901
## trt:ftime2    2.995749 0.2432095  12.317567  0.0000
## trt:ftime3    6.854956 0.2432095  28.185395  0.0000
##
## Correlation:
##           (Intr) trt      ftime2 ftime3 y0      trt:f2
## trt       -0.215
## ftime2    -0.177  0.378
## ftime3    -0.177  0.378  0.500
## y0        -0.944 -0.020  0.000  0.000
## trt:ftime2 0.125 -0.534 -0.707 -0.354  0.000
## trt:ftime3 0.125 -0.534 -0.354 -0.707  0.000  0.500
##
## Standardized residuals:
##           Min           Q1           Med           Q3           Max
## -6.20175096 -0.57355827 -0.02488356  0.58786327  3.62598121
##
## Residual standard error: 3.593636
## Degrees of freedom: 3000 total; 2993 residual

getVarCov(m1.gls)

## Marginal variance covariance matrix
##           [,1]  [,2]  [,3]
## [1,] 12.9140  5.5376  5.5376
## [2,]  5.5376 12.9140  5.5376
## [3,]  5.5376  5.5376 12.9140
##   Standard Deviations: 3.5936 3.5936 3.5936

AIC_MARGINAL_MODEL_2 <- extractAIC(m1.gls)[2]

```

```

m2.gls <- gls(change~trt*ftime + y0,correlation=corAR1(form=~1|subject)
,method="ML",data=data) # Autoregressive of order 1
summary(m2.gls) # R = 58% , AIC = 15480.07

## Generalized least squares fit by maximum likelihood
## Model: change ~ trt * ftime + y0
## Data: data
## AIC BIC logLik
## 15480.07 15534.13 -7731.037
##
## Correlation Structure: AR(1)
## Formula: ~1 | subject
## Parameter estimate(s):
## Phi
## 0.5803187
##
## Coefficients:
## Value Std.Error t-value p-value
## (Intercept) 2.533363 0.5053338 5.013248 0.0000
## trt 0.944940 0.2312177 4.086798 0.0000
## ftime2 0.084476 0.1497571 0.564089 0.5727
## ftime3 0.125314 0.1882607 0.665639 0.5057
## y0 -0.008698 0.0096319 -0.903057 0.3666
## trt:ftime2 2.995749 0.2117885 14.145004 0.0000
## trt:ftime3 6.854956 0.2662409 25.747196 0.0000
##
## Correlation:
## (Intr) trt ftime2 ftime3 y0 trt:f2
## trt -0.209
## ftime2 -0.148 0.324
## ftime3 -0.186 0.407 0.629
## y0 -0.946 -0.021 0.000 0.000
## trt:ftime2 0.105 -0.458 -0.707 -0.444 0.000
## trt:ftime3 0.132 -0.576 -0.444 -0.707 0.000 0.629
##
## Standardized residuals:
## Min Q1 Med Q3 Max
## -6.10921710 -0.56252911 -0.02449452 0.57718053 3.56838840
##
## Residual standard error: 3.650819
## Degrees of freedom: 3000 total; 2993 residual

getVarCov(m2.gls)

## Marginal variance covariance matrix
## [,1] [,2] [,3]
## [1,] 13.3280 7.7348 4.4886
## [2,] 7.7348 13.3280 7.7348
## [3,] 4.4886 7.7348 13.3280
## Standard Deviations: 3.6508 3.6508 3.6508

```

```

AIC_MARGINAL_MODEL_3 <- extractAIC(m2.gls)[2]

m2.gls.heter <- gls(change~trt*ftime + y0,correlation=corAR1(form=~1|subject),weights=varIdent(form=~1|ftime),method="ML",data=data)
summary(m2.gls.heter) # R = 67.6% , AIC = 14213.04

## Generalized least squares fit by maximum likelihood
## Model: change ~ trt * ftime + y0
## Data: data
##      AIC      BIC    logLik
## 14213.04 14279.11 -7095.518
##
## Correlation Structure: AR(1)
## Formula: ~1 | subject
## Parameter estimate(s):
##      Phi
## 0.6761418
## Variance function:
## Structure: Different standard deviations per stratum
## Formula: ~1 | ftime
## Parameter estimates:
##      1      2      3
## 1.000000 1.633186 2.801374
##
## Coefficients:
##              Value Std.Error   t-value p-value
## (Intercept)  2.184395 0.3178453   6.872511  0.0000
## trt          0.941438 0.1204847   7.813755  0.0000
## ftime2       0.084476 0.1028652   0.821234  0.4116
## ftime3       0.125314 0.2135371   0.586848  0.5574
## y0          -0.001669 0.0061683  -0.270533  0.7868
## trt:ftime2   2.995749 0.1454733  20.593114  0.0000
## trt:ftime3   6.854956 0.3019870  22.699506  0.0000
##
## Correlation:
##      (Intr) trt      ftime2 ftime3 y0      trt:f2
## trt      -0.165
## ftime2    0.023 -0.061
## ftime3    0.030 -0.079  0.564
## y0       -0.963 -0.026  0.000  0.000
## trt:ftime2 -0.016  0.086 -0.707 -0.399  0.000
## trt:ftime3 -0.021  0.112 -0.399 -0.707  0.000  0.564
##
## Standardized residuals:
##      Min      Q1      Med      Q3      Max
## -4.20626160 -0.67232839 -0.03397613  0.69938955  3.78393721
##
## Residual standard error: 1.902188
## Degrees of freedom: 3000 total; 2993 residual

```

```

getVarCov(m2.gls.heter)

## Marginal variance covariance matrix
##      [,1]      [,2]      [,3]
## [1,] 3.6183   3.9956   4.634
## [2,] 3.9956   9.6511  11.193
## [3,] 4.6340  11.1930  28.395
##      Standard Deviations: 1.9022 3.1066 5.3287

AIC_MARGINAL_MODEL_4 <- extractAIC(m2.gls.heter)[2]

m3.gls <- gls(change~trt*ftime + y0,correlation=corSymm(form=~1|subject
),method="ML",data=data) # Unstructured Cov matrix
summary(m3.gls) # AIC = 14884.43

## Generalized least squares fit by maximum likelihood
##      Model: change ~ trt * ftime + y0
##      Data: data
##           AIC      BIC    logLik
##  14884.43 14950.5 -7431.213
##
## Correlation Structure: General
## Formula: ~1 | subject
## Parameter estimate(s):
## Correlation:
##      1      2
## 2 0.884
## 3 0.166 0.347
##
## Coefficients:
##              Value Std.Error   t-value p-value
## (Intercept)  2.385056 0.5058446   4.714997  0.0000
## trt          0.943452 0.2445770   3.857485  0.0001
## ftime2       0.084476 0.0833037   1.014078  0.3106
## ftime3       0.125314 0.2233535   0.561056  0.5748
## y0          -0.005711 0.0095756  -0.596379  0.5510
## trt:ftime2   2.995749 0.1178092  25.428825  0.0000
## trt:ftime3   6.854956 0.3158696  21.701852  0.0000
##
## Correlation:
##              (Intr) trt      ftime2 ftime3 y0      trt:f2
## trt          -0.223
## ftime2       -0.082  0.170
## ftime3       -0.221  0.457  0.478
## y0           -0.940 -0.020  0.000  0.000
## trt:ftime2   0.058 -0.241 -0.707 -0.338  0.000
## trt:ftime3   0.156 -0.646 -0.338 -0.707  0.000  0.478
##

```

```

## Standardized residuals:
##           Min           Q1           Med           Q3           Max
## -5.78752780 -0.53206572 -0.02501618  0.54825292  3.37128246
##
## Residual standard error: 3.861852
## Degrees of freedom: 3000 total; 2993 residual

getVarCov(m3.gls)

## Marginal variance covariance matrix
##           [,1]      [,2]      [,3]
## [1,] 14.9140 13.1830  2.4713
## [2,] 13.1830 14.9140  5.1771
## [3,]  2.4713  5.1771 14.9140
## Standard Deviations: 3.8619 3.8619 3.8619

AIC_MARGINAL_MODEL_5 <- extractAIC(m3.gls)[2]

m3.gls.heter <- gls(change~trt*ftime + y0,correlation=corSymm(form=~1|s
subject),weights=varIdent(form=~1|ftime),method="ML",data=data) # Unstru
ctured Cov matrix, heteroscedastic
summary(m3.gls.heter) # AIC = 14006.02

## Generalized least squares fit by maximum likelihood
## Model: change ~ trt * ftime + y0
## Data: data
##           AIC      BIC    logLik
## 14006.02 14084.1 -6990.009
##
## Correlation Structure: General
## Formula: ~1 | subject
## Parameter estimate(s):
## Correlation:
##   1      2
## 2 0.804
## 3 0.327 0.535
## Variance function:
## Structure: Different standard deviations per stratum
## Formula: ~1 | ftime
## Parameter estimates:
##           1      2      3
## 1.000000 1.545533 2.490193
##
## Coefficients:
##           Value Std.Error   t-value p-value
## (Intercept) 2.047604 0.3200922   6.396919  0.0000
## trt          0.940065 0.1273622   7.381035  0.0000
## ftime2       0.084476 0.0855852   0.987045  0.3237

```



```

## ftime3      0.125314 0.2125411 0.589598 0.5555
## y0          0.001087 0.0061874 0.175629 0.8606
## trt:ftime2  2.995749 0.1210357 24.750947 0.0000
## trt:ftime3  6.854956 0.3005785 22.805876 0.0000
##
## Correlation:
##      (Intr) trt      ftime2 ftime3 y0      trt:f2
## trt      -0.176
## ftime2    0.072 -0.180
## ftime3   -0.022 0.056 0.447
## y0        -0.960 -0.024 0.000 0.000
## trt:ftime2 -0.051 0.255 -0.707 -0.316 0.000
## trt:ftime3 0.016 -0.079 -0.316 -0.707 0.000 0.447
##
## Standardized residuals:
##      Min      Q1      Med      Q3      Max
## -4.48486603 -0.67604421 -0.03312136 0.70171419 3.59092048
##
## Residual standard error: 2.010834
## Degrees of freedom: 3000 total; 2993 residual

getVarCov(m3.gls.heter)

## Marginal variance covariance matrix
##      [,1] [,2] [,3]
## [1,] 4.0435 5.0240 3.2915
## [2,] 5.0240 9.6585 8.3306
## [3,] 3.2915 8.3306 25.0740
## Standard Deviations: 2.0108 3.1078 5.0074

AIC_MARGINAL_MODEL_6 <- extractAIC(m3.gls.heter)[2]

m4.gls <- gls(change~trt*ftime + y0,correlation=corARMA(0.8,form=~1|subject,q=1),method="ML",data=data) # MA(1) (banded) correlation matrix, homoscedastic errors
summary(m4.gls) # R = 70% , AIC = 15568.97

## Generalized least squares fit by maximum likelihood
## Model: change ~ trt * ftime + y0
## Data: data
##      AIC      BIC      logLik
## 15568.97 15623.03 -7775.487
##
## Correlation Structure: ARMA(0,1)
## Formula: ~1 | subject
## Parameter estimate(s):
## Theta1
## 0.7072953

```

```
##
## Coefficients:
##              Value Std.Error   t-value p-value
## (Intercept)  2.492769 0.4379371  5.692072  0.0000
## trt          0.944533 0.2257066  4.184781  0.0000
## ftime2       0.084476 0.1640655  0.514894  0.6067
## ftime3       0.125314 0.2256695  0.555298  0.5787
## y0          -0.007880 0.0082151 -0.959265  0.3375
## trt:ftime2   2.995749 0.2320237 12.911392  0.0000
## trt:ftime3   6.854956 0.3191449 21.479135  0.0000
##
## Correlation:
##      (Intr) trt      ftime2 ftime3 y0      trt:f2
## trt      -0.241
## ftime2   -0.187  0.363
## ftime3   -0.258  0.500  0.688
## y0       -0.931 -0.018  0.000  0.000
## trt:ftime2 0.132 -0.514 -0.707 -0.486  0.000
## trt:ftime3 0.182 -0.707 -0.486 -0.707  0.000  0.688
##
## Standardized residuals:
##           Min           Q1           Med           Q3           Max
## -6.26167194 -0.57840938 -0.02396143  0.59104314  3.65470628
##
## Residual standard error: 3.563983
## Degrees of freedom: 3000 total; 2993 residual

getVarCov(m4.gls)

## Marginal variance covariance matrix
##           [,1]  [,2]  [,3]
## [1,] 12.7020  5.9883  0.0000
## [2,]  5.9883 12.7020  5.9883
## [3,]  0.0000  5.9883 12.7020
## Standard Deviations: 3.564 3.564 3.564

AIC_MARGINAL_MODEL_7 <- extractAIC(m4.gls)[2]

m5.gls <- gls(change~trt*ftime + y0,correlation=corARMA(c(0.8,0.8),form
=~1|subject,q=2),method="ML",data=data) # MA(2) (banded) correlation ma
trix, homoscedastic errors
summary(m5.gls) # AIC = 15467.21

## Generalized least squares fit by maximum likelihood
## Model: change ~ trt * ftime + y0
## Data: data
##           AIC           BIC      logLik
```

```

## 15467.21 15527.27 -7723.603
##
## Correlation Structure: ARMA(0,2)
## Formula: ~1 | subject
## Parameter estimate(s):
##   Theta1   Theta2
## 0.6684490 0.4547901
##
## Coefficients:
##              Value Std.Error   t-value p-value
## (Intercept)  2.511347 0.5000698   5.021993  0.0000
## trt          0.944719 0.2321576   4.069301  0.0000
## ftime2       0.084476 0.1489733   0.567057  0.5707
## ftime3       0.125314 0.1976311   0.634079  0.5261
## y0          -0.008255 0.0095151  -0.867532  0.3857
## trt:ftime2   2.995749 0.2106801  14.219424  0.0000
## trt:ftime3   6.854956 0.2794925  24.526436  0.0000
##
## Correlation:
##              (Intr) trt    ftime2 ftime3 y0      trt:f2
## trt          -0.213
## ftime2       -0.149  0.321
## ftime3       -0.198  0.426  0.663
## y0           -0.945 -0.020  0.000  0.000
## trt:ftime2   0.105 -0.454 -0.707 -0.469  0.000
## trt:ftime3   0.140 -0.602 -0.469 -0.707  0.000  0.663
##
## Standardized residuals:
##              Min           Q1           Med           Q3           Max
## -6.08634233 -0.56184798 -0.02356306  0.57468339  3.55358727
##
## Residual standard error: 3.665685
## Degrees of freedom: 3000 total; 2993 residual

getVarCov(m5.gls)

## Marginal variance covariance matrix
##           [,1]  [,2]  [,3]
## [1,] 13.4370  7.9019  3.6955
## [2,]  7.9019 13.4370  7.9019
## [3,]  3.6955  7.9019 13.4370
## Standard Deviations: 3.6657 3.6657 3.6657

AIC_MARGINAL_MODEL_8 <- extractAIC(m5.gls)[2]

m6.gls <- gls(change~trt*ftime + y0,correlation=corARMA(c(0.8,0.8),form

```

```

=~1|subject,p=1,q=1),method='ML',data=data) # ARMA(1,1)
summary(m6.gls) # AIC = 15467.21

