Mandatory 2 - STK9900

Inger Annett Grünbeck

2023-04-13

Importing libraries:

```
include = FALSE
message = FALSE
warning = FALSE

library(rcompanion)
library(car)
```

```
## Loading required package: carData
```

```
library(data.table)
library(ggplot2)
library(MASS)
```

Exercise 1

Importing the horseshoe crab dataset:

```
crabs=read.table("https://www.uio.no/studier/emner/matnat/math/STK4900/data/crabs.txt", header=TRUE, co
```

The variables y, color, spine were imported as factors, width and weigh as numerical variables.

1a)

In this dataset the outcome variable y corresponds to whether a crab had one or more satellites (binary outcome), where 1 and 0 respectively correspond to yes and no. It is therefore reasonable to use a regression model that calculates the probability of y=1 taking place, based on the models covariates. The log regression is a good fit for this: $p = \frac{\exp(\beta_0+\beta_1 x_1)}{1+\exp(\beta_0+\beta_1 x_1)}$, where p corresponds to the probability of y=1 (indicating one or more satellites), and x_1 corresponds to the crabs width.

```
fit_width = glm(y~width, data=crabs, family=binomial)
summary(fit_width)
```

```
##
## Call:
## glm(formula = y ~ width, family = binomial, data = crabs)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.0281  -1.0458   0.5480   0.9066   1.6942
##
## Coefficients:
```

```
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -12.3508     2.6287  -4.698 2.62e-06 ***
## width         0.4972     0.1017   4.887 1.02e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 225.76  on 172  degrees of freedom
## Residual deviance: 194.45  on 171  degrees of freedom
## AIC: 198.45
##
## Number of Fisher Scoring iterations: 4
```

1b)

In order to find the odds ratio of presence of satellites between crabs that differ 1 cm in width, and the 95% confidence interval, I apply the code used in datalab 18:

```
exp.coef.func = function(fit_width) {

  alpha = 0.05
  coef.mat = summary(fit_width)$coef

  lower = exp(coef.mat[,1] - qnorm(p=1-alpha/2)*coef.mat[,2])
  upper = exp(coef.mat[,1] + qnorm(p=1-alpha/2)*coef.mat[,2])
  result = cbind(estimate=exp(coef.mat[,1]), lower, upper)

  return(result)
}

exp.coef.func(fit_width)
```

```
##                 estimate        lower        upper
## (Intercept) 4.326214e-06 2.503452e-08 0.0007476128
## width       1.644162e+00 1.346936e+00 2.0069749360
```

The odds ratio for the presence of satellites between crabs that differ with one cm is 1.644, meaning that the odds for the presence of satellites is more than the double for the one cm larger crab, compared to the smaller crab's odds.

The relative risk is calculated using the probability for satellites occurring when the width of the crab is 0 and 1 for one cm increase, p(0) and p(1):
$RR = \frac{p(1)}{p(0)}$.

If p(1) and p(0) both are small, OR≈RR, as the odds ratio is defined as:
$OR = \frac{\frac{p(1)}{[1-p(1)]}}{\frac{p(0)}{[1-p(0)]}}$.

We can calculate the probabilities by using the equation expressing p, mentioned in 1a). From the r output in 1b) we can see that that p(0)=0.00000433. from the r output in 1a), we can see that β_1=0.4971, which is relative low increase from −12.3. We can therefore assume that p(1) also will be small. It can be controlled by calculating p(1) using the equation from 1a) again:

```
beta_0 = summary(fit_width)$coef[1]
beta_1 = summary(fit_width)$coef[2]
```

```
p = (exp(beta_0+beta_1))/(1 + exp(beta_0+beta_1))
print(p)
```

```
## [1] 7.112945e-06
```

The calculation confirms that both p(0) and p(1) are low, and therefore the odds ratio can be considered as an approximation to a relative risk in this situation.

1c)

As the color and spine conditions have been categorized into groups, these covariates will be included as categorical variables. The width and weight are both continous variables, and therefore included as numerical variables.

I define one log regression model for each of the other covariates, adressing them one at a time:

Model based on weight as predictor:

```
fit_weight = glm(y~weight, data=crabs, family=binomial)
summary(fit_weight)
```

```
##
## Call:
## glm(formula = y ~ weight, family = binomial, data = crabs)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.1108  -1.0749   0.5426   0.9122   1.6285
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -3.6947     0.8802  -4.198 2.70e-05 ***
## weight        1.8151     0.3767   4.819 1.45e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 225.76  on 172  degrees of freedom
## Residual deviance: 195.74  on 171  degrees of freedom
## AIC: 199.74
##
## Number of Fisher Scoring iterations: 4
```

Model with color as predictor:

```
fit_color = glm(y~color, data=crabs, family=binomial)
summary(fit_color)
```

```
##
## Call:
## glm(formula = y ~ color, family = binomial, data = crabs)
##
```

```
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6651  -1.3370   0.7997   0.7997   1.5134
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   1.0986     0.6667   1.648   0.0994 .
## color2       -0.1226     0.7053  -0.174   0.8620
## color3       -0.7309     0.7338  -0.996   0.3192
## color4       -1.8608     0.8087  -2.301   0.0214 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 225.76  on 172  degrees of freedom
## Residual deviance: 212.06  on 169  degrees of freedom
## AIC: 220.06
##
## Number of Fisher Scoring iterations: 4
```

Model with spine as predictor:

```
fit_spine = glm(y~spine, data=crabs, family=binomial)
summary(fit_spine)
```

```
##
## Call:
## glm(formula = y ~ spine, family = binomial, data = crabs)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.5576  -1.4385   0.8400   0.9371   1.2346
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   0.8602     0.3597   2.392   0.0168 *
## spine2       -0.9937     0.6303  -1.577   0.1149
## spine3       -0.2647     0.4068  -0.651   0.5152
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 225.76  on 172  degrees of freedom
## Residual deviance: 223.23  on 170  degrees of freedom
## AIC: 229.23
##
## Number of Fisher Scoring iterations: 4
```

Based on the summary from each of the three models, weight is definitely a significant variable, due to its high z-value.
In the predictor "color", we can see that color 4 differs significantly from the reference color "color 1". We can also see that the models residual deviance is slightly closer to its null deviance than the weight- and

width-based models' residual deviance. This could indicate that the color-based model is a worse fit than the others.
In the spine-based model, only the intercept is marked as significant, and its residual deviance is even closer to the null deviance than the color-based model.

