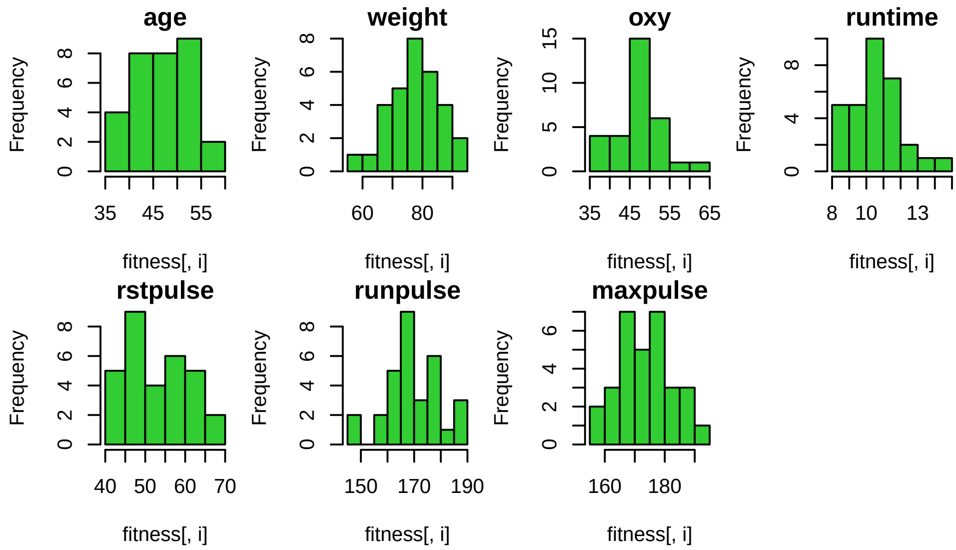