## Generalized least squares fit by maximum likelihood
## Model: change ~ trt * ftime + y0
## Data: data
##      AIC      BIC    logLik
## 15467.21 15527.27 -7723.603
##
## Correlation Structure: ARMA(1,1)
## Formula: ~1 | subject
## Parameter estimate(s):
##      Phi1      Theta1
## 0.4676731 0.1862880
##
## Coefficients:
##              Value Std.Error   t-value p-value
## (Intercept)  2.511347 0.5000698   5.021993  0.0000
## trt          0.944719 0.2321576   4.069301  0.0000
## ftime2       0.084476 0.1489733   0.567057  0.5707
## ftime3       0.125314 0.1976311   0.634079  0.5261
## y0          -0.008255 0.0095151  -0.867532  0.3857
## trt:ftime2   2.995749 0.2106801  14.219424  0.0000
## trt:ftime3   6.854956 0.2794925  24.526436  0.0000
##
## Correlation:
##      (Intr) trt      ftime2 ftime3 y0      trt:f2
## trt      -0.213
## ftime2   -0.149  0.321
## ftime3   -0.198  0.426  0.663
## y0       -0.945 -0.020  0.000  0.000
## trt:ftime2 0.105 -0.454 -0.707 -0.469  0.000
## trt:ftime3 0.140 -0.602 -0.469 -0.707  0.000  0.663
##
## Standardized residuals:
##      Min      Q1      Med      Q3      Max
## -6.08634239 -0.56184798 -0.02356306  0.57468340  3.55358730
##
## Residual standard error: 3.665685
## Degrees of freedom: 3000 total; 2993 residual

getVarCov(m6.gls)

## Marginal variance covariance matrix
##      [,1] [,2] [,3]
## [1,] 13.4370  7.9019  3.6955
## [2,]  7.9019 13.4370  7.9019
## [3,]  3.6955  7.9019 13.4370
## Standard Deviations: 3.6657 3.6657 3.6657

```

```
AIC_MARGINAL_MODEL_9 <- extractAIC(m6.gls)[2]
```

```
m6.gls.heter <- gls(change~trt*ftime + y0,correlation=corARMA(c(0.8,0.8),form=~1|subject,p=1,q=1),weights=varIdent(form=~1|ftime),method='ML',data=data) # ARMA(1,1), heteroscedastic  
summary(m6.gls.heter) # AIC = 14150.88
```

```
## Generalized least squares fit by maximum likelihood
```

```
## Model: change ~ trt * ftime + y0
```

```
## Data: data
```

```
## AIC BIC logLik
```

```
## 14150.88 14222.96 -7063.441
```

```
##
```

```
## Correlation Structure: ARMA(1,1)
```

```
## Formula: ~1 | subject
```

```
## Parameter estimate(s):
```

```
## Phi1 Theta1
```

```
## 0.4756221 0.4055784
```

```
## Variance function:
```

```
## Structure: Different standard deviations per stratum
```

```
## Formula: ~1 | ftime
```

```
## Parameter estimates:
```

```
## 1 2 3
```

```
## 1.000000 1.663796 2.924310
```

```
##
```

```
## Coefficients:
```

```
## Value Std.Error t-value p-value
```

```
## (Intercept) 2.150141 0.3103592 6.927912 0.0000
```

```
## trt 0.941094 0.1182779 7.956637 0.0000
```

```
## ftime2 0.084476 0.1028051 0.821714 0.4113
```

```
## ftime3 0.125314 0.2314819 0.541355 0.5883
```

```
## y0 -0.000979 0.0060206 -0.162567 0.8709
```

```
## trt:ftime2 2.995749 0.1453884 20.605150 0.0000
```

```
## trt:ftime3 6.854956 0.3273648 20.939807 0.0000
```

```
##
```

```
## Correlation:
```

```
## (Intr) trt ftime2 ftime3 y0 trt:f2
```

```
## trt -0.166
```

```
## ftime2 0.028 -0.074
```

```
## ftime3 -0.006 0.015 0.654
```

```
## y0 -0.963 -0.025 0.000 0.000
```

```
## trt:ftime2 -0.020 0.104 -0.707 -0.463 0.000
```

```
## trt:ftime3 0.004 -0.021 -0.463 -0.707 0.000 0.654
```

```
##
```

```
## Standardized residuals:
```

```
## Min Q1 Med Q3 Max
```

```
## -4.10658387 -0.66353309 -0.03410693 0.70405431 3.85760551
##
## Residual standard error: 1.867354
## Degrees of freedom: 3000 total; 2993 residual

getVarCov(m6.gls.heter)

## Marginal variance covariance matrix
##      [,1] [,2] [,3]
## [1,] 3.4870 3.9338 3.2885
## [2,] 3.9338 9.6528 11.5040
## [3,] 3.2885 11.5040 29.8190
## Standard Deviations: 1.8674 3.1069 5.4607

AIC_MARGINAL_MODEL_10 <- extractAIC(m6.gls.heter)[2]

m7.gls.heter <- gls(change~trt*ftime + y0,correlation=corARMA(c(0.2,0.2
,0.5,0.5),form=~1|subject,p=2,q=2),weights=varIdent(form=~1|ftime),meth
od='ML',data=data) # ARMA(2,2), heteroscedastic
summary(m7.gls.heter) # AIC = 14150.88

## Generalized least squares fit by maximum likelihood
## Model: change ~ trt * ftime + y0
## Data: data
##      AIC      BIC    logLik
## 14154.88 14238.97 -7063.441
##
## Correlation Structure: ARMA(2,2)
## Formula: ~1 | subject
## Parameter estimate(s):
##      Phi1      Phi2      Theta1      Theta2
## 0.20863861 -0.03794702 0.64678813 0.46007575
## Variance function:
## Structure: Different standard deviations per stratum
## Formula: ~1 | ftime
## Parameter estimates:
##      1      2      3
## 1.000000 1.663778 2.924297
##
## Coefficients:
##      Value Std.Error t-value p-value
## (Intercept) 2.150151 0.3103594 6.927939 0.0000
## trt          0.941094 0.1182777 7.956648 0.0000
## ftime2       0.084476 0.1028046 0.821718 0.4113
## ftime3       0.125314 0.2314816 0.541355 0.5883
## y0          -0.000979 0.0060206 -0.162599 0.8708
## trt:ftime2   2.995749 0.1453877 20.605251 0.0000
## trt:ftime3   6.854956 0.3273644 20.939830 0.0000
```

```

##
## Correlation:
##      (Intr) trt      ftime2 ftime3 y0      trt:f2
## trt      -0.166
## ftime2    0.028 -0.074
## ftime3   -0.006  0.015  0.654
## y0       -0.963 -0.025  0.000  0.000
## trt:ftime2 -0.020  0.104 -0.707 -0.463  0.000
## trt:ftime3  0.004 -0.021 -0.463 -0.707  0.000  0.654
##
## Standardized residuals:
##      Min      Q1      Med      Q3      Max
## -4.10660687 -0.66353283 -0.03410712  0.70405807  3.85760990
##
## Residual standard error: 1.867351
## Degrees of freedom: 3000 total; 2993 residual

getVarCov(m7.gls.heter)

## Marginal variance covariance matrix
##      [,1] [,2] [,3]
## [1,] 3.4870 3.9338 3.2884
## [2,] 3.9338 9.6526 11.5030
## [3,] 3.2884 11.5030 29.8190
## Standard Deviations: 1.8674 3.1069 5.4607

AIC_MARGINAL_MODEL_11 <- extractAIC(m7.gls.heter)[2]

m8.gls <- gls(change~trt*ftime + y0,correlation=corExp(form=~1|subject)
,method='ML',data=data)
summary(m8.gls) # AIC = 15480.07

## Generalized least squares fit by maximum likelihood
## Model: change ~ trt * ftime + y0
## Data: data
##      AIC      BIC    logLik
## 15480.07 15534.13 -7731.037
##
## Correlation Structure: Exponential spatial correlation
## Formula: ~1 | subject
## Parameter estimate(s):
## range
## 1.837634
##
## Coefficients:
##      Value Std.Error   t-value p-value
## (Intercept)  2.533363 0.5053338   5.013248  0.0000

```

```

## trt          0.944940 0.2312177  4.086798  0.0000
## ftime2       0.084476 0.1497571  0.564089  0.5727
## ftime3       0.125314 0.1882607  0.665639  0.5057
## y0          -0.008698 0.0096319 -0.903057  0.3666
## trt:ftime2   2.995749 0.2117885 14.145004  0.0000
## trt:ftime3   6.854956 0.2662409 25.747196  0.0000
##
## Correlation:
##          (Intr) trt      ftime2 ftime3 y0      trt:f2
## trt      -0.209
## ftime2   -0.148  0.324
## ftime3   -0.186  0.407  0.629
## y0       -0.946 -0.021  0.000  0.000
## trt:ftime2 0.105 -0.458 -0.707 -0.444  0.000
## trt:ftime3 0.132 -0.576 -0.444 -0.707  0.000  0.629
##
## Standardized residuals:
##          Min          Q1          Med          Q3          Max
## -6.10921710 -0.56252911 -0.02449452  0.57718053  3.56838840
##
## Residual standard error: 3.650819
## Degrees of freedom: 3000 total; 2993 residual

getVarCov(m8.gls)

## Marginal variance covariance matrix
##          [,1]      [,2]      [,3]
## [1,] 13.3280  7.7348  4.4886
## [2,]  7.7348 13.3280  7.7348
## [3,]  4.4886  7.7348 13.3280
## Standard Deviations: 3.6508 3.6508 3.6508

AIC_MARGINAL_MODEL_12 <- extractAIC(m8.gls)[2]

# We observe that the model with the lowest AIC value is the Unrestricted Cov matrix with heteroscedastic errors!

# Create a matrix with the AIC of the marginal models run above
mat <- matrix(c(AIC_MARGINAL_MODEL_1,AIC_MARGINAL_MODEL_2,AIC_MARGINAL_MODEL_3,AIC_MARGINAL_MODEL_4,AIC_MARGINAL_MODEL_5,AIC_MARGINAL_MODEL_6,AIC_MARGINAL_MODEL_7,AIC_MARGINAL_MODEL_8,AIC_MARGINAL_MODEL_9,AIC_MARGINAL_MODEL_10,AIC_MARGINAL_MODEL_11,AIC_MARGINAL_MODEL_12),nrow=12,ncol=1,byrow=T)
mat

##          [,1]
## [1,] 17019.90
## [2,] 15705.87
## [3,] 15480.07

```



```
## [4,] 14213.04
## [5,] 14884.43
## [6,] 14006.02
## [7,] 15568.97
## [8,] 15467.21
## [9,] 15467.21
## [10,] 14150.88
## [11,] 14154.88
## [12,] 15480.07

rownames(mat) <- c('Independent + Homoscedastic', 'Compound Symmetry + H
omoscedastic', 'AR(1) + Homoscedastic', 'AR(1) + Heteroscedastic',
'Unstructured + Homoscedastic', 'Unstructured + Heteroscedastic', 'MA(1)
+ Homoscedastic', 'MA(2) + Homoscedastic', 'ARMA(1,1) + Homoscedastic',
'ARMA(1,1) + Heteroscedastic', 'ARMA(2,2) + Heteroscedastic', 'Exponentia
l + Homoscedastic')

colnames(mat) <- 'AIC'
mat

##                               AIC
## Independent + Homoscedastic    17019.90
## Compound Symmetry + Homoscedastic 15705.87
## AR(1) + Homoscedastic          15480.07
## AR(1) + Heteroscedastic        14213.04
## Unstructured + Homoscedastic    14884.43
## Unstructured + Heteroscedastic  14006.02
## MA(1) + Homoscedastic           15568.97
## MA(2) + Homoscedastic           15467.21
## ARMA(1,1) + Homoscedastic       15467.21
## ARMA(1,1) + Heteroscedastic     14150.88
## ARMA(2,2) + Heteroscedastic     14154.88
## Exponential + Homoscedastic     15480.07
```

Create a matrix with the AIC of the marginal models run above

```
mat <- matrix(c(AIC_MARGINAL_MODEL_1,AIC_MARGINAL_MODEL_2,AIC_MARGINAL_
MODEL_3,AIC_MARGINAL_MODEL_4,AIC_MARGINAL_MODEL_5,AIC_MARGINAL_MODEL_6,
AIC_MARGINAL_MODEL_7,AIC_MARGINAL_MODEL_8,
AIC_MARGINAL_MODEL_9,AIC_MARGINAL_MODEL_10,AIC_MARGINAL_MODEL_11,AIC_MA
RGINAL_MODEL_12),nrow=12,ncol=1,byrow=T)
mat

##           [,1]
## [1,] 17019.90
## [2,] 15705.87
## [3,] 15480.07
## [4,] 14213.04
## [5,] 14884.43
## [6,] 14006.02
## [7,] 15568.97
```

```
## [8,] 15467.21
## [9,] 15467.21
## [10,] 14150.88
## [11,] 14154.88
## [12,] 15480.07

rownames(mat) <- c('Independent + Homoscedastic', 'Compound Symmetry + H
omoscedastic', 'AR(1) + Homoscedastic', 'AR(1) + Heteroscedastic',
'Unstructured + Homoscedastic', 'Unstructured + Heteroscedastic', 'MA(1)
+ Homoscedastic', 'MA(2) + Homoscedastic', 'ARMA(1,1) + Homoscedastic',
'ARMA(1,1) + Heteroscedastic', 'ARMA(2,2) + Heteroscedastic', 'Exponentia
l + Homoscedastic')

colnames(mat) <- 'AIC'
mat

##                               AIC
## Independent + Homoscedastic      17019.90
## Compound Symmetry + Homoscedastic 15705.87
## AR(1) + Homoscedastic            15480.07
## AR(1) + Heteroscedastic          14213.04
## Unstructured + Homoscedastic      14884.43
## Unstructured + Heteroscedastic     14006.02
## MA(1) + Homoscedastic             15568.97
## MA(2) + Homoscedastic             15467.21
## ARMA(1,1) + Homoscedastic         15467.21
## ARMA(1,1) + Heteroscedastic       14150.88
## ARMA(2,2) + Heteroscedastic       14154.88
## Exponential + Homoscedastic       15480.07

# Multiple comparisons for the nested models
anova(m1.gls,m2.gls,m2.gls.heter,m3.gls,m3.gls.heter,m4.gls,m5.gls,m6.g
ls,m6.gls.heter,m7.gls.heter,m8.gls) # Only for the nested models

##           Model df      AIC      BIC    logLik      Test  L.Ratio
p-value
## m1.gls           1  9 15705.87 15759.93 -7843.936
## m2.gls           2  9 15480.07 15534.13 -7731.037
## m2.gls.heter     3 11 14213.03 14279.10 -7095.518    2 vs 3 1271.0390
<.0001
## m3.gls           4 11 14884.43 14950.50 -7431.213
## m3.gls.heter     5 13 14006.02 14084.10 -6990.009    4 vs 5  882.4076
<.0001
## m4.gls           6  9 15568.97 15623.03 -7775.487    5 vs 6 1570.9562
<.0001
## m5.gls           7 10 15467.21 15527.27 -7723.603    6 vs 7  103.7681
<.0001
## m6.gls           8 10 15467.21 15527.27 -7723.603
## m6.gls.heter     9 12 14150.88 14222.96 -7063.441    8 vs 9 1320.3237
<.0001
## m7.gls.heter    10 14 14154.88 14238.97 -7063.441    9 vs 10    0.0000
```