Based on the numbers, I would conclude that at least weight has a significant influence on the presence of satellites. Maybe also the color variable. But I would not include the spine-variable.

We can compare all models using a deviance test and the test statistic G:
$G = D_0 - D$,
where D_0 is the residual deviance of a reference model and D is the residual deviance of the model we want to compare to the reference. If G is large, there is a significant difference between the models. I use the model based on width as the reference model, assuming that width has a significant effect on the presence of satellites:

```
anova(fit_width,fit_weight,fit_color, fit_spine, test="Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: y ~ width
## Model 2: y ~ weight
## Model 3: y ~ color
## Model 4: y ~ spine
##   Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
## 1       171     194.45
## 2       171     195.74  0  -1.2845
## 3       169     212.06  2 -16.3237
## 4       170     223.23 -1 -11.1716 0.0008306 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

From the comparison, we can see that only the spine-based model differs significantly from the width-based model. I would therefore assume that both weight and color might have significant effect on the satellite presence, and try to include them in a model together with width as predictors.

d)

I construct a log regression model using all variables as predictors:

```
fit_crabs = glm(y~weight+width+color+spine, data=crabs, family=binomial)
summary(fit_crabs)
```

```
##
## Call:
## glm(formula = y ~ weight + width + color + spine, family = binomial,
##     data = crabs)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.1977  -0.9424   0.4849   0.8491   2.1198
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -8.06501    3.92855  -2.053   0.0401 *
## weight       0.82578    0.70383   1.173   0.2407
## width        0.26313    0.19530   1.347   0.1779
```

```
## color2      -0.10290    0.78259  -0.131   0.8954
## color3      -0.48886    0.85312  -0.573   0.5666
## color4      -1.60867    0.93553  -1.720   0.0855 .
## spine2      -0.09598    0.70337  -0.136   0.8915
## spine3       0.40029    0.50270   0.796   0.4259
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 225.76  on 172  degrees of freedom
## Residual deviance: 185.20  on 165  degrees of freedom
## AIC: 201.2
##
## Number of Fisher Scoring iterations: 4
```

We can see that the constructed model has a lower residual deviance than the width-based model. This could indicate that it is as good or better fitting.

We can also see that neither width or weight are considered as significant predictors, in contrast to earlier models. It could be that either width or weight are confounding variables, and therefore effect each others' effect on the presence of satellites. It would make sense that a wider crab also is heavier. So width could be correlated both to the weight and the outcome of the model. We can check whether they are correlated:

```
cor(crabs$width, crabs$weight)
```

```
## [1] 0.8868715
```

We can see that weight and width are highly correlated. Therefore both variables should be included in the final model.

Further, based on the model and previous knowledge I choose to cunstruct a model excluding the spine predictor. I am also constructing a model only including weight and width for comparison:
Model based on weight, width and color:

```
fit_crabs2 = glm(y~weight+width+color, data=crabs, family=binomial)
summary(fit_crabs2)
```

```
##
## Call:
## glm(formula = y ~ weight + width + color, family = binomial,
##     data = crabs)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.1886  -1.0085   0.4949   0.8625   2.1520
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -8.6445     3.7703  -2.293   0.0219 *
## weight        0.7727     0.6978   1.107   0.2681
## width         0.2906     0.1901   1.528   0.1264
## color2        0.1310     0.7419   0.177   0.8598
## color3       -0.1610     0.7801  -0.206   0.8364
## color4       -1.2453     0.8554  -1.456   0.1455
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 225.76  on 172  degrees of freedom
## Residual deviance: 186.21  on 167  degrees of freedom
## AIC: 198.21
##
## Number of Fisher Scoring iterations: 4
```

Model based on weight and width:

```
fit_crabs3 = glm(y~weight+width, data=crabs, family=binomial)
summary(fit_crabs3)
```

```
##
## Call:
## glm(formula = y ~ weight + width, family = binomial, data = crabs)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.1127  -1.0344   0.5304   0.9006   1.7207
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -9.3547     3.5280  -2.652  0.00801 **
## weight        0.8338     0.6716   1.241  0.21445
## width         0.3068     0.1819   1.686  0.09177 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 225.76  on 172  degrees of freedom
## Residual deviance: 192.89  on 170  degrees of freedom
## AIC: 198.89
##
## Number of Fisher Scoring iterations: 4
```

In order to compare the last two models to the model including all variables, I use a deviance test again:

```
anova(fit_crabs,fit_crabs2, fit_crabs3, test="Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: y ~ weight + width + color + spine
## Model 2: y ~ weight + width + color
## Model 3: y ~ weight + width
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1       165     185.20
## 2       167     186.21 -2  -1.0091   0.6038
## 3       170     192.89 -3  -6.6808   0.0828 .
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Based on this and the summaries of the models, it does not matter much whether I include spine or not. So I remove it. There is some difference between model 3 and model 1. As model 3's residual deviance is closer to the null deviance/is larger than the other models's residual deviance, I would stick to model 2 as the final model. This is because a lower residual deviance means that the model fits the data better. But it is important to note that model 2 and 3's residual deviance is not very different.

e)