```

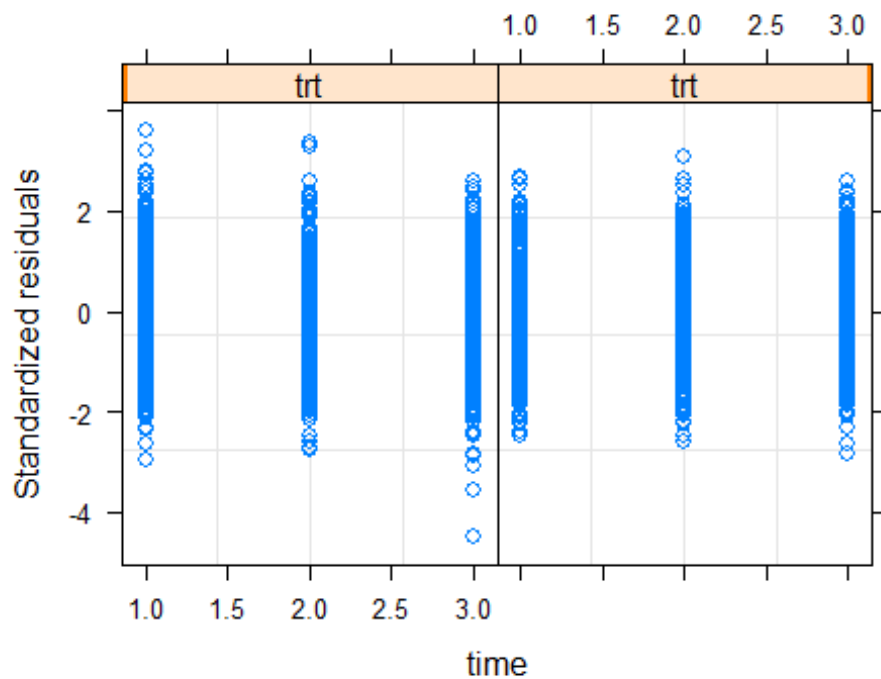
1
## m8.gls          11  9 15480.07 15534.13 -7731.037 10 vs 11 1335.1914
<.0001

intervals(m3.gls.heter) # 95% CI's of the coeffs of the best model base
d on AIC

## Approximate 95% confidence intervals
##
## Coefficients:
##           lower      est.      upper
## (Intercept) 1.41998099 2.047604077 2.67522716
## trt          0.69033886 0.940065264 1.18979167
## ftime2       -0.08333536 0.084476397 0.25228816
## ftime3       -0.29142765 0.125313758 0.54205516
## y0           -0.01104534 0.001086692 0.01321873
## trt:ftime2    2.75842756 2.995749224 3.23307089
## trt:ftime3    6.26559431 6.854955661 7.44431701
##
## Correlation structure:
##           lower      est.      upper
## cor(1,2) 0.7808297 0.8039371 0.8248480
## cor(1,3) 0.2704132 0.3268949 0.3811342
## cor(2,3) 0.4896404 0.5353195 0.5780682
##
## Variance function:
##           lower      est.      upper
## 2 1.489601 1.545533 1.603566
## 3 2.348500 2.490193 2.640434
##
## Residual standard error:
##           lower      est.      upper
## 1.924527 2.010834 2.101012

plot(m3.gls.heter,resid(.,type="p")~time|trt) # pearson residual plot v
s time for each treatment

```

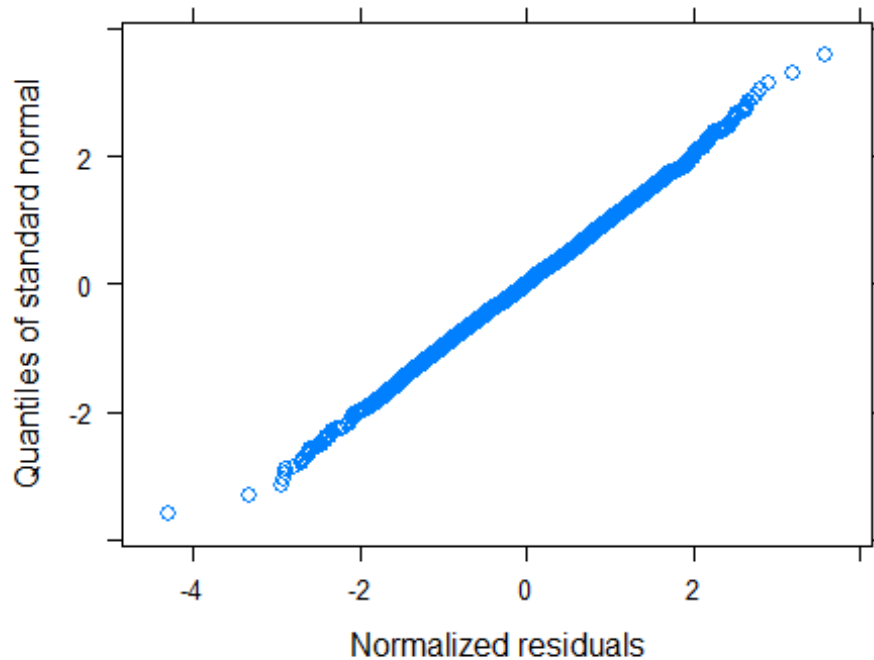


The heights of the bars seem to be different, indicating that there's heteroscedasticity

```
head(predict(m3.gls.heter))
```

```
##          1          2          3          4          5          6
## 3.028126 6.108351 10.008395 3.044181 6.124406 10.024450
```

```
qqnorm(m3.gls.heter,~resid(.,type="n")) # resid type="response", "pearson", "normalized"
```



Fitting random effects (or subject specific models) models

```
m1.lme <- lme(change~trt*ftime + y0,random=~1|subject,method="ML",data=
data) # one random intercept (for each subject)
summary(m1.lme) # AIC = 15705.87
```

```
## Linear mixed-effects model fit by maximum likelihood
##   Data: data
##       AIC      BIC    logLik
## 15705.87 15759.93 -7843.936
##
## Random effects:
## Formula: ~1 | subject
##      (Intercept) Residual
## StdDev:      2.353214 2.715991
##
## Fixed effects:  change ~ trt * ftime + y0
##              Value Std.Error   DF   t-value p-value
## (Intercept)  2.586481 0.4857190 1996   5.325056  0.0000
## trt          0.945473 0.2275936   997   4.154217  0.0000
## ftime2       0.084476 0.1719751 1996   0.491213  0.6233
## ftime3       0.125314 0.1719751 1996   0.728674  0.4663
## y0          -0.009768 0.0092316   997  -1.058116  0.2903
## trt:ftime2   2.995749 0.2432095 1996  12.317567  0.0000
## trt:ftime3   6.854956 0.2432095 1996  28.185395  0.0000
## Correlation:
##              (Intr) trt    ftime2 ftime3 y0      trt:f2
```

```

## trt          -0.215
## ftime2       -0.177  0.378
## ftime3       -0.177  0.378  0.500
## y0           -0.944 -0.020  0.000  0.000
## trt:ftime2    0.125 -0.534 -0.707 -0.354  0.000
## trt:ftime3    0.125 -0.534 -0.354 -0.707  0.000  0.500
##
## Standardized Within-Group Residuals:
##           Min           Q1           Med           Q3           Max
## -5.626034378 -0.506636443 -0.004832299  0.490722140  3.157681668
##
## Number of Observations: 3000
## Number of Groups: 1000

AIC_RE_MODEL_1 <- extractAIC(m1.lme)[2]

m2.lme <- lme(change~trt*ftime + y0,random=~1|trt,method="ML",data=data
) # one random intercept (for each treatment group)
summary(m2.lme) # AIC = 16206.62

## Warning in pt(-abs(tVal), fDF): NaNs produced

## Linear mixed-effects model fit by maximum likelihood
##   Data: data
##       AIC      BIC    logLik
##  16206.62 16260.68 -8094.309
##
## Random effects:
## Formula: ~1 | trt
##      (Intercept) Residual
## StdDev: 0.000102844 3.593636
##
## Fixed effects:  change ~ trt * ftime + y0
##              Value Std.Error   DF   t-value p-value
## (Intercept)  2.586481 0.3727685 2993   6.938572  0.0000
## trt          0.945473 0.2275722    0   4.154608    NaN
## ftime2       0.084476 0.2275471 2993   0.371248  0.7105
## ftime3       0.125314 0.2275471 2993   0.550716  0.5819
## y0           -0.009768 0.0067733 2993  -1.442146  0.1494
## trt:ftime2    2.995749 0.3218003 2993   9.309344  0.0000
## trt:ftime3    6.854956 0.3218003 2993  21.301896  0.0000
## Correlation:
##      (Intr) trt      ftime2 ftime3 y0      trt:f2
## trt          -0.292
## ftime2       -0.305  0.500
## ftime3       -0.305  0.500  0.500
## y0           -0.902 -0.015  0.000  0.000
## trt:ftime2    0.216 -0.707 -0.707 -0.354  0.000

```

```
## trt:ftime3  0.216 -0.707 -0.354 -0.707  0.000  0.500
##
## Standardized Within-Group Residuals:
##           Min           Q1           Med           Q3           Max
## -6.20175099 -0.57355827 -0.02488356  0.58786327  3.62598122
##
## Number of Observations: 3000
## Number of Groups: 2

AIC_RE_MODEL_2 <- extractAIC(m2.lme)[2]

m3.lme <- lme(change~trt*ftime + y0,random=~trt|subject,method="ML",data=
a=data) # one random intercept and one random slope for each subject (different
for each treatment arm)
summary(m3.lme) # AIC = 15709.73

## Linear mixed-effects model fit by maximum likelihood
##   Data: data
##       AIC      BIC    logLik
## 15709.73 15775.8 -7843.866
##
## Random effects:
## Formula: ~trt | subject
## Structure: General positive-definite, Log-Cholesky parametrization
##           StdDev   Corr
## (Intercept) 2.381506 (Intr)
## trt          3.112858 -0.672
## Residual    2.715993
##
## Fixed effects:  change ~ trt * ftime + y0
##              Value Std.Error   DF   t-value p-value
## (Intercept)  2.583617 0.4858234 1996   5.318018  0.0000
## trt          0.945445 0.2275934  997   4.154094  0.0000
## ftime2       0.084476 0.1719752 1996   0.491213  0.6233
## ftime3       0.125314 0.1719752 1996   0.728673  0.4663
## y0          -0.009710 0.0092279  997  -1.052286  0.2929
## trt:ftime2   2.995749 0.2432097 1996  12.317557  0.0000
## trt:ftime3   6.854956 0.2432097 1996  28.185372  0.0000
## Correlation:
##           (Intr) trt    ftime2 ftime3 y0      trt:f2
## trt        -0.218
## ftime2     -0.177  0.378
## ftime3     -0.177  0.378  0.500
## y0         -0.943 -0.020  0.000  0.000
## trt:ftime2  0.125 -0.534 -0.707 -0.354  0.000
## trt:ftime3  0.125 -0.534 -0.354 -0.707  0.000  0.500
##
## Standardized Within-Group Residuals:
```

```

##           Min           Q1           Med           Q3           Max
## -5.607259187 -0.506569078 -0.003151258  0.490095470  3.163837553
##
## Number of Observations: 3000
## Number of Groups: 1000

AIC_RE_MODEL_3 <- extractAIC(m3.lme)[2]

m4.lme <- lme(change~trt*ftime + y0,random=~time|subject,method="ML",da
ta=data) # one random intercept and one random slope for each subject (
different for each time point)
summary(m4.lme) # AIC = 14930.64

## Linear mixed-effects model fit by maximum likelihood
##   Data: data
##       AIC      BIC    logLik
## 14930.64 14996.71 -7454.322
##
## Random effects:
## Formula: ~time | subject
## Structure: General positive-definite, Log-Cholesky parametrization
##           StdDev   Corr
## (Intercept) 2.487531 (Intr)
## time         1.972484 -0.76
## Residual     1.867049
##
## Fixed effects: change ~ trt * ftime + y0
##           Value Std.Error   DF   t-value p-value
## (Intercept)  2.402418 0.3817530 1996   6.293120  0.0000
## trt          0.943626 0.1565203   997   6.028779  0.0000
## ftime2       0.084476 0.1475659 1996   0.572466  0.5671
## ftime3       0.125314 0.2125428 1996   0.589593  0.5555
## y0          -0.006060 0.0073597   997  -0.823462  0.4104
## trt:ftime2   2.995749 0.2086897 1996  14.355043  0.0000
## trt:ftime3   6.854956 0.3005810 1996  22.805688  0.0000
## Correlation:
##           (Intr) trt    ftime2 ftime3 y0      trt:f2
## trt        -0.182
## ftime2     -0.118  0.288
## ftime3     -0.078  0.191  0.720
## y0         -0.957 -0.023  0.000  0.000
## trt:ftime2  0.084 -0.408 -0.707 -0.509  0.000
## trt:ftime3  0.055 -0.269 -0.509 -0.707  0.000  0.720
##
## Standardized Within-Group Residuals:
##           Min           Q1           Med           Q3           Max
## -2.998306619 -0.488825805 -0.002147128  0.479480023  3.105882781
##

```



```

## Number of Observations: 3000
## Number of Groups: 1000

# Note that stdDev(time) = 1.972, which is not close to zero, thus indi
cating that random slope is meaningful!
# Default is the Unrestricted Cov structure
AIC_RE_MODEL_4 <- extractAIC(m4.lme)[2]

m5.lme <- lme(change~trt*ftime + y0,random=list(subject=pdDiag(~1+time)
),method="ML",data=data) # covmat random effects, diagonal
summary(m5.lme) # AIC = 15047.3

## Linear mixed-effects model fit by maximum likelihood
##   Data: data
##       AIC      BIC    logLik
## 15047.3 15107.37 -7513.651
##
## Random effects:
## Formula: ~1 + time | subject
## Structure: Diagonal
##           (Intercept)      time Residual
## StdDev: 0.0004066823 1.317816 2.193552
##
## Fixed effects:  change ~ trt * ftime + y0
##               Value Std.Error   DF   t-value p-value
## (Intercept)  2.371950 0.4015228 1996   5.907385  0.0000
## trt          0.943320 0.1620784  997   5.820149  0.0000
## ftime2       0.084476 0.1509075 1996   0.559789  0.5757
## ftime3       0.125314 0.1822561 1996   0.687570  0.4918
## y0          -0.005447 0.0077517  997  -0.702644  0.4824
## trt:ftime2   2.995749 0.2134155 1996 14.037169  0.0000
## trt:ftime3   6.854956 0.2577490 1996 26.595466  0.0000
## Correlation:
##           (Intr) trt      ftime2 ftime3 y0      trt:f2
## trt          -0.179
## ftime2       -0.102  0.252
## ftime3       -0.037  0.091  0.604
## y0           -0.958 -0.024  0.000  0.000
## trt:ftime2   0.072 -0.356 -0.707 -0.427  0.000
## trt:ftime3   0.026 -0.128 -0.427 -0.707  0.000  0.604
##
## Standardized Within-Group Residuals:
##           Min           Q1           Med           Q3           Max
## -3.615476388 -0.586136925 -0.008293428  0.583651454  2.849813852
##
## Number of Observations: 3000
## Number of Groups: 1000

```

```

AIC_RE_MODEL_5 <- extractAIC(m5.lme)[2]

m6.lme <- lme(change~trt*ftime + y0,random=list(subject=pdIdent(~time))
,method="ML",data=data) # covmat random effects, identity
summary(m6.lme) # AIC = 15088.33

## Linear mixed-effects model fit by maximum likelihood
##   Data: data
##       AIC      BIC    logLik
## 15088.33 15142.39 -7535.167
##
## Random effects:
## Formula: ~time | subject
## Structure: Multiple of an Identity
##           (Intercept)   time Residual
## StdDev:      1.2286 1.2286 2.145314
##
## Fixed effects:  change ~ trt * ftime + y0
##               Value Std.Error   DF   t-value p-value
## (Intercept)  2.383776 0.4370232 1996   5.454576  0.0000
## trt          0.943439 0.1748546   997   5.395562  0.0000
## ftime2       0.084476 0.1465555 1996   0.576412  0.5644
## ftime3       0.125314 0.1748040 1996   0.716882  0.4735
## y0          -0.005685 0.0084437   997  -0.673279  0.5009
## trt:ftime2   2.995749 0.2072608 1996 14.454009  0.0000
## trt:ftime3   6.854956 0.2472102 1996 27.729260  0.0000
## Correlation:
##           (Intr) trt    ftime2 ftime3 y0      trt:f2
## trt        -0.177
## ftime2     -0.097  0.242
## ftime3     -0.042  0.104  0.596
## y0         -0.959 -0.024  0.000  0.000
## trt:ftime2  0.068 -0.342 -0.707 -0.422  0.000
## trt:ftime3  0.029 -0.147 -0.422 -0.707  0.000  0.596
##
## Standardized Within-Group Residuals:
##           Min           Q1           Med           Q3           Ma
x
## -3.821009e+00 -5.527777e-01 -1.804182e-05  5.549804e-01  2.720726e+0
0
##
## Number of Observations: 3000
## Number of Groups: 1000

AIC_RE_MODEL_6 <- extractAIC(m6.lme)[2]

```

```

m7.lme <-ortho.lme1.2<-lme(change~trt*ftime + y0,random=~1|subject,weights=varIdent(form=~1|trt),method="ML",data=data) # one random intercept
for each subject, two residual error variances one for each treatment group
summary(m7.lme) # AIC = 15706.05

## Linear mixed-effects model fit by maximum likelihood
##   Data: data
##       AIC      BIC    logLik
## 15706.05 15766.12 -7843.027
##
## Random effects:
## Formula: ~1 | subject
##      (Intercept) Residual
## StdDev:      2.353016 2.658888
##
## Variance function:
## Structure: Different standard deviations per stratum
## Formula: ~1 | trt
## Parameter estimates:
##      1      0
## 1.000000 1.042561
## Fixed effects: change ~ trt * ftime + y0
##              Value Std.Error   DF   t-value p-value
## (Intercept)  2.584291 0.4862098 1996   5.315178  0.0000
## trt          0.945451 0.2275887  997   4.154211  0.0000
## ftime2       0.084476 0.1755248 1996   0.481279  0.6304
## ftime3       0.125314 0.1755248 1996   0.713937  0.4753
## y0          -0.009724 0.0092286  997  -1.053684  0.2923
## trt:ftime2   2.995749 0.2432156 1996  12.317255  0.0000
## trt:ftime3   6.854956 0.2432156 1996  28.184682  0.0000
## Correlation:
##      (Intr) trt      ftime2 ftime3 y0      trt:f2
## trt      -0.220
## ftime2    -0.181  0.386
## ftime3    -0.181  0.386  0.500
## y0        -0.942 -0.020  0.000  0.000
## trt:ftime2 0.130 -0.534 -0.722 -0.361  0.000
## trt:ftime3 0.130 -0.534 -0.361 -0.722  0.000  0.500
##
## Standardized Within-Group Residuals:
##      Min      Q1      Med      Q3      Max
## -5.544473079 -0.505292486 -0.005590577  0.486611217  3.214513319
##
## Number of Observations: 3000
## Number of Groups: 1000

AIC_RE_MODEL_7 <- extractAIC(m7.lme)[2]

```

```

m8.lme.cor <- lme(change~trt*ftime + y0,random=~1|subject,correlation=c
orCompSymm(value=0.3,form=~1|subject),method="ML",data=data) # compound
symmetry correlation matirx plus a random intercept (for each subject)
summary(m8.lme.cor) # AIC = 15707.87

## Linear mixed-effects model fit by maximum likelihood
##   Data: data
##           AIC      BIC    logLik
##  15707.87 15767.94 -7843.936
##
## Random effects:
## Formula: ~1 | subject
##      (Intercept) Residual
## StdDev:      1.865277 3.071639
##
## Correlation Structure: Compound symmetry
## Formula: ~1 | subject
## Parameter estimate(s):
##      Rho
## 0.218163
## Fixed effects:  change ~ trt * ftime + y0
##              Value Std.Error   DF   t-value p-value
## (Intercept)  2.586481 0.4857191 1996   5.325056  0.0000
## trt          0.945473 0.2275936   997   4.154217  0.0000
## ftime2       0.084476 0.1719751 1996   0.491213  0.6233
## ftime3       0.125314 0.1719751 1996   0.728674  0.4663
## y0          -0.009768 0.0092316   997  -1.058116  0.2903
## trt:ftime2   2.995749 0.2432095 1996  12.317568  0.0000
## trt:ftime3   6.854956 0.2432095 1996  28.185397  0.0000
## Correlation:
##      (Intr) trt    ftime2 ftime3 y0      trt:f2
## trt      -0.215
## ftime2   -0.177  0.378
## ftime3   -0.177  0.378  0.500
## y0       -0.944 -0.020  0.000  0.000
## trt:ftime2 0.125 -0.534 -0.707 -0.354  0.000
## trt:ftime3 0.125 -0.534 -0.354 -0.707  0.000  0.500
##
## Standardized Within-Group Residuals:
##           Min           Q1           Med           Q3           Max
## -5.822506485 -0.485457211 -0.005932086  0.495095091  3.265219990
##
## Number of Observations: 3000
## Number of Groups: 1000

AIC_RE_MODEL_8 <- extractAIC(m8.lme.cor)[2]

```

```

m9.lme.cor <- lme(change~trt*ftime + y0,random=~1|subject,correlation=c
orSymm(form=~1|subject),method="ML",data=data) # Unstructured Cov matrix
plus a random intercept (for each subject)
summary(m9.lme.cor) # AIC = 14886.43

## Linear mixed-effects model fit by maximum likelihood
##   Data: data
##       AIC      BIC    logLik
##  14886.43 14958.5 -7431.213
##
## Random effects:
## Formula: ~1 | subject
##      (Intercept) Residual
## StdDev:      2.66815 2.791939
##
## Correlation Structure: General
## Formula: ~1 | subject
## Parameter estimate(s):
## Correlation:
##   1      2
## 2  0.778
## 3 -0.596 -0.249
## Fixed effects:  change ~ trt * ftime + y0
##               Value Std.Error   DF   t-value p-value
## (Intercept)  2.385062 0.5058498 1996   4.714961  0.0000
## trt          0.943452 0.2445773   997   3.857479  0.0001
## ftime2       0.084476 0.0833035 1996   1.014080  0.3107
## ftime3       0.125314 0.2233510 1996   0.561062  0.5748
## y0          -0.005711 0.0095757   997  -0.596386  0.5511
## trt:ftime2   2.995749 0.1178090 1996  25.428872  0.0000
## trt:ftime3   6.854956 0.3158661 1996  21.702096  0.0000
## Correlation:
##           (Intr) trt    ftime2 ftime3 y0      trt:f2
## trt       -0.223
## ftime2    -0.082  0.170
## ftime3    -0.221  0.457  0.478
## y0        -0.940 -0.020  0.000  0.000
## trt:ftime2 0.058 -0.241 -0.707 -0.338  0.000
## trt:ftime3 0.156 -0.646 -0.338 -0.707  0.000  0.478
##
## Standardized Within-Group Residuals:
##           Min           Q1           Med           Q3           Max
## -4.22272151 -0.57960701 -0.01671123  0.58786503  2.76875538
##
## Number of Observations: 3000
## Number of Groups: 1000

AIC_RE_MODEL_9 <- extractAIC(m9.lme.cor)[2]

```

```

m10.lme.cor.heter <- lme(change~trt*ftime + y0,random=~1|subject,correlation=corSymm(form=~1|subject),weights=varIdent(form=~1|trt), method="ML",data=data) # Unstructured Cov matrix plus a random intercept (for each subject) and two residual error variances (one for each treatment)
summary(m10.lme.cor.heter) # AIC = 14885.31

## Linear mixed-effects model fit by maximum likelihood
##   Data: data
##           AIC      BIC    logLik
## 14885.31 14963.39 -7429.654
##
## Random effects:
## Formula: ~1 | subject
##      (Intercept) Residual
## StdDev:      2.6866 2.697094
##
## Correlation Structure: General
## Formula: ~1 | subject
## Parameter estimate(s):
## Correlation:
##   1      2
## 2  0.775
## 3 -0.617 -0.265
## Variance function:
## Structure: Different standard deviations per stratum
## Formula: ~1 | trt
## Parameter estimates:
##      1      0
## 1.000000 1.057078
## Fixed effects:  change ~ trt * ftime + y0
##              Value Std.Error   DF   t-value p-value
## (Intercept)  2.384101 0.5065908 1996   4.706167  0.0000
## trt          0.943442 0.2446202   997   3.856763  0.0001
## ftime2       0.084476 0.0855586 1996   0.987351  0.3236
## ftime3       0.125314 0.2295665 1996   0.545871  0.5852
## y0          -0.005691 0.0095733   997  -0.594518  0.5523
## trt:ftime2   2.995749 0.1177768 1996  25.435827  0.0000
## trt:ftime3   6.854956 0.3160126 1996  21.692036  0.0000
## Correlation:
##      (Intr) trt      ftime2 ftime3 y0      trt:f2
## trt      -0.230
## ftime2    -0.084  0.175
## ftime3    -0.227  0.469  0.478
## y0        -0.938 -0.019  0.000  0.000
## trt:ftime2 0.061 -0.241 -0.726 -0.347  0.000
## trt:ftime3 0.165 -0.646 -0.347 -0.726  0.000  0.478
##
## Standardized Within-Group Residuals:

```

```

##           Min           Q1           Med           Q3           Max
## -4.11622279 -0.58249183 -0.01331649  0.59262177  2.82751485
##
## Number of Observations: 3000
## Number of Groups: 1000

# From the 10 models constructed using lme command, this one has the Lo
west AIC value
AIC_RE_MODEL_10 <- extractAIC(m10.lme.cor.heter)[2]

# Create a matrix with the AIC of the random effects models run above
mat2 <- matrix(c(AIC_RE_MODEL_1,AIC_RE_MODEL_2,AIC_RE_MODEL_3,AIC_RE_MODEL_4,AIC_RE_MODEL_5,
                 AIC_RE_MODEL_6,AIC_RE_MODEL_7,AIC_RE_MODEL_8,
                 AIC_RE_MODEL_9,AIC_RE_MODEL_10),nrow=10,ncol=1,byrow=T
)
mat2

##           [,1]
## [1,] 15705.87
## [2,] 16206.62
## [3,] 15709.73
## [4,] 14930.64
## [5,] 15047.30
## [6,] 15088.33
## [7,] 15706.05
## [8,] 15707.87
## [9,] 14886.43
## [10,] 14885.31

rownames(mat2) <- c('Random Intercept (for each subject)', 'Random Intercept (for each treatment)',
'Random Intercept + Random Slope for each subject (different for each treatment arm)',
'Random Intercept + Random Slope for each subject (different for each time point)', 'Covmat Random Effects, Diagonal',
'Covmat Random Effects, Identity',
'Random Intercept for each subject + Two Residual Error Variances (one for each treatment group)',
'Compound Symmetry Correlation Matrix + Random Intercept (for each subject)',
'Unstructured Cov Matrix + Random Intercept (for each subject)',
'Unstructured Cov Matrix + Random Intercept (for each subject) + Two Residual Error Variances (one for each treatment)')

colnames(mat2) <- 'AIC'
mat2

```

```

##
AIC
## Random Intercept (for each subject)
15705.87
## Random Intercept (for each treatment)
16206.62
## Random Intercept + Random Slope for each subject (different for each
treatment arm) 15709.73
## Random Intercept + Random Slope for each subject (different for each
time point) 14930.64
## Covmat Random Effects, Diagonal
15047.30
## Covmat Random Effects, Identity
15088.33
## Random Intercept for each subject + Two Residual Error Variances (on
e for each treatment group) 15706.05
## Compound Symmetry Correlation Matrix + Random Intercept (for each su
bject) 15707.87
## Unstructured Cov Matrix + Random Intercept (for each subject)
14886.43
## Unstructured Cov Matrix + Random Intercept (for each subject) + Two R
esidual Error Variances (one for each treatment) 14885.31

# More output based on lme
VarCorr(m10.lme.cor.heter) # estimated variance (correlations) of rand
om effects

## subject = pdLogChol(1)
## Variance StdDev
## (Intercept) 7.217818 2.686600
## Residual 7.274314 2.697094

tmp0 <- summary(m10.lme.cor.heter)
names(tmp0) # provides useful functions

## [1] "modelStruct" "dims" "contrasts" "coefficients" "va
rFix"
## [6] "sigma" "apVar" "logLik" "numIter" "gr
oups"
## [11] "call" "terms" "method" "fitted" "re
siduals"
## [16] "fixDF" "na.action" "data" "corFixed" "tT
able"
## [21] "BIC" "AIC"

head(random.effects(m10.lme.cor.heter))

## (Intercept)
## 1 1.63133019
## 2 -2.47484120
## 3 -0.04866213

```

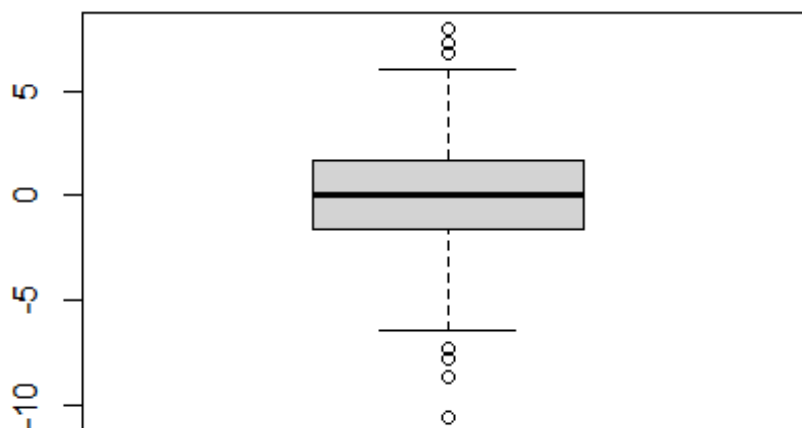


```
## 4 4.31078301
## 5 6.85607148
## 6 -1.88474954

summary(random.effects(m10.lme.cor.heter))