I construct a model with all covariates and their interactions:

```
fit_interaction = glm(y~weight+width+color+spine+weight*width+weight*color+width*color+width*spine+weigh
summary(fit_interaction)
```

```
##
## Call:
## glm(formula = y ~ weight + width + color + spine + weight * width +
##     weight * color + width * color + width * spine + weight *
##     spine + color * spine, family = binomial, data = crabs)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.1773  -0.8188   0.2594   0.7360   1.8287
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)     61.2314    44.9689   1.362    0.173
## weight          -2.5473    13.5681  -0.188    0.851
## width           -3.2244     2.3936  -1.347    0.178
## color2         -38.2416    41.5873  -0.920    0.358
## color3         -25.9169  1959.1796  -0.013    0.989
## color4         -48.5556  3956.4195  -0.012    0.990
## spine2          20.0129  2371.3719   0.008    0.993
## spine3         -29.3133  3956.2019  -0.007    0.994
## weight:width     0.4888     0.2865   1.706    0.088 .
## weight:color2   -9.9615    10.8531  -0.918    0.359
## weight:color3   -5.4695    11.1445  -0.491    0.624
## weight:color4   -8.8334    11.1164  -0.795    0.427
## width:color2     2.3231     2.3365   0.994    0.320
## width:color3     2.0637     2.4392   0.846    0.398
## width:color4     1.9216     2.4413   0.787    0.431
## width:spine2    -0.6915     1.1523  -0.600    0.548
## width:spine3     0.5077     0.6584   0.771    0.441
## weight:spine2    6.5991     4.7078   1.402    0.161
## weight:spine3   -2.1379     2.3024  -0.929    0.353
## color2:spine2  -17.1558  2371.2516  -0.007    0.994
## color3:spine2  -32.9556  3075.5928  -0.011    0.991
## color4:spine2  -18.5168  6076.6416  -0.003    0.998
## color2:spine3   21.9981  3956.1828   0.006    0.996
## color3:spine3    5.7598  4414.5008   0.001    0.999
## color4:spine3   38.2766  5594.8857   0.007    0.995
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 225.76  on 172  degrees of freedom
## Residual deviance: 159.25  on 148  degrees of freedom
## AIC: 209.25
##
## Number of Fisher Scoring iterations: 16
```

From the summary we can see that only the interaction between weight and width seems significant. As only one of the spine-interactions has a high z-value (weight:spine2, z=1.4), I first remove spine and all of the variable's interactions:

```
fit_interaction2 = glm(y~weight+width+color+weight*width+weight*color+width*color, data=crabs, family=bi
summary(fit_interaction2)
```

```
##
## Call:
## glm(formula = y ~ weight + width + color + weight * width + weight *
##     color + width * color, family = binomial, data = crabs)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.2845  -0.8460   0.4546   0.8861   1.8790
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)    33.6859    23.6336   1.425    0.154
## weight         -8.5471     7.5721  -1.129    0.259
## width          -1.4490     1.0603  -1.367    0.172
## color2        -18.4106    16.8661  -1.092    0.275
## color3        -24.8338    18.8059  -1.321    0.187
## color4        -19.5835    17.9876  -1.089    0.276
## weight:width    0.4067     0.2500   1.627    0.104
## weight:color2  -2.0668     3.7023  -0.558    0.577
## weight:color3   1.0899     4.0507   0.269    0.788
## weight:color4  -3.2670     3.8733  -0.843    0.399
## width:color2    0.8869     0.8801   1.008    0.314
## width:color3    0.8594     0.9661   0.890    0.374
## width:color4    0.9813     0.9271   1.059    0.290
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 225.76  on 172  degrees of freedom
## Residual deviance: 175.63  on 160  degrees of freedom
## AIC: 201.63
##
## Number of Fisher Scoring iterations: 6
```

Next, I'll remove the color interactions, as these all have high p-values (p>0.28). But I'll keep the main effect of color, as the color categories have a lower p-value:

```
fit_interaction3 = glm(y~weight+width+color+weight*width, data=crabs, family=binomial)
summary(fit_interaction3)
```

```
##
## Call:
## glm(formula = y ~ weight + width + color + weight * width, family = binomial,
##     data = crabs)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.3001  -1.0330   0.4632   0.9251   1.9602
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   1.80121   12.63042   0.143    0.887
## weight       -3.97001    5.59136  -0.710    0.478
## width        -0.10477    0.49502  -0.212    0.832
## color2        0.07719    0.74996   0.103    0.918
## color3       -0.22823    0.78799  -0.290    0.772
## color4       -1.27871    0.86141  -1.484    0.138
## weight:width  0.17861    0.20942   0.853    0.394
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 225.76  on 172  degrees of freedom
## Residual deviance: 185.44  on 166  degrees of freedom
## AIC: 199.44
##
## Number of Fisher Scoring iterations: 5
```

I'll also remove color due to the high p-values:

```
fit_interaction4 = glm(y~weight+width+weight*width, data=crabs, family=binomial)
summary(fit_interaction4)
```

```
##
## Call:
## glm(formula = y ~ weight + width + weight * width, family = binomial,
##     data = crabs)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.2380  -1.0299   0.4855   0.9484   1.5167
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    1.6580    12.2587   0.135    0.892
## weight        -4.2244     5.5120  -0.766    0.443
## width         -0.1118     0.4827  -0.232    0.817
## weight:width   0.1904     0.2065   0.922    0.357
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 225.76  on 172  degrees of freedom
## Residual deviance: 191.99  on 169  degrees of freedom
## AIC: 199.99
##
```

```
## Number of Fisher Scoring iterations: 5
```

It does not look like there are any interactions that should be included, but to be sure I perform a deviance test again, comparing the interaction models to model "fit_crabs2" (weight, width, color) from 1d):

```
anova(fit_crabs2,fit_interaction, fit_interaction2, fit_interaction3, fit_interaction4, test="Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: y ~ weight + width + color
## Model 2: y ~ weight + width + color + spine + weight * width + weight *
##     color + width * color + width * spine + weight * spine +
##     color * spine
## Model 3: y ~ weight + width + color + weight * width + weight * color +
##     width * color
## Model 4: y ~ weight + width + color + weight * width
## Model 5: y ~ weight + width + weight * width
##   Resid. Df Resid. Dev  Df Deviance Pr(>Chi)
## 1       167     186.21
## 2       148     159.25  19  26.9577  0.10564
## 3       160     175.63 -12 -16.3743  0.17469
## 4       166     185.44  -6  -9.8135  0.13273
## 5       169     191.99  -3  -6.5475  0.08781 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

As we can see from the results of the test, even though the simple model without interactions has a higher residual deviance than some of the other models, only one of them shows any sign of significant difference (model 5). But as the p-value of model 5 is > 0.05, Im not sure I would choose this model above model 1 without interactions.

Exercise 2:

Importing the dataset:

```
olympic=read.table("https://www.uio.no/studier/emner/matnat/math/STK4900/data/olympic.txt",sep="\t",hea
```

a) When modeling a poisson regression model, we "model rate data that is predicting the number of counts over a period of time or grouping." [cited from https://www.dataquest.io/blog/tutorial-poisson-regression-in-r/]. A general poisson regression model can be described as:
$log(\frac{X}{n}) = \beta_0 + \sum_i \beta_i x_i$,
where X is the event to happen and n the grouping or time period. β_i are the regression coefficients, and x_i the predictors. This can be rewritten as:
$log(X) = log(n) + \beta_0 + \sum_i \beta_i X_i$.

log(n) has the regression coefficient 1, and is called the "offset". We can also define the model as: $Y_i = Po(w_i \lambda_i)$, where w_i corresponds to n, or the number of subjects in grouping i, and λ_i to the poisson parameter.

In our case X represents the number of medals for a given nation in 2000, and n, or w_i, the logarithm of the number of athletes representing that given nation. Then λ_i is the rate of the number of medals won in 2000 per athlete representing a nation, $\frac{X_i}{n_i}$, for each nation i.

Log.athletes is a sensible choice as offset, as the number of athletes representing a nation is correlated to how many medals a nation can win during the Olympics. If we for example assume that one athlete maximum can

compete for one medal, the max number of possible medals won by the nation is the same as the number of athletes. Of course an athlete can compete for multiple medals, but the number of athletes representing the nation will still determine how many medals a nation can win. It is therefore reasonable to use the number of athletes representing the nation as offset/grouping in order to estimate the number of medals won by a nation.

In order for the model to hold as a poisson regression, we assume:
* That the rate of events λ is constant over time
* The number of events in disjoint intervals are independent
* Events do not occur together

b)

Defining a poisson model using Total2000 as outcome, log.athlete as offset and the remaining variables as predictors:

```
fit_olympic = glm(Total2000~offset(Log.athletes)+Total1996+Log.population+GDP.per.cap,data=olympic,famil
summary(fit_olympic)
```

```
##
## Call:
## glm(formula = Total2000 ~ offset(Log.athletes) + Total1996 +
##     Log.population + GDP.per.cap, family = poisson, data = olympic)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.4602  -1.0273   0.1670   0.9475   2.7748
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -2.862299   0.319076  -8.971  < 2e-16 ***
## Total1996       0.011832   0.001607   7.364 1.79e-13 ***
## Log.population  0.027510   0.031539   0.872    0.383
## GDP.per.cap    -0.014924   0.003208  -4.652 3.29e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 254.11  on 65  degrees of freedom
## Residual deviance: 131.63  on 62  degrees of freedom
## AIC: 392.31
##
## Number of Fisher Scoring iterations: 4
```

As we can see from the model output, the Log.population is not considered to be significant when estimating the number of medals for a nation in the 2000's Olympics. I therefore create a new model excluding this variable:

```
fit_olympic2 = glm(Total2000~offset(Log.athletes)+Total1996+GDP.per.cap,data=olympic,family=poisson)
summary(fit_olympic2)
```

```
##
## Call:
## glm(formula = Total2000 ~ offset(Log.athletes) + Total1996 +
##     GDP.per.cap, family = poisson, data = olympic)
```

```
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.3973  -1.0236   0.1788   0.9326   2.8277
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -2.589318   0.057648 -44.916  < 2e-16 ***
## Total1996    0.012825   0.001140  11.248  < 2e-16 ***
## GDP.per.cap -0.015800   0.003059  -5.164 2.41e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 254.11  on 65  degrees of freedom
## Residual deviance: 132.39  on 63  degrees of freedom
## AIC: 391.07
##
## Number of Fisher Scoring iterations: 4
```

From the new model, we can see that all remaining predictors are considered significant for the estimation of the outcome. We can perform a deviance test in order to compare the models and see if removing log(population) has an effect on the estimation of the outcome (based on the parameter estimations of the models, I do not expect the models to differ noticeably):

```
anova(fit_olympic, fit_olympic2, test="Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: Total2000 ~ offset(Log.athletes) + Total1996 + Log.population +
##     GDP.per.cap
## Model 2: Total2000 ~ offset(Log.athletes) + Total1996 + GDP.per.cap
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1        62     131.63
## 2        63     132.39 -1 -0.75529   0.3848
```

As we can see, there is no significant difference between the models, and I therefore choose model 2 as the final model, as this one is less complex. Based on model 2 we can examine how the estimated parameters effect the outcome:

```
exp.coef.func(fit_olympic2)
```

```
##              estimate      lower     upper
## (Intercept) 0.0750712 0.06705069 0.0840511
## Total1996   1.0129072 1.01064613 1.0151734
## GDP.per.cap 0.9843241 0.97843943 0.9902442
```

We can see that both the estimate of the Total1996 variable and the estimate of the GDP variable are close to 1, meaning the rate ratio corresponding to one units increase for a covariate is close to 1 when holding the other covariate constant. An increase of one unit for medals won in 1996 will slightly increase the medal in 2000 per athlete rate. The GDP estimate contributes slightly negatively to the rate, as the estimate is < 1. This means a nation with a high GDP will have a lower estimated medal/athlete rate in 2000 relative to another nation with a lower GDP and the same amount of medals won in 1996.

Based on this, I would conclude that wealthy nations are not more likely to win medals in competitions like the olympics, based on this dataset.

Exercise 3:

Importing the dataset. The variables status, treat, sex, asc and agegr are imported as categorival variables:

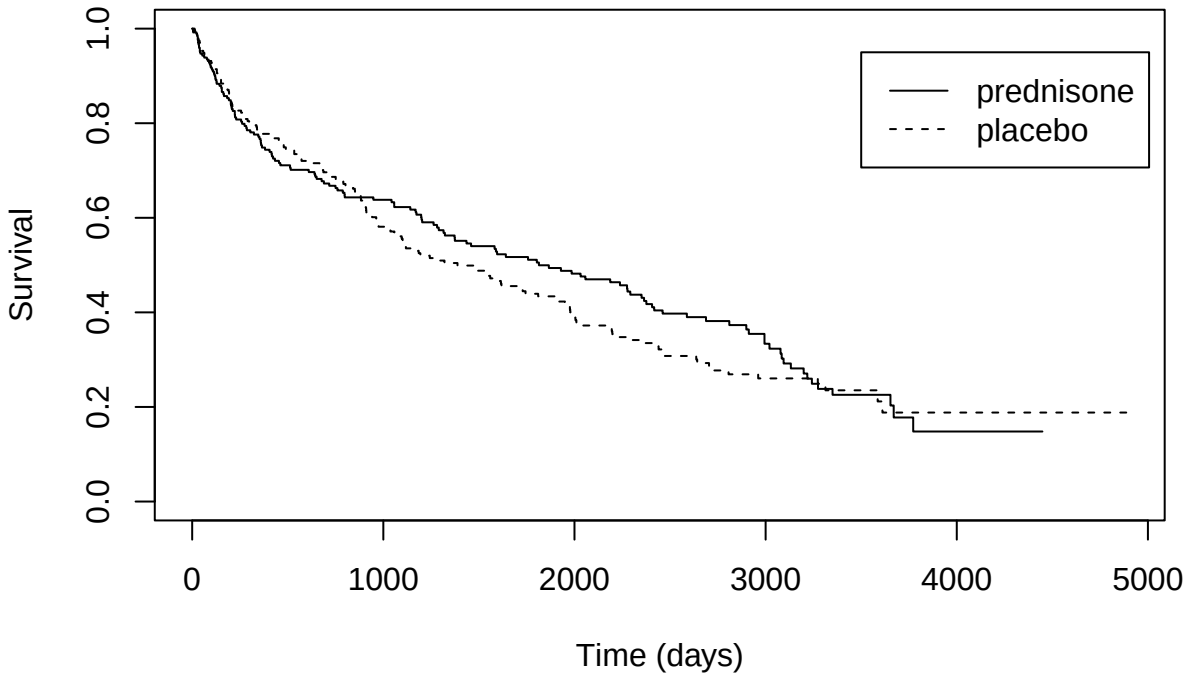
```
cirr = read.table("https://www.uio.no/studier/emner/matnat/math/STK4900/data/cirrhosis.txt", header = TI
```

```
library(survival)
```

a)

Kaplan-Meier Plot for treatment comparison:

```
surv_treat = survfit(Surv(cirr$time, cirr$status==1)~cirr$treat, conf.type="plain")
plot(surv_treat, lty=1:2, xlab="Time (days)", ylab="Survival")
legend(3500,0.95,c("prednisone","placebo"), lty=1:2)
```

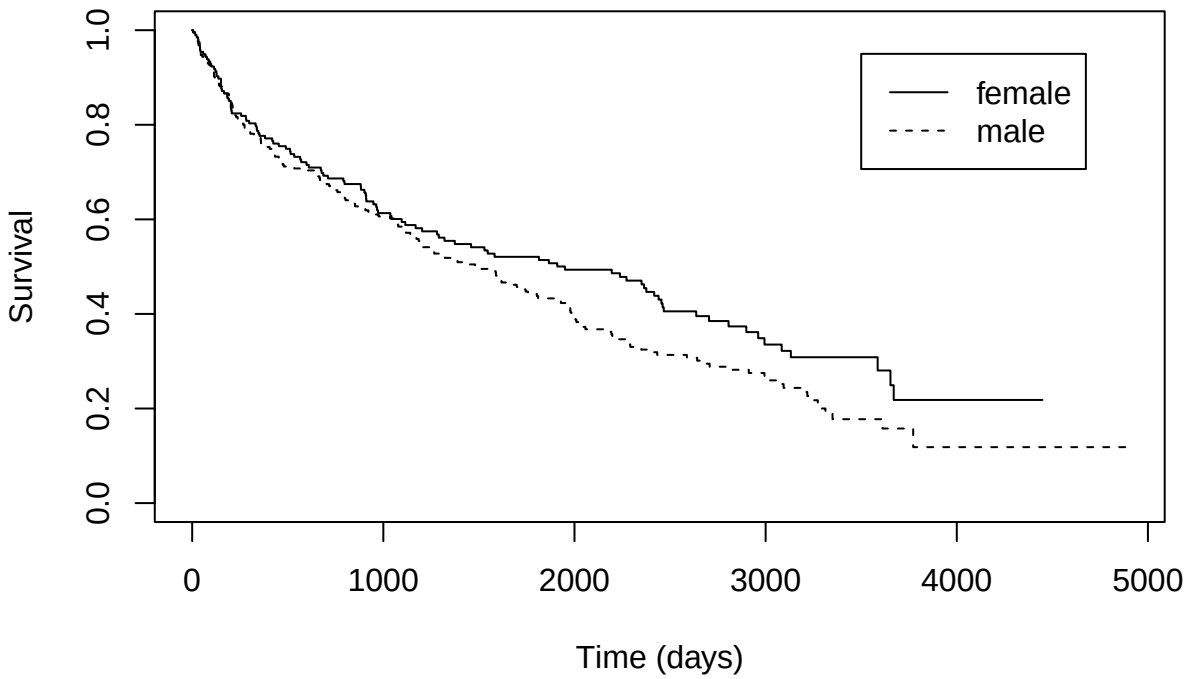


As including marks for censored patients in the plot makes the plot harder to read, I dont include the censoring marks.
From the plot we can see that until day ~1000, the groups behave similar. Maybe the prednisone has a slightly larger death rate in this period. After day ~1000 we can see a change; the patients given the placebo die quicker than the prednisone group. This lasts until day ~3300. After this day the groups behave similar again. After day ~4000, no patients die anymore, regardless of group. At the end of the study 20% of the patients are still included in the trial. We can also see that the placebo group has a slightly smaller median. This, together with the overall plot, indicates that there is a difference in effect of the prednisole and the placebo.

Kaplan-Meier Plot for gender comparison:

```
surv_sex = survfit(Surv(cirr$time, cirr$status==1)~cirr$sex, conf.type="plain")
plot(surv_sex, lty=1:2,xlab="Time (days)", ylab="Survival")
```

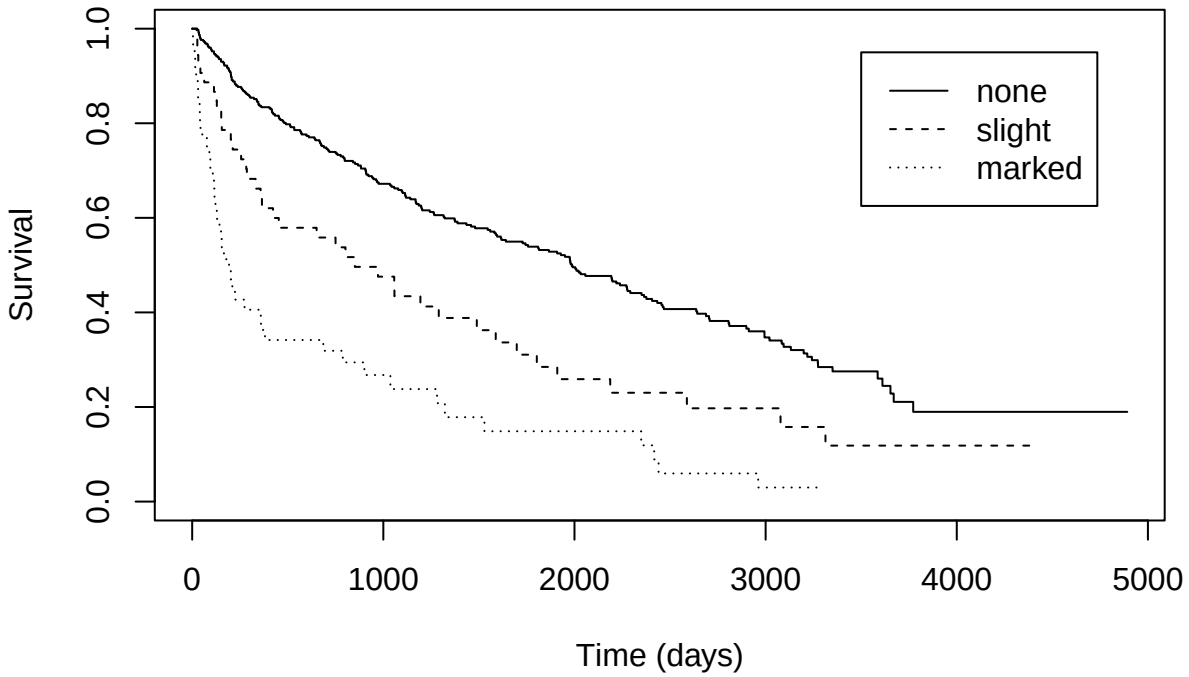
```
legend(3500,0.95,c("female","male"), lty=1:2)
```



From the plot, we can see that male patients tend to die earlier/quicker than female patients. Their median time differ by approximately ~500 days, with the female group reporting a higher median. In the start of the study, the groups behave similar until day ~1000, but diverge after this. The reasons for this difference can be many. Either that more females were included in the prednisole group, or that they were represented differently in the age groups. So it could be a coincidence that the females survive longer.

Kaplan-Meier Plot for ascites status comparison:

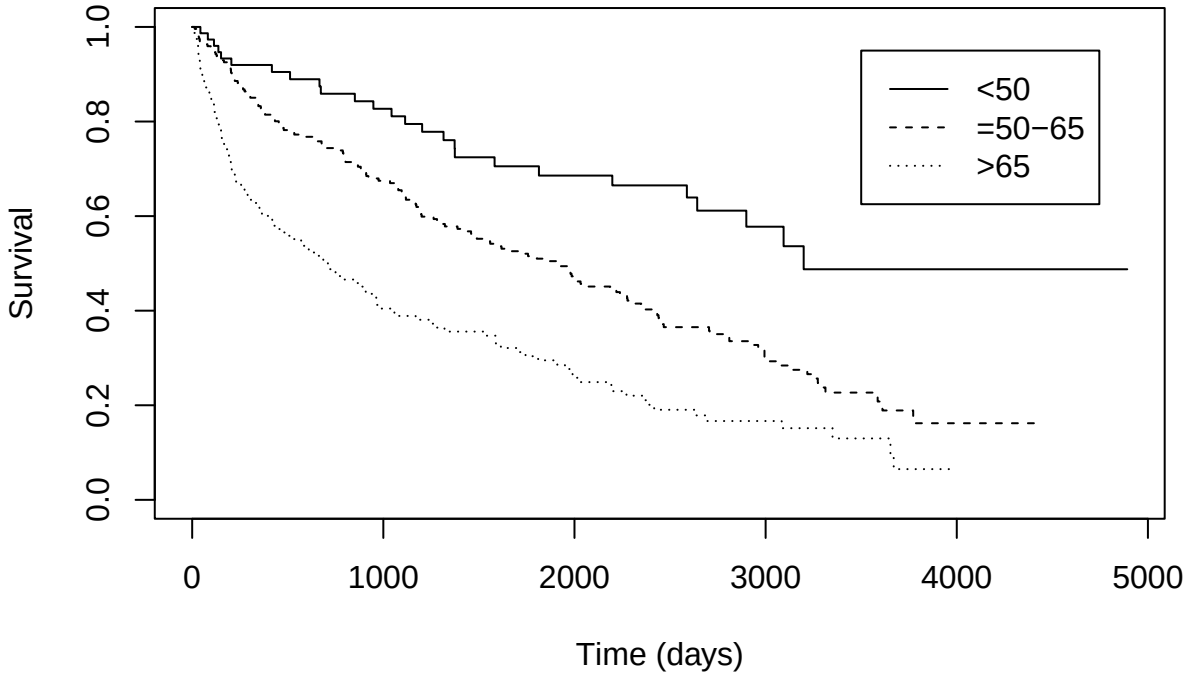
```
surv_asc = survfit(Surv(cirr$time, cirr$status==1)~cirr$asc, conf.type="plain")
plot(surv_asc, lty=1:3,xlab="Time (days)", ylab="Survival")
legend(3500,0.95,c("none","slight", "marked"), lty=1:3)
```



As can be expected, patients with no ascites in the start of the treatment survive longer than both patients with a slight or marked ascites. Patients with only a slight ascites at the start of the treatment also survive longer than patients in the "marked" group. Based on the plot, I would also assume that the groups have significant different medians.

Kaplan-Meier Plot for age comparison:

```
surv_agegr = survfit(Surv(cirr$time, cirr$status==1)~cirr$agegr, conf.type="plain")
plot(surv_agegr, lty=1:3,xlab="Time (days)", ylab="Survival")
legend(3500,0.95,c("<50", "=50-65", ">65"), lty=1:3)
```



From the plot, we can see that younger patients survive considerably longer than elderly people(>65). This could be because of different reasons. for example could the younger patients ascites be less developed at the

start of the treatment than the older patients' ascites. Or the older patients have a weaker health in general. As could be expected, the second group's (50-65 y) survival rate lies in between the other two groups. Finally, we can also see that not only is the death rate slower in the youngest group, but they also have a (maybe significantly) higher number of survivors at the end of the experiment than the other groups (50% vs 20% and 10% in the other two groups).

b)

I perform the log-rank test for each of the covariates:

For treatment:

```
survdiff(Surv(cirr$time, cirr$status==1)~cirr$treat)
```

```
## Call:
## survdiff(formula = Surv(cirr$time, cirr$status == 1) ~ cirr$treat)
##
##                 N Observed Expected (O-E)^2/E (O-E)^2/V
## cirr$treat=0 251      142      149     0.355     0.728
## cirr$treat=1 237      150      143     0.371     0.728
##
##  Chisq= 0.7  on 1 degrees of freedom, p= 0.4
```

For gender:

```
survdiff(Surv(cirr$time, cirr$status==1)~cirr$sex)
```

```
## Call:
## survdiff(formula = Surv(cirr$time, cirr$status == 1) ~ cirr$sex)
##
##              N Observed Expected (O-E)^2/E (O-E)^2/V
## cirr$sex=0 198      111      127      2.00      3.55
## cirr$sex=1 290      181      165      1.54      3.55
##
##  Chisq= 3.5  on 1 degrees of freedom, p= 0.06
```

For ascites status:

```
survdiff(Surv(cirr$time, cirr$status==1)~cirr$asc)
```

```
## Call:
## survdiff(formula = Surv(cirr$time, cirr$status == 1) ~ cirr$asc)
##
##              N Observed Expected (O-E)^2/E (O-E)^2/V
## cirr$asc=0 386      211    251.9      6.63     48.66
## cirr$asc=1  54       39     26.2      6.30      6.94
## cirr$asc=2  48       42     14.0     56.17     59.60
##
##  Chisq= 69.9  on 2 degrees of freedom, p= 7e-16
```

For agegroup:

```
survdiff(Surv(cirr$time, cirr$status==1)~cirr$agegr)
```

```
## Call:
## survdiff(formula = Surv(cirr$time, cirr$status == 1) ~ cirr$agegr)
```

```
##
##                N Observed Expected (O-E)^2/E (O-E)^2/V
## cirr$agegr=1  80       26     58.7     18.18     22.87
## cirr$agegr=2 250      148    162.0      1.21      2.72
## cirr$agegr=3 158      118     71.3     30.51     40.87
##
##  Chisq= 50.6  on 2 degrees of freedom, p= 1e-11
```

To my surprise, there was no significant difference in effect of the placebo and prednisone (p-value = 0.4). Also, as expected, the gender does not have a significant effect on the survival outcome (p-value = 0.06).

Both the ascites status and the patients' age seem to have a significant effect on the patients' survival. Both of the log-rank tests returned a p-value « 0.05. This could already be assumed based on the Kaplan-Meier plots. In the ascites status test, the group with "marked" status at the start of the treatment stands especially out, with a considerably higher difference in observed and expected deaths relative to the other groups. In the youngest group (<50) in the age test, less deaths are observed than expected. Also, more deaths than expected were observed in the oldest age group (>65). All of these observations seem reasonable.

c)

Constructing the Cox regression model:

```
fit_cirr = coxph(Surv(cirr$time, cirr$status==1)~cirr$treat+cirr$sex+cirr$asc+cirr$age)
summary(fit_cirr)
```