## (Intercept)
## Min.      :-10.61537
## 1st Qu.: -1.63464
## Median :  0.02988
## Mean      :  0.00000
## 3rd Qu.:  1.71246
## Max.      :  8.00003

boxplot(random.effects(m10.lme.cor.heter))
```



```
##=====
## Part 2)
##=====

# Loading the library to be used for multiple comparisons
library(emmeans)

## Warning: package 'emmeans' was built under R version 4.1.3

summary(m3.gls.heter) # The best model based on AIC (m3.gls.heter)
```

```

## Generalized least squares fit by maximum likelihood
##   Model: change ~ trt * ftime + y0
##   Data: data
##           AIC      BIC    logLik
##   14006.02 14084.1 -6990.009
##
## Correlation Structure: General
## Formula: ~1 | subject
## Parameter estimate(s):
## Correlation:
##   1      2
## 2 0.804
## 3 0.327 0.535
## Variance function:
## Structure: Different standard deviations per stratum
## Formula: ~1 | ftime
## Parameter estimates:
##           1      2      3
## 1.000000 1.545533 2.490193
##
## Coefficients:
##           Value Std.Error   t-value p-value
## (Intercept) 2.047604 0.3200922   6.396919  0.0000
## trt          0.940065 0.1273622   7.381035  0.0000
## ftime2       0.084476 0.0855852   0.987045  0.3237
## ftime3       0.125314 0.2125411   0.589598  0.5555
## y0           0.001087 0.0061874   0.175629  0.8606
## trt:ftime2   2.995749 0.1210357  24.750947  0.0000
## trt:ftime3   6.854956 0.3005785  22.805876  0.0000
##
## Correlation:
##           (Intr) trt      ftime2 ftime3 y0      trt:f2
## trt          -0.176
## ftime2        0.072 -0.180
## ftime3       -0.022  0.056  0.447
## y0           -0.960 -0.024  0.000  0.000
## trt:ftime2   -0.051  0.255 -0.707 -0.316  0.000
## trt:ftime3   0.016 -0.079 -0.316 -0.707  0.000  0.447
##
## Standardized residuals:
##           Min      Q1      Med      Q3      Max
## -4.48486603 -0.67604421 -0.03312136  0.70171419  3.59092048
##
## Residual standard error: 2.010834
## Degrees of freedom: 3000 total; 2993 residual

```

We observe that ftime2 and ftime3 are not statistically significant but the interaction terms are, so we will not remove any components of the fixed parts of the model

```

intervals(m3.gls.heter) # 95% CI's of the models coefficients

## Approximate 95% confidence intervals
##
## Coefficients:
##           lower      est.      upper
## (Intercept) 1.41998099 2.047604077 2.67522716
## trt          0.69033886 0.940065264 1.18979167
## ftime2       -0.08333536 0.084476397 0.25228816
## ftime3       -0.29142765 0.125313758 0.54205516
## y0           -0.01104534 0.001086692 0.01321873
## trt:ftime2    2.75842756 2.995749224 3.23307089
## trt:ftime3    6.26559431 6.854955661 7.44431701
##
## Correlation structure:
##           lower      est.      upper
## cor(1,2) 0.7808297 0.8039371 0.8248480
## cor(1,3) 0.2704132 0.3268949 0.3811342
## cor(2,3) 0.4896404 0.5353195 0.5780682
##
## Variance function:
##           lower      est.      upper
## 2 1.489601 1.545533 1.603566
## 3 2.348500 2.490193 2.640434
##
## Residual standard error:
##           lower      est.      upper
## 1.924527 2.010834 2.101012

emmeans(m3.gls.heter,pairwise~trt|ftime,adjust="tukey") # Multiple comparisons at different time points for each treatment

## Analytical Satterthwaite method not available; using appx-satterthwaite

## $emmeans
## ftime = 1:
##   trt emmean    SE    df lower.CL upper.CL
##    0   2.10 0.090  997     1.93     2.28
##    1   3.04 0.090  997     2.87     3.22
##
## ftime = 2:
##   trt emmean    SE    df lower.CL upper.CL
##    0   2.19 0.139  998     1.91     2.46
##    1   6.12 0.139  998     5.85     6.40
##
## ftime = 3:
##   trt emmean    SE    df lower.CL upper.CL
##    0   2.23 0.224 1000     1.79     2.67
##    1  10.02 0.224 1000     9.58    10.46

```

```

##
## Degrees-of-freedom method: appx-satterthwaite
## Confidence level used: 0.95
##
## $contrasts
## ftime = 1:
## contrast estimate    SE    df t.ratio p.value
## 0 - 1             -0.94 0.127  997  -7.381  <.0001
##
## ftime = 2:
## contrast estimate    SE    df t.ratio p.value
## 0 - 1             -3.94 0.197  998 -19.998  <.0001
##
## ftime = 3:
## contrast estimate    SE    df t.ratio p.value
## 0 - 1             -7.80 0.317 1000 -24.584  <.0001
##
## Degrees-of-freedom method: appx-satterthwaite

# We observe that mean change for patients receiving SoC treatment is 2
# .1 at the first time point (which refers to Week 4), while mean change
# for patients receiving Drug X is 3.04
# We observe that mean change for patients receiving SoC treatment is 2
# .19 at the second time point (which refers to Week 8), while mean chang
# e for patients receiving Drug X is 6.12
# We observe that mean change for patients receiving SoC treatment is 2
# .23 at the third time point (which refers to Week 12), while mean chang
# e for patients receiving Drug X is 10.02
# Note that we do trust the point estimates and the standard errors but
# not the p_values when using emmeans (p_values can get quite conservativ
# e!)

##=====
## Part 3
##=====

# Based on AIC, best model is the m3.gls.heter
# It's the marginal model that has unstructured covariance matrix with
# heteroscedastic errors!
summary(m3.gls.heter)

## Generalized least squares fit by maximum likelihood
## Model: change ~ trt * ftime + y0

```

```

## Data: data
##      AIC      BIC    logLik
## 14006.02 14084.1 -6990.009
##
## Correlation Structure: General
## Formula: ~1 | subject
## Parameter estimate(s):
## Correlation:
## 1      2
## 2 0.804
## 3 0.327 0.535
## Variance function:
## Structure: Different standard deviations per stratum
## Formula: ~1 | ftime
## Parameter estimates:
##      1      2      3
## 1.000000 1.545533 2.490193
##
## Coefficients:
##              Value Std.Error   t-value p-value
## (Intercept) 2.047604 0.3200922   6.396919  0.0000
## trt          0.940065 0.1273622   7.381035  0.0000
## ftime2       0.084476 0.0855852   0.987045  0.3237
## ftime3       0.125314 0.2125411   0.589598  0.5555
## y0           0.001087 0.0061874   0.175629  0.8606
## trt:ftime2   2.995749 0.1210357  24.750947  0.0000
## trt:ftime3   6.854956 0.3005785  22.805876  0.0000
##
## Correlation:
##              (Intr) trt      ftime2 ftime3 y0      trt:f2
## trt          -0.176
## ftime2       0.072 -0.180
## ftime3      -0.022  0.056  0.447
## y0          -0.960 -0.024  0.000  0.000
## trt:ftime2  -0.051  0.255 -0.707 -0.316  0.000
## trt:ftime3   0.016 -0.079 -0.316 -0.707  0.000  0.447
##
## Standardized residuals:
##              Min      Q1      Med      Q3      Max
## -4.48486603 -0.67604421 -0.03312136  0.70171419  3.59092048
##
## Residual standard error: 2.010834
## Degrees of freedom: 3000 total; 2993 residual

# Interpretations: (Πρακτικά αρκούσαν τα interpretations από το emmeans
!)
```

Intercept = 2.04 : Η μέση μέτρηση του change των ασθενών που λαμβάνουν
ν SoC στην χρονική στιγμή 0 (στην baseline!)

trt = 0.94 : Η μέση διαφορά του change μεταξύ ασθενών που λαμβάνουν D
rug X και SoC στην 1η χρονική στιγμή (και είναι στατιστικά σημαντική!)

```

# ftime2 = 0.08 : Η μέση διαφορά του change μεταξύ των ασθενών που λαμβάνουν SoC στην 1η και στην 2η χρονική στιγμή (και είναι στατιστικά ασήμαντη)
# ftime3 = 0.12 : Η μέση διαφορά του change μεταξύ των ασθενών που λαμβάνουν SoC στην 1η και στην 3η χρονική στιγμή (και είναι στατιστικά ασήμαντη)
# γ0 = 0.001 : Ο ρυθμός αλλαγής της μέτρησης της μέσης τιμής του change ως προς το χρόνο για τους ασθενείς που λαμβάνουν SoC (το επίπεδο αναφοράς) και είναι στατιστικά ασήμαντος όρος, η προσθήκη αυτού του όρου στο μοντέλο κάνει ευκολότερη την ερμηνεία των υπολοίπων!
# trt:ftime2 = 2.99 : Η διαφορά των ρυθμών αλλαγής των ασθενών που λαμβάνουν Drug X-SoC, δηλαδή οι ασθενείς που λαμβάνουν Drug X έχουν μεγαλύτερη διαφορά μεταξύ των χρονικών στιγμών 1 και 2! (και είναι στατιστικά σημαντικός όρος)
# trt:ftime3 = 6.85 : Η διαφορά των ρυθμών αλλαγής των ασθενών που λαμβάνουν Drug X-SoC, δηλαδή οι ασθενείς που λαμβάνουν Drug X έχουν μεγαλύτερη διαφορά μεταξύ των χρονικών στιγμών 1 και 3! (και είναι στατιστικά σημαντικός όρος)

# Ο πίνακας Correlation μετά τις εκτιμήσεις των coefficients, αναφέρεται στις συσχετίσεις μεταξύ των εκτιμώμενων παραμέτρων του (fixed part) μοντέλου

# Για τον αδόμητο πίνακα διακύμανσης-συνδιακύμανσης του μοντέλου έχουμε ότι:
getVarCov(m3.gls.heter)

## Marginal variance covariance matrix
##      [,1]  [,2]  [,3]
## [1,] 4.0435 5.0240 3.2915
## [2,] 5.0240 9.6585 8.3306
## [3,] 3.2915 8.3306 25.0740
## Standard Deviations: 2.0108 3.1078 5.0074

# Στην διαγώνιο του πίνακα βρίσκονται οι διακυμάνσεις των Yi για την κάθε χρονική στιγμή, i=1,2,3
# Τα μη διαγώνια στοιχεία του πίνακα είναι οι συνδιακυμάνσεις, δηλαδή τα Cov(Yij,Yij')
# 6 parameters to be estimated in this covariance matrix (3 parameters for the correlation estimates, 2 parameters for the proportions of the parameter variance estimates, 1 parameter for the residual standard error)
# Var(jk)=(σ^2)*θ(jk)
# Π.χ. για την ερμηνεία, έστω το: Cov(Yi1,Yi2) = 5.02 => Cor(Yi1,Yi2) = 5.02/(sqrt(4.0435)*sqrt(9.6585)) = 5.02/6.249332 = 0.8032859 > 0
# Αυτό λοιπόν σημαίνει ότι εφόσον έχουμε υψηλή τιμή την 1η χρονική στιγμή θα έχουμε υψηλή τιμή και την 2η χρονική στιγμή!

```

```
##=====
```

```
## Part 4a)
```

```
##=====
```

```
head(data)
```

```
## # A tibble: 6 x 5
```

```
##   id    trt    y0  time change  
##   <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1     1     1  37.2     1   1.88  
## 2     1     1  37.2     2   3.77  
## 3     1     1  37.2     3  14.4  
## 4     2     1  52.0     1   1.23  
## 5     2     1  52.0     2   4.16  
## 6     2     1  52.0     3   6.05
```

```
# Let's create a data frame that provides the data for time point 1 (w  
high refers to week 4)
```

```
df_week_4 <- data[data$time==1,]  
str(df_week_4)
```

```
## tibble [1,000 x 5] (S3: tbl_df/tbl/data.frame)  
## $ id      : num [1:1000] 1 2 3 4 5 6 7 8 9 10 ...  
## $ trt     : num [1:1000] 1 1 1 1 1 1 1 1 1 1 ...  
## $ y0      : num [1:1000] 37.2 52 50.4 63.5 42.6 ...  
## $ time    : num [1:1000] 1 1 1 1 1 1 1 1 1 1 ...  
## $ change  : num [1:1000] 1.88 1.23 1.72 6.47 6.63 ...
```

```
table(df_week_4$time)
```

```
##
```

```
##    1  
## 1000
```

```
# Confirmation that we got the right time point
```

```
# Creating the variables for the first time point
```

```
id_w4 <- df_week_4$id  
trt_w4 <- df_week_4$trt  
table(df_week_4$trt)
```

```
##
```

```
##    0    1  
## 500 500
```

Initially, at week 4 there are 500 patients that were treated with So C and 500 patients that were treated with Drug X

```
y0_w4 <- df_week_4$y0  
time_w4 <- df_week_4$time
```

```
progress_w4 <- df_week_4$change  
# Naming 'progress' the variable 'change', as we will transform it to a  
binary one:  
# 0 refers to no improvement of the patients scores, 1 refers to improv  
ement of the patients scores!  
# We were given that a change from baseline of 5 (or more) points indic  
ates an improvement in the GHS/QoL scale (scores)
```

```
# Transforming the elements of variable 'progress_w4'  
progress_w4[!progress_w4>=5] <- 0  
progress_w4[progress_w4>=5] <- 1
```

```
head(progress_w4)
```

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

```
##=====
```

Part 4b)

```
##=====
```

```
# Creating a new data frame that has the variable 'progress_w4' now  
new_df_week_4 <- data.frame(id_w4,time_w4,trt_w4,y0_w4,progress_w4)  
head(new_df_week_4,20)
```

```
##   id_w4 time_w4 trt_w4    y0_w4 progress_w4  
## 1     1      1      1 37.22891           0  
## 2     2      1      1 52.00300           0  
## 3     3      1      1 50.39272           0  
## 4     4      1      1 63.48535           1  
## 5     5      1      1 42.58701           1  
## 6     6      1      1 47.10957           0  
## 7     7      1      1 61.83045           0  
## 8     8      1      1 55.76020           0  
## 9     9      1      1 67.24634           0  
## 10    10      1      1 38.80290           0  
## 11    11      1      1 60.37134           0  
## 12    12      1      1 44.27946           0  
## 13    13      1      1 58.68293           0  
## 14    14      1      1 57.68385           0  
## 15    15      1      1 33.97440           0  
## 16    16      1      1 31.96804           0
```



```
## 17      17      1      1 39.11905      0
## 18      18      1      1 46.24886      0
## 19      19      1      1 39.77010      0
## 20      20      1      1 31.57691      0
```

```
tail(new_df_week_4,20)
```

```
##      id_w4 time_w4 trt_w4    y0_w4 progress_w4
## 981    981      1      0 52.93449      0
## 982    982      1      0 42.20966      0
## 983    983      1      0 55.11476      0
## 984    984      1      0 51.76072      1
## 985    985      1      0 41.55425      0
## 986    986      1      0 42.15235      0
## 987    987      1      0 47.49286      0
## 988    988      1      0 42.74372      0
## 989    989      1      0 57.00127      0
## 990    990      1      0 72.29195      0
## 991    991      1      0 43.36028      0
## 992    992      1      0 53.65821      0
## 993    993      1      0 51.82955      0
## 994    994      1      0 46.29451      0
## 995    995      1      0 48.09774      0
## 996    996      1      0 37.66346      0
## 997    997      1      0 49.34790      0
## 998    998      1      0 60.35074      0
## 999    999      1      0 48.57516      0
## 1000  1000      1      0 41.74584      0
```

```
table(new_df_week_4$progress_w4)
```

```
##
##    0    1
## 885 115
```

Not quite optimistic results at a first sight: We observe at week 4 that the patients that made progress (that is to mean their scores improved by 5 or more points) were only 115, while the patients that made no progress were 885

Creating a new dataframe with the patients receiving SoC at Week 4