```
## Call:
## coxph(formula = Surv(cirr$time, cirr$status == 1) ~ cirr$treat +
##     cirr$sex + cirr$asc + cirr$age)
##
##   n= 488, number of events= 292
##
##                 coef exp(coef) se(coef)     z Pr(>|z|)
## cirr$treat1 0.044818  1.045837 0.117657 0.381 0.703263
## cirr$sex1   0.461877  1.587050 0.125631 3.676 0.000236 ***
## cirr$asc1   0.603507  1.828520 0.175019 3.448 0.000564 ***
## cirr$asc2   1.187254  3.278068 0.175224 6.776 1.24e-11 ***
## cirr$age    0.048877  1.050091 0.006844 7.141 9.26e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##             exp(coef) exp(-coef) lower .95 upper .95
## cirr$treat1     1.046     0.9562    0.8305     1.317
## cirr$sex1       1.587     0.6301    1.2407     2.030
## cirr$asc1       1.829     0.5469    1.2975     2.577
## cirr$asc2       3.278     0.3051    2.3252     4.621
## cirr$age        1.050     0.9523    1.0361     1.064
##
## Concordance= 0.682  (se = 0.017 )
## Likelihood ratio test= 109.3  on 5 df,   p=<2e-16
## Wald test            = 115.4  on 5 df,   p=<2e-16
## Score (logrank) test = 123.9  on 5 df,   p=<2e-16
```

When finding the hazard ratio for men relative to women, we want to compare the proportional hazard model, or cox regression model, for men(x=1) vs women(x=0) while holding all other covariates constant: $\frac{h(t|x_1,\ldots,x_5=1)}{h(t|x_1,\ldots,x_5=0)} = \exp(\beta_5)$.

So when finding the 95% confidence interval for the hazard ratio for men versus women, the 95% confidence interval of $\exp(\beta_5)$ is calculated. It has already been calculated as part of the cox regression output, but to make it clearer I have calulated it below using the confint function:

```
exp(confint(fit_cirr))[5,]
```

```
##    2.5 %   97.5 %
## 1.036098 1.064273
```

TYD MODELLEN KOMMENTER: interpret model and conclude on the effect on prednisone

Exercise 4:

Importing the dataset and libraries:

```
library(lme4)
```

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

```
library(nlme)
```

```
##
## Attaching package: 'nlme'
```

```
## The following object is masked from 'package:lme4':
##
##     lmList
```

```
data(sleepstudy)
```

a)

Approach 2:

```
day0 = sleepstudy$Reaction[sleepstudy$Days==0]
day9 = sleepstudy$Reaction[sleepstudy$Days==9]
diff = day9-day0
t.test(diff)
```

```
##
##  One Sample t-test
##
## data:  diff
## t = 6.9576, df = 17, p-value = 2.311e-06
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##   65.63465 122.76419
## sample estimates:
## mean of x
##  94.19942
```

b)

Approach 3: Fixed effect

```
fit_fixed = lm(Reaction~Days + factor(Subject), data=sleepstudy)
summary(fit_fixed)
```

```
##
## Call:
## lm(formula = Reaction ~ Days + factor(Subject), data = sleepstudy)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -100.540  -16.389   -0.341   15.215  131.159
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)         295.0310    10.4471  28.240  < 2e-16 ***
## Days                 10.4673     0.8042  13.015  < 2e-16 ***
## factor(Subject)309 -126.9009    13.8597  -9.156 2.35e-16 ***
## factor(Subject)310 -111.1326    13.8597  -8.018 2.07e-13 ***
## factor(Subject)330  -38.9124    13.8597  -2.808 0.005609 **
## factor(Subject)331  -32.6978    13.8597  -2.359 0.019514 *
## factor(Subject)332  -34.8318    13.8597  -2.513 0.012949 *
## factor(Subject)333  -25.9755    13.8597  -1.874 0.062718 .
## factor(Subject)334  -46.8318    13.8597  -3.379 0.000913 ***
## factor(Subject)335  -92.0638    13.8597  -6.643 4.51e-10 ***
## factor(Subject)337   33.5872    13.8597   2.423 0.016486 *
## factor(Subject)349  -66.2994    13.8597  -4.784 3.87e-06 ***
## factor(Subject)350  -28.5312    13.8597  -2.059 0.041147 *
## factor(Subject)351  -52.0361    13.8597  -3.754 0.000242 ***
## factor(Subject)352   -4.7123    13.8597  -0.340 0.734300
## factor(Subject)369  -36.0992    13.8597  -2.605 0.010059 *
## factor(Subject)370  -50.4321    13.8597  -3.639 0.000369 ***
## factor(Subject)371  -47.1498    13.8597  -3.402 0.000844 ***
## factor(Subject)372  -24.2477    13.8597  -1.750 0.082108 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 30.99 on 161 degrees of freedom
## Multiple R-squared:  0.7277, Adjusted R-squared:  0.6973
## F-statistic: 23.91 on 18 and 161 DF,  p-value: < 2.2e-16
```

```
anova(fit_fixed)
```

```
## Analysis of Variance Table
##
## Response: Reaction
##                  Df Sum Sq Mean Sq F value    Pr(>F)
## Days              1 162703  162703 169.401 < 2.2e-16 ***
## factor(Subject)  17 250618   14742  15.349 < 2.2e-16 ***
## Residuals       161 154634     960
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

c)

Approach 4: Random effects model

```
fit_random = lme(Reaction~Days, random=~1|Subject, data=sleepstudy)
summary(fit_random)
```

```
## Linear mixed-effects model fit by REML
```

```
##   Data: sleepstudy
##        AIC      BIC    logLik
##   1794.465 1807.192 -893.2325
##
## Random effects:
##  Formula: ~1 | Subject
##         (Intercept) Residual
## StdDev:    37.12383 30.99123
##
## Fixed effects:  Reaction ~ Days
##                 Value Std.Error  DF  t-value p-value
## (Intercept) 251.40510  9.746716 161 25.79383       0
## Days         10.46729  0.804221 161 13.01543       0
##  Correlation:
##      (Intr)
## Days -0.371
##
## Standardized Within-Group Residuals:
##        Min         Q1        Med         Q3        Max
## -3.2256707 -0.5528788  0.0108521  0.5187971  4.2506162
##
## Number of Observations: 180
## Number of Groups: 18
```

```
sd_intercept = 37.12383
sigma = fit_random$sigma

corr_reaction = sd_intercept^2/(sd_intercept^2+sigma^2)
print(corr_reaction)
```

```
## [1] 0.589309
```