```
new_df_week_4_SoC <- new_df_week_4[new_df_week_4$trt_w4==0,]
head(new_df_week_4_SoC)
```

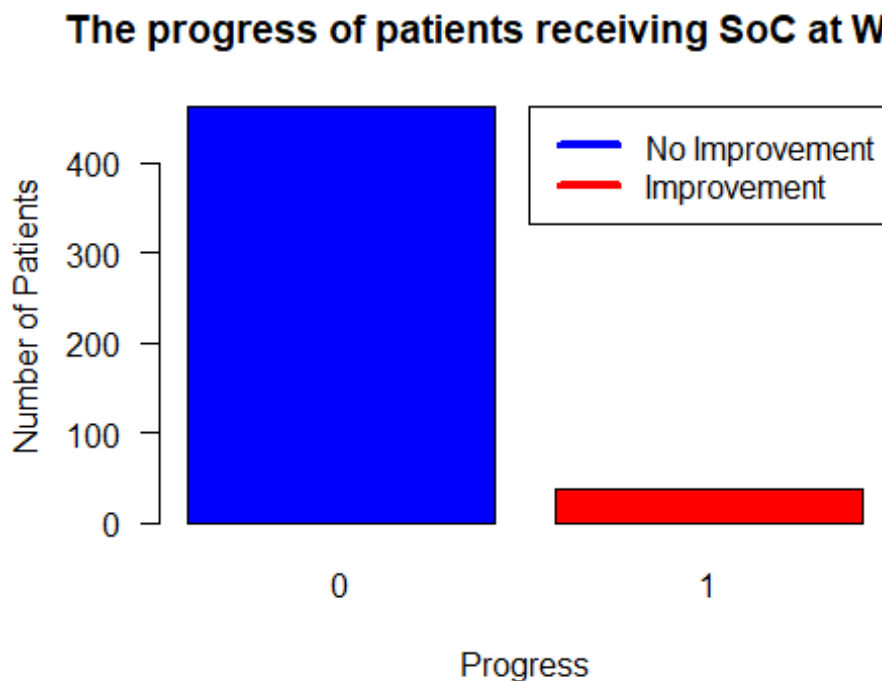
```
##      id_w4 time_w4 trt_w4    y0_w4 progress_w4
## 501    501      1      0 47.06659      0
## 502    502      1      0 37.02825      0
## 503    503      1      0 51.44153      0
```

```
## 504    504      1      0 46.57226      0
## 505    505      1      0 42.71679      1
## 506    506      1      0 47.07900      0

table(new_df_week_4_SoC$trt_w4) # Check that we got the patients receiving SoC only

##
## 0
## 500

# Visualizing the progress of the patients receiving SoC at Week 4
barplot(table(new_df_week_4_SoC$progress_w4), xlab = 'Progress', ylab = 'Number of Patients',
          main = 'The progress of patients receiving SoC at Week 4', col = c('blue', 'red'), las=1)
legend("topright", c('No Improvement', 'Improvement'), lwd=4, col=c('blue', 'red'))
```



The vast majority of patients receiving SoC at week 4 hasn't made progress!

Creating a new dataframe with the patients receiving Drug X at Week 4

```

new_df_week_4_DrugX <- new_df_week_4[new_df_week_4$trt_w4==1,]
head(new_df_week_4_DrugX)

##   id_w4 time_w4 trt_w4    y0_w4 progress_w4
## 1     1      1      1 37.22891           0
## 2     2      1      1 52.00300           0
## 3     3      1      1 50.39272           0
## 4     4      1      1 63.48535           1
## 5     5      1      1 42.58701           1
## 6     6      1      1 47.10957           0

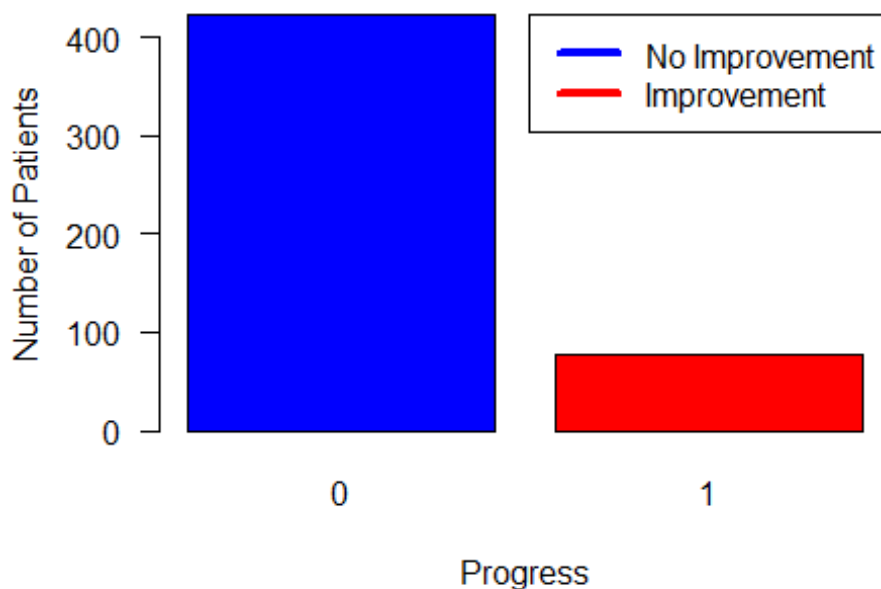
table(new_df_week_4_DrugX$trt_w4) # Check that we got the patients rece
iving Drug X only

##
##    1
## 500

# Visualizing the progress of the patients receiving Drug X at Week 4
barplot(table(new_df_week_4_DrugX$progress_w4), xlab = 'Progress', ylab =
'Number of Patients',
        main = 'The progress of patients receiving Drug X at Week 4', co
l = c('blue', 'red'), las=1)
legend("topright", c('No Improvement', 'Improvement'), lwd=4, col=c('blu
e', 'red'))

```

The progress of patients receiving Drug X at Week



```

# We observe the majority of patients receiving Drug X at week 4 hasn't
made progress either!

```

```
ftime_w4 <- as.factor(time_w4) # Convert to factor

with(new_df_week_4, tapply(progress_w4, list(trt_w4, ftime_w4), mean))

##          1
## 0 0.076
## 1 0.154
```

We observe that patients receiving Drug X seem to perform better as they have about twice the mean change scores vs patients receiving SoC

```
##=====
## Part 4c)
##=====
```

Let's build a logistic regression model since the response is binary

```
target_w4 <- as.factor(progress_w4)
ftrt_w4 <- as.factor(trt_w4)

logistic_reg.1_w4 <- glm(target_w4~ftrt_w4 + y0_w4, binomial)
summary(logistic_reg.1_w4)
```

```
##
## Call:
## glm(formula = target_w4 ~ ftrt_w4 + y0_w4, family = binomial)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.6053  -0.5762  -0.4029  -0.3942   2.2988
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -2.34002    0.53351  -4.386 1.15e-05 ***
## ftrt_w41     0.79605    0.20944   3.801 0.000144 ***
## y0_w4       -0.00319    0.01025  -0.311 0.755550
## ---
```

```

## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 713.69  on 999  degrees of freedom
## Residual deviance: 698.38  on 997  degrees of freedom
## AIC: 704.38
##
## Number of Fisher Scoring iterations: 5

coef(logistic_reg.1_w4)

##      (Intercept)      ftrt_w41      y0_w4
## -2.340018705    0.796053028 -0.003189763

exp(coef(logistic_reg.1_w4))

##      (Intercept)      ftrt_w41      y0_w4
##  0.09632584    2.21677409    0.99681532

# Building the optimal model since the baseline was statistically insignificant
# logistic_reg.2 <- glm(target~ftrt_w4,binomial)
# summary(logistic_reg.2)
# coef(logistic_reg.2)
# exp(coef(logistic_reg.2))

# Interpretation: (Στο μοντέλο logistic_reg.1_w4)
# Το exp(β1) (εφόσον ο γραμμικός προβλεπτής είναι σε logit scale) είναι
ίσο με 2.21(=1+1.21), άρα οι ασθενείς που λαμβάνουν Drug X είναι 121% πιο
πιθανό να έχουν progress σε σχέση με τους ασθενείς που λαμβάνουν SoC
!

##=====
## Part 4d)
##=====

# Let's create a data frame that provides the data for time point 2 (which
refers to week 8)

```

```

df_week_8 <- data[data$time==2,]
str(df_week_8)

## tibble [1,000 x 5] (S3: tbl_df/tbl/data.frame)
## $ id      : num [1:1000] 1 2 3 4 5 6 7 8 9 10 ...
## $ trt     : num [1:1000] 1 1 1 1 1 1 1 1 1 1 ...
## $ y0      : num [1:1000] 37.2 52 50.4 63.5 42.6 ...
## $ time    : num [1:1000] 2 2 2 2 2 2 2 2 2 2 ...
## $ change: num [1:1000] 3.77 4.16 3.98 11.73 12.06 ...

table(df_week_8$time)

##
##      2
## 1000

# Confirmation that we got the right time point

# Creating the variables for the second time point
id_w8 <- df_week_8$id
trt_w8 <- df_week_8$trt
table(df_week_8$trt)

##
##      0      1
## 500 500

# At week 8 there are 500 patients that were treated with SoC and 500 patients that were treated with Drug X

y0_w8 <- df_week_8$y0
time_w8 <- df_week_8$time

progress_w8 <- df_week_8$change
# Naming 'progress' the variable 'change', as we will transform it to a binary one:
# 0 refers to no improvement of the patients scores, 1 refers to improvement of the patients scores!
# We were given that a change from baseline of 5 (or more) points indicates an improvement in the GHS/QoL scale (scores)

# Transforming the elements of variable 'progress_w8'
progress_w8[!progress_w8>=5] <- 0
progress_w8[progress_w8>=5] <- 1

head(progress_w8)

## [1] 0 0 0 1 1 0

```

```
# Creating a new data frame that has the variable 'progress_w8' now
new_df_week_8 <- data.frame(id_w8,time_w8,trt_w8,y0_w8,progress_w8)
head(new_df_week_8,20)
```

```
##      id_w8 time_w8 trt_w8      y0_w8 progress_w8
## 1         1         2         1 37.22891          0
## 2         2         2         1 52.00300          0
## 3         3         2         1 50.39272          0
## 4         4         2         1 63.48535          1
## 5         5         2         1 42.58701          1
## 6         6         2         1 47.10957          0
## 7         7         2         1 61.83045          1
## 8         8         2         1 55.76020          1
## 9         9         2         1 67.24634          1
## 10        10         2         1 38.80290          1
## 11        11         2         1 60.37134          0
## 12        12         2         1 44.27946          0
## 13        13         2         1 58.68293          1
## 14        14         2         1 57.68385          1
## 15        15         2         1 33.97440          1
## 16        16         2         1 31.96804          1
## 17        17         2         1 39.11905          0
## 18        18         2         1 46.24886          0
## 19        19         2         1 39.77010          0
## 20        20         2         1 31.57691          0
```

```
tail(new_df_week_8,20)
```

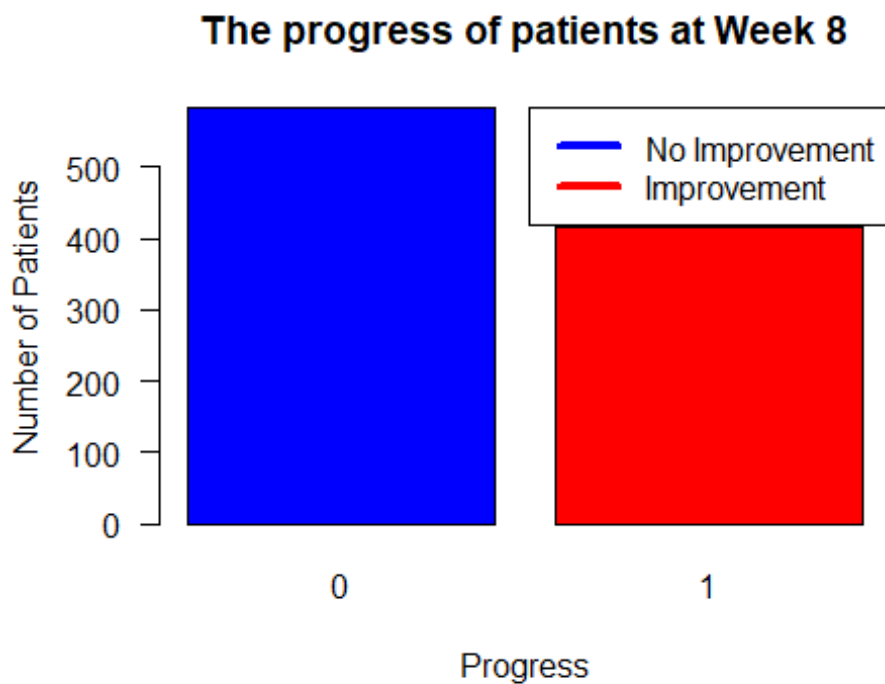
```
##      id_w8 time_w8 trt_w8      y0_w8 progress_w8
## 981      981         2         0 52.93449          0
## 982      982         2         0 42.20966          0
## 983      983         2         0 55.11476          0
## 984      984         2         0 51.76072          1
## 985      985         2         0 41.55425          0
## 986      986         2         0 42.15235          1
## 987      987         2         0 47.49286          0
## 988      988         2         0 42.74372          0
## 989      989         2         0 57.00127          0
## 990      990         2         0 72.29195          1
## 991      991         2         0 43.36028          1
## 992      992         2         0 53.65821          0
## 993      993         2         0 51.82955          0
## 994      994         2         0 46.29451          0
## 995      995         2         0 48.09774          0
## 996      996         2         0 37.66346          0
## 997      997         2         0 49.34790          0
## 998      998         2         0 60.35074          0
## 999      999         2         0 48.57516          0
## 1000     1000         2         0 41.74584          0
```

```
table(new_df_week_8$progress_w8)
```

```
##
##    0    1
## 583 417

# At week 8 we observe that the patients that made progress (that is to
mean their scores improved by 5 or more points) were 417, while the pat
ients that made no progress were 583

barplot(table(new_df_week_8$progress_w8), xlab = 'Progress', ylab = 'Nu
mber of Patients',
        main = 'The progress of patients at Week 8', col = c('blue','red
'), las=1)
legend("topright", c('No Improvement','Improvement'), lwd=4, col=c('blu
e', 'red'))
```



The results now are more balanced in comparison to Week 4

```
ftime_w8 <- as.factor(time_w8) # Convert to factor

with(new_df_week_8, tapply(progress_w8, list(trt_w8, ftime_w8), mean))
```



```
##          2
## 0 0.196
## 1 0.638

# We observe that patients receiving Drug X seem to perform better as they have about the triple mean change scores vs patients receiving SoC

# Let's build a logistic regression model since the response is binary
target_w8 <- as.factor(progress_w8)
ftrt_w8 <- as.factor(trt_w8)

logistic_reg.1_w8 <- glm(target_w8~ftrt_w8 + y0_w8,binomial)
summary(logistic_reg.1_w8)

##
## Call:
## glm(formula = target_w8 ~ ftrt_w8 + y0_w8, family = binomial)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.4425  -0.6641  -0.6573   0.9487   1.8204
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.337340    0.382670  -3.495 0.000474 ***
## ftrt_w81     1.979015    0.146196  13.537 < 2e-16 ***
## y0_w8        -0.001495    0.007379  -0.203 0.839470
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 1358.6  on 999  degrees of freedom
## Residual deviance: 1149.3  on 997  degrees of freedom
## AIC: 1155.3
##
## Number of Fisher Scoring iterations: 4

coef(logistic_reg.1_w8)

## (Intercept)      ftrt_w81      y0_w8
## -1.33733986   1.97901523  -0.00149477

exp(coef(logistic_reg.1_w8))

## (Intercept)      ftrt_w81      y0_w8
##  0.2625431    7.2356141    0.9985063
```

```
# Interpretation: (Στο μοντέλο logistic_reg.1_w8)
# To  $\exp(\beta_1)$  (εφόσον ο γραμμικός προβλεπτής είναι σε Logit scale) είναι
ίσο με 7.23(=1+6.23), άρα οι ασθενείς που λαμβάνουν Drug X είναι 623% π
ιο πιθανό να έχουν progress σε σχέση με τους ασθενείς που λαμβάνουν SoC
!
```

```
# Let's create a data frame that provides the data for time point 3 (w
hich refers to week 12)
```

```
df_week_12 <- data[data$time==3,]
str(df_week_12)
```

```
## tibble [1,000 x 5] (S3: tbl_df/tbl/data.frame)
## $ id      : num [1:1000] 1 2 3 4 5 6 7 8 9 10 ...
## $ trt     : num [1:1000] 1 1 1 1 1 1 1 1 1 1 ...
## $ y0      : num [1:1000] 37.2 52 50.4 63.5 42.6 ...
## $ time    : num [1:1000] 3 3 3 3 3 3 3 3 3 3 ...
## $ change: num [1:1000] 14.38 6.05 10.78 17.18 22.97 ...
```

```
table(df_week_12$time)
```

```
##
##      3
## 1000
```

```
# Confirmation that we got the right time point
```

```
# Creating the variables for the third time point
```

```
id_w12 <- df_week_12$id
trt_w12 <- df_week_12$trt
table(df_week_12$trt)
```

```
##
##      0      1
## 500 500
```

```
# At week 12 there are 500 patients that were treated with SoC and 500
patients that were treated with Drug X
```

```
y0_w12 <- df_week_12$y0
time_w12 <- df_week_12$time
```

```
progress_w12 <- df_week_12$change
```

```
# Naming 'progress' the variable 'change', as we will transform it to a
```

binary one:

0 refers to no improvement of the patients scores, 1 refers to improvement of the patients scores!

We were given that a change from baseline of 5 (or more) points indicates an improvement in the GHS/QoL scale (scores)

Transforming the elements of variable 'progress_w12'

```
progress_w12[!progress_w12>=5] <- 0
```

```
progress_w12[progress_w12>=5] <- 1
```

```
head(progress_w12)
```

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

Creating a new data frame that has the variable 'progress_w12' now

```
new_df_week_12 <- data.frame(id_w12,time_w12,trt_w12,y0_w12,progress_w12)
```

```
head(new_df_week_12,20)
```

```
##   id_w12 time_w12 trt_w12   y0_w12 progress_w12
## 1      1        3      1 37.22891            1
## 2      2        3      1 52.00300            1
## 3      3        3      1 50.39272            1
## 4      4        3      1 63.48535            1
## 5      5        3      1 42.58701            1
## 6      6        3      1 47.10957            1
## 7      7        3      1 61.83045            1
## 8      8        3      1 55.76020            1
## 9      9        3      1 67.24634            1
## 10     10        3      1 38.80290            1
## 11     11        3      1 60.37134            0
## 12     12        3      1 44.27946            0
## 13     13        3      1 58.68293            1
## 14     14        3      1 57.68385            1
## 15     15        3      1 33.97440            1
## 16     16        3      1 31.96804            1
## 17     17        3      1 39.11905            1
## 18     18        3      1 46.24886            0
## 19     19        3      1 39.77010            1
## 20     20        3      1 31.57691            1
```

```
tail(new_df_week_12,20)
```

```
##   id_w12 time_w12 trt_w12   y0_w12 progress_w12
## 981    981        3      0 52.93449            0
## 982    982        3      0 42.20966            1
## 983    983        3      0 55.11476            0
```

```
## 984      984      3      0 51.76072      1
## 985      985      3      0 41.55425      0
## 986      986      3      0 42.15235      0
## 987      987      3      0 47.49286      0
## 988      988      3      0 42.74372      0
## 989      989      3      0 57.00127      0
## 990      990      3      0 72.29195      1
## 991      991      3      0 43.36028      0
## 992      992      3      0 53.65821      0
## 993      993      3      0 51.82955      0
## 994      994      3      0 46.29451      1
## 995      995      3      0 48.09774      1
## 996      996      3      0 37.66346      0
## 997      997      3      0 49.34790      1
## 998      998      3      0 60.35074      1
## 999      999      3      0 48.57516      0
## 1000     1000      3      0 41.74584      0
```

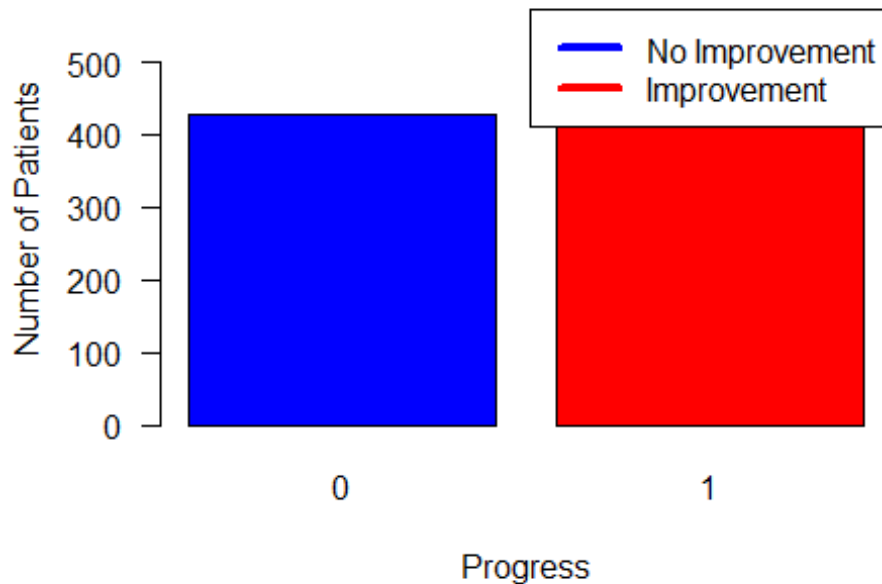
```
table(new_df_week_12$progress_w12)
```

```
##
##    0    1
## 428 572
```

At week 12 we observe that the patients that made progress (that is to mean their scores improved by 5 or more points) were 417, while the patients that made no progress were 583

```
barplot(table(new_df_week_12$progress_w12), xlab = 'Progress', ylab = '
Number of Patients',
        main = 'The progress of patients at Week 12', col = c('blue','red'), las=1)
legend("topright", c('No Improvement','Improvement'), lwd=4, col=c('blue', 'red'))
```

The progress of patients at Week 12



The results now are definitely more encouraging

```
ftime_w12 <- as.factor(time_w12) # Convert to factor
```

```
with(new_df_week_12, tapply(progress_w12, list(trt_w12, ftime_w12), mean))
```

```
##      3
```

```
## 0 0.298
```

```
## 1 0.846
```

We observe that patients receiving Drug X seem to perform better as they have about the triple mean change scores vs patients receiving SoC

Let's build a logistic regression model since the response is binary

```
target_w12 <- as.factor(progress_w12)
```

```
ftrt_w12 <- as.factor(trt_w12)
```

```
logistic_reg.1_w12 <- glm(target_w12~ftrt_w12 + y0_w12, binomial)
```

```
summary(logistic_reg.1_w12)
```

```
##
```

```
## Call:
```

```
## glm(formula = target_w12 ~ ftrt_w12 + y0_w12, family = binomial)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q        Max
## -1.9592  -0.8410   0.5703   0.5843   1.5944
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.015927   0.406988  -2.496   0.0126 *
## ftrt_w121    2.559344   0.157851  16.214 <2e-16 ***
## y0_w12       0.003201   0.007939   0.403   0.6868
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 1365.5  on 999  degrees of freedom
## Residual deviance: 1038.6  on 997  degrees of freedom
## AIC: 1044.6
##
## Number of Fisher Scoring iterations: 4

coef(logistic_reg.1_w12)

##      (Intercept)      ftrt_w121      y0_w12
## -1.015927245    2.559344293    0.003200751

exp(coef(logistic_reg.1_w12))

##      (Intercept)      ftrt_w121      y0_w12
##    0.3620665    12.9273380    1.0032059

# Interpretation: (Στο μοντέλο logistic_reg.1_w12)
# Το exp(β1) (εφόσον ο γραμμικός προβλεπτής είναι σε logit scale) είναι
ίσο με 12.97(=1+11.97), άρα οι ασθενείς που λαμβάνουν Drug X είναι 1197
% πιο πιθανό να έχουν progress σε σχέση με τους ασθενείς που λαμβάνουν
SoC!

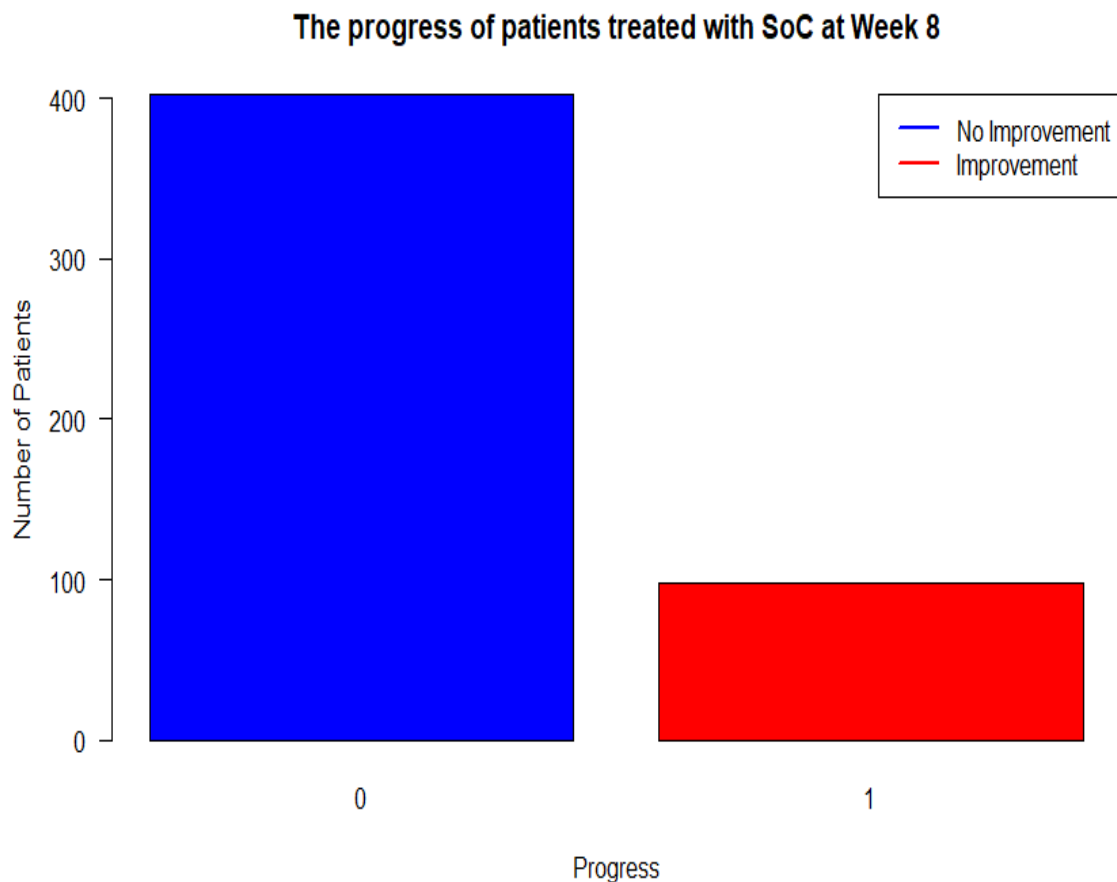
# Creating a new dataframe with the patients receiving SoC at Week 8
new_df_week_8_SoC <- new_df_week_8[new_df_week_8$ftrt_w8==0,]
head(new_df_week_8_SoC)
```

```
##      id_w8 time_w8 trt_w8    y0_w8 progress_w8
## 501    501      2      0 47.06659           0
## 502    502      2      0 37.02825           0
## 503    503      2      0 51.44153           0
## 504    504      2      0 46.57226           0
## 505    505      2      0 42.71679           1
## 506    506      2      0 47.07900           0
```

```
table(new_df_week_8_SoC$trt_w8) # Check that we got the patients receiving SoC only
```

```
##
## 0
## 500
```

```
# Visualizing the progress of the patients receiving SoC at Week 8
barplot(table(new_df_week_8_SoC$progress_w8), xlab = 'Progress', ylab = 'Number of Patients',
        main = 'The progress of patients receiving SoC at Week 8', col = c('blue', 'red'), las=1)
legend("topright", c('No Improvement', 'Improvement'), lwd=4, col=c('blue', 'red'))
```



We observe that the majority of patients receiving SoC at Week 8 made no progress!

Creating a new dataframe with the patients receiving Drug X at Week 8
new_df_week_8_DrugX <- new_df_week_8[new_df_week_8\$trt_w8==1,]
head(new_df_week_8_DrugX)

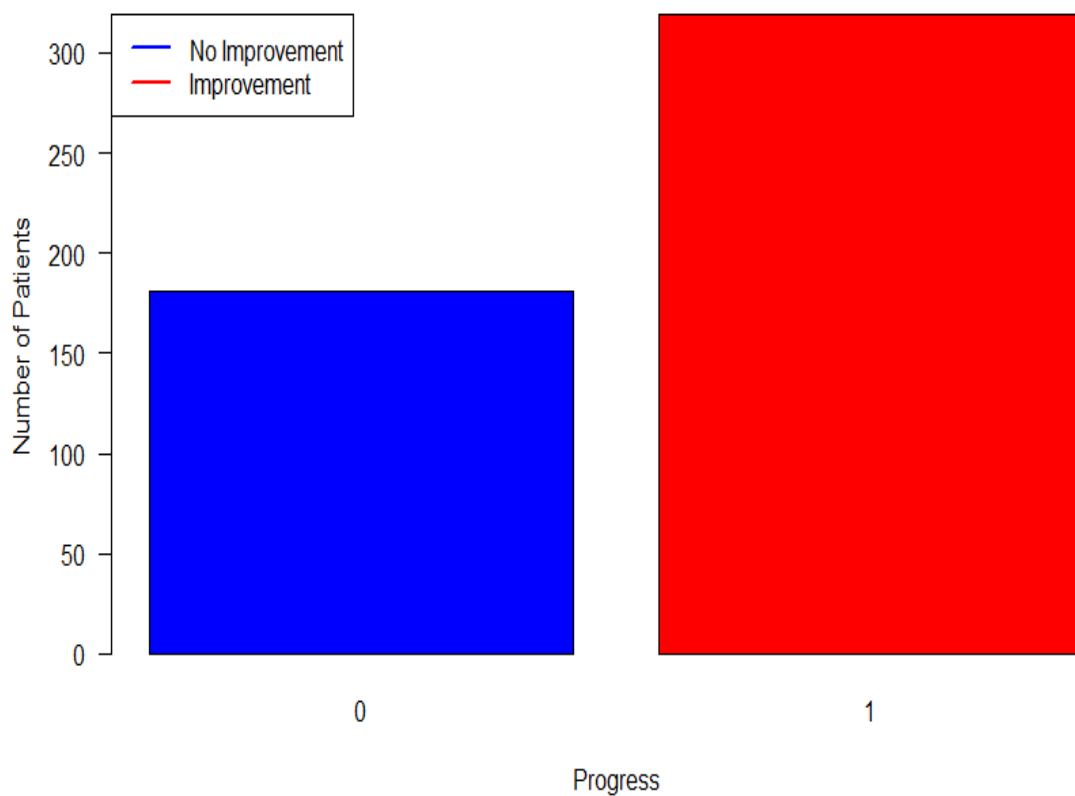
##	id_w8	time_w8	trt_w8	y0_w8	progress_w8
## 1	1	2	1	37.22891	0
## 2	2	2	1	52.00300	0
## 3	3	2	1	50.39272	0
## 4	4	2	1	63.48535	1
## 5	5	2	1	42.58701	1
## 6	6	2	1	47.10957	0

table(new_df_week_8_DrugX\$trt_w8) *# Check that we got the patients receiving Drug X only*

```
##  
## 1  
## 500
```

Visualizing the progress of the patients receiving Drug X at Week 8
barplot(table(new_df_week_8_DrugX\$progress_w8), xlab = 'Progress', ylab = 'Number of Patients',
main = 'The progress of patients receiving Drug X at Week 8', col = c('blue','red'), las=1)
legend("topleft", c('No Improvement','Improvement'), lwd=4, col=c('blue', 'red'))

The progress of patients treated with Drug X at Week 8



We observe a noticeable difference in comparison to week 8 where there wasn't a considerable difference between the patients that made progress and those who didn't
The majority of patients receiving Drug X has made progress!

Creating a new dataframe with the patients receiving SoC at Week 12
`new_df_week_12_SoC <- new_df_week_12[new_df_week_12$trt_w12==0,]`
`head(new_df_week_12_SoC)`

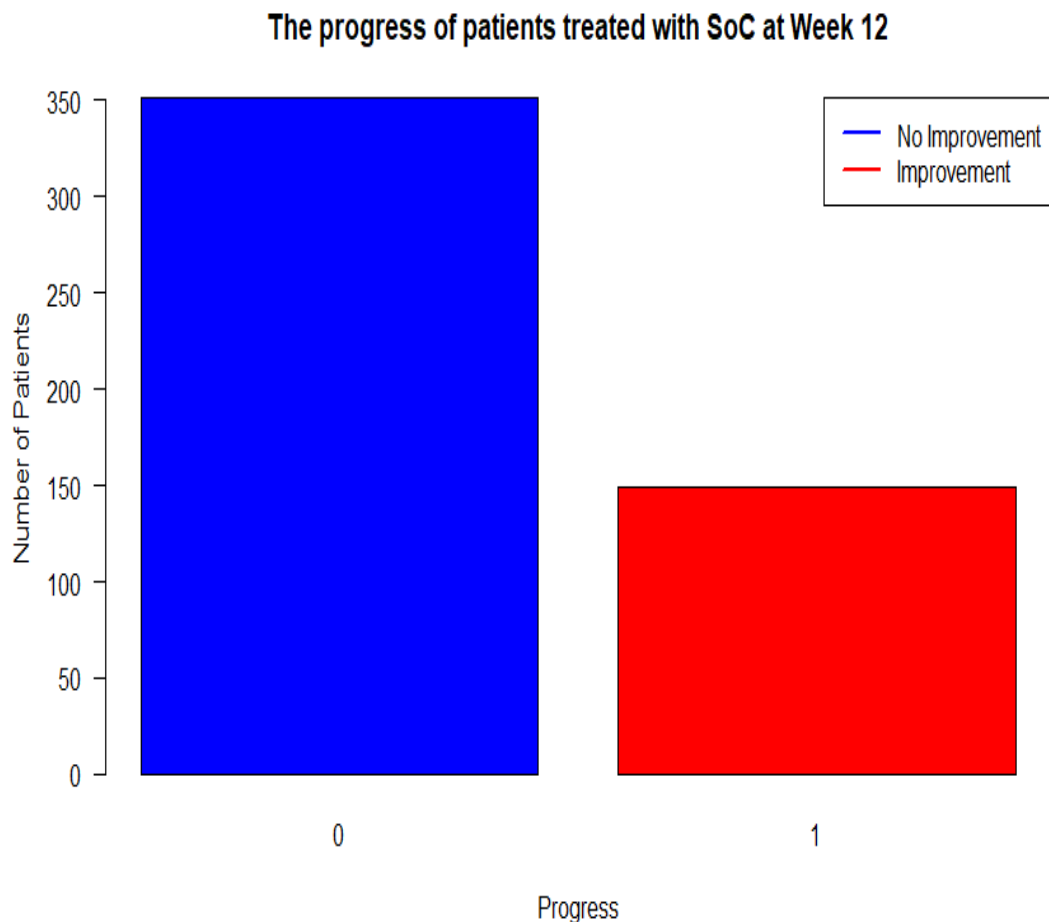
```
##      id_w12 time_w12 trt_w12   y0_w12 progress_w12
## 501     501       3       0 47.06659           0
## 502     502       3       0 37.02825           0
## 503     503       3       0 51.44153           1
## 504     504       3       0 46.57226           0
```

```
## 505      505      3      0 42.71679      1
## 506      506      3      0 47.07900      1

table(new_df_week_12_SoC$trt_w12) # Check that we got the patients receiving SoC only

##
## 0
## 500

# Visualizing the progress of the patients receiving SoC at Week 12
barplot(table(new_df_week_12_SoC$progress_w12), xlab = 'Progress', ylab = 'Number of Patients',
          main = 'The progress of patients receiving SoC at Week 12', col = c('blue', 'red'), las=1)
legend("topright", c('No Improvement', 'Improvement'), lwd=4, col=c('blue', 'red'))
```



```
# We observe that the majority of patients receiving SoC at Week 12 made no progress
```

```

# Creating a new dataframe with the patients receiving Drug X at Week 1
2
new_df_week_12_DrugX <- new_df_week_12[new_df_week_12$trt_w12==1,]
head(new_df_week_12_DrugX)

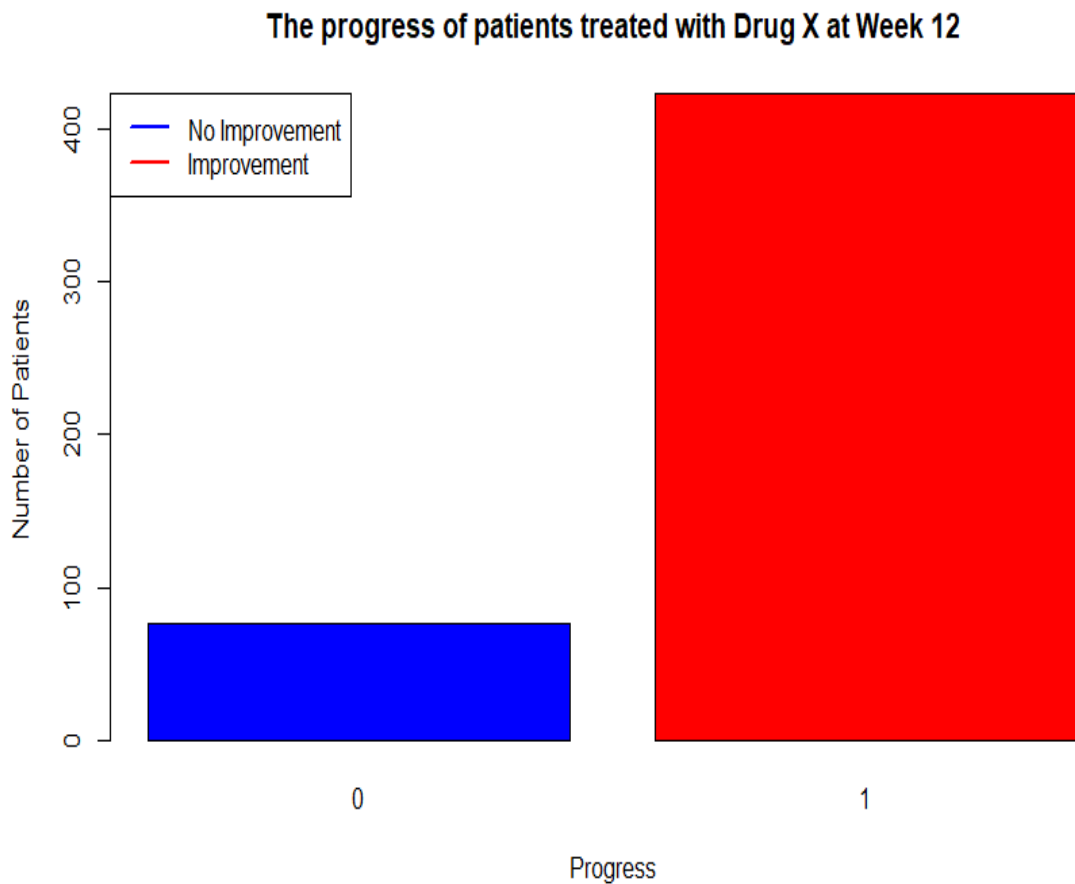
##   id_w12 time_w12 trt_w12   y0_w12 progress_w12
## 1      1       3       1 37.22891           1
## 2      2       3       1 52.00300           1
## 3      3       3       1 50.39272           1
## 4      4       3       1 63.48535           1
## 5      5       3       1 42.58701           1
## 6      6       3       1 47.10957           1

table(new_df_week_12_DrugX$trt_w12) # Check that we got the patients re
ceiving Drug X only

##
## 1
## 500

# Visualizing the progress of the patients receiving Drug X at Week 12
barplot(table(new_df_week_12_DrugX$progress_w12), xlab = 'Progress', ylab = 'Number of Patients',
        main = 'The progress of patients receiving Drug X at Week 12', col = c('blue', 'red'), las=1)
legend("topleft", c('No Improvement', 'Improvement'), lwd=4, col=c('blue', 'red'))

```



We observe a noticeable difference in comparison to week 12 where there wasn't a considerable difference between the patients that made progress and those who didn't
The vast majority of patients receiving Drug X has made progress, the results now are overwhelming! Drug X seems clearly superior to SoC!

GEE Approach

```
library(geepack)

## Warning: package 'geepack' was built under R version 4.1.3

head(data)

## # A tibble: 6 x 5
##   id   trt   y0 time change
##   <dbl> <dbl> <dbl> <dbl>   <dbl>
## 1     1     1 37.2     1     1.88
## 2     1     1 37.2     2     3.77
## 3     1     1 37.2     3    14.4
## 4     2     1 52.0     1     1.23
```

```
## 5      2      1 52.0      2  4.16
## 6      2      1 52.0      3  6.05

str(data)

## tibble [3,000 x 5] (S3: tbl_df/tbl/data.frame)
## $ id      : num [1:3000] 1 1 1 2 2 2 3 3 3 4 ...
## $ trt     : num [1:3000] 1 1 1 1 1 1 1 1 1 1 ...
## $ y0      : num [1:3000] 37.2 37.2 37.2 52 52 ...
## $ time    : num [1:3000] 1 2 3 1 2 3 1 2 3 1 ...
## $ change: num [1:3000] 1.88 3.77 14.38 1.23 4.16 ...

overall_progress <- c(progress_w4,progress_w8,progress_w12)

# Descriptive statistics
with(data,tapply(overall_progress,list(trt,ftime),mean))

##          1          2          3
## 0 0.442 0.440 0.458
## 1 0.294 0.306 0.268

with(data,tapply(overall_progress,list(trt,ftime),var))

##          1          2          3
## 0 0.2471303 0.2468938 0.2487335
## 1 0.2079800 0.2127896 0.1965691

# Fitting GEE models
gee1 <- geeglm(overall_progress~trt*ftime + y0,binomial,id=subject,cors
tr="exchangeable",std.err="san.se",data=data)
summary(gee1)

##
## Call:
## geeglm(formula = overall_progress ~ trt * ftime + y0, family = binom
ial,
## data = data, id = subject, corstr = "exchangeable", std.err = "s
an.se")
##
## Coefficients:
##             Estimate   Std.err   Wald Pr(>|W|)
## (Intercept)  0.046863  0.268699  0.030   0.862
## trt         -0.640589  0.133347 23.078 1.56e-06 ***
## ftime2      -0.008116  0.104287  0.006   0.938
## ftime3       0.064677  0.100985  0.410   0.522
## y0          -0.005641  0.005143  1.203   0.273
## trt:ftime2   0.065322  0.152399  0.184   0.668
## trt:ftime3  -0.193560  0.156284  1.534   0.216
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation structure = exchangeable
```

```

## Estimated Scale Parameters:
##
##           Estimate Std.err
## (Intercept)  0.9997 0.01893
##   Link = identity
##
## Estimated Correlation Parameters:
##           Estimate Std.err
## alpha    0.3357 0.02398
## Number of clusters: 1000 Maximum cluster size: 3

gee2 <- geeglm(overall_progress~trt*ftime + y0,binomial,id=subject,corstr="ar1",data=data)
summary(gee2)

##
## Call:
## geeglm(formula = overall_progress ~ trt * ftime + y0, family = binomial,
##   data = data, id = subject, corstr = "ar1")
##
## Coefficients:
##           Estimate Std.err Wald Pr(>|W|)
## (Intercept)  0.02521  0.27243  0.01    0.93
## trt         -0.64076  0.13335 23.09 1.5e-06 ***
## ftime2       -0.00810  0.10428  0.01    0.94
## ftime3        0.06471  0.10097  0.41    0.52
## y0           -0.00520  0.00523  0.99    0.32
## trt:ftime2    0.06535  0.15238  0.18    0.67
## trt:ftime3   -0.19353  0.15627  1.53    0.22
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation structure = ar1
## Estimated Scale Parameters:
##
##           Estimate Std.err
## (Intercept)          1  0.0189
##   Link = identity
##
## Estimated Correlation Parameters:
##           Estimate Std.err
## alpha    0.406  0.0254
## Number of clusters: 1000 Maximum cluster size: 3

gee3 <- geeglm(overall_progress~trt*time + y0,binomial,id=subject,corstr="unstructured",data=data)
summary(gee3)

##
## Call:

```

```

## geeglm(formula = overall_progress ~ trt * time + y0, family = binomial,
##       data = data, id = subject, constr = "unstructured")
##
## Coefficients:
##             Estimate Std.err Wald Pr(>|W|)
## (Intercept)  0.00219  0.28218  0.00   0.9938
## trt          -0.49286  0.18336  7.22   0.0072 **
## time          0.03230  0.05057  0.41   0.5230
## y0           -0.00566  0.00515  1.21   0.2715
## trt:time      -0.09534  0.07736  1.52   0.2178
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation structure = unstructured
## Estimated Scale Parameters:
##
##             Estimate Std.err
## (Intercept)          1  0.0189
## Link = identity
##
## Estimated Correlation Parameters:
##             Estimate Std.err
## alpha.1:2    0.343  0.0324
## alpha.1:3    0.325  0.0323
## alpha.2:3    0.339  0.0325
## Number of clusters: 1000 Maximum cluster size: 3

# We observe that the exchangeable is preferred over the unstructured

gee4 <- geeglm(overall_progress~trt + y0,binomial,id=subject,constr="exchangeable",std.err="san.se",data=data)
summary(gee4)

##
## Call:
## geeglm(formula = overall_progress ~ trt + y0, family = binomial,
##       data = data, id = subject, constr = "exchangeable", std.err = "san.se")
##
## Coefficients:
##             Estimate Std.err Wald Pr(>|W|)
## (Intercept)  0.06776  0.26170  0.07   0.80
## trt          -0.68215  0.09953 46.97 7.2e-12 ***
## y0           -0.00568  0.00514  1.22   0.27
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation structure = exchangeable

```

```
## Estimated Scale Parameters:
##
##           Estimate Std.err
## (Intercept)      1  0.0188
##   Link = identity
##
## Estimated Correlation Parameters:
##           Estimate Std.err
## alpha      0.335  0.0239
## Number of clusters: 1000 Maximum cluster size: 3

exp(coef(gee4))

## (Intercept)      trt      y0
##      1.070      0.506      0.994

# Residuals plot
plot(fitted(gee4), resid(gee1, type="pearson"))
```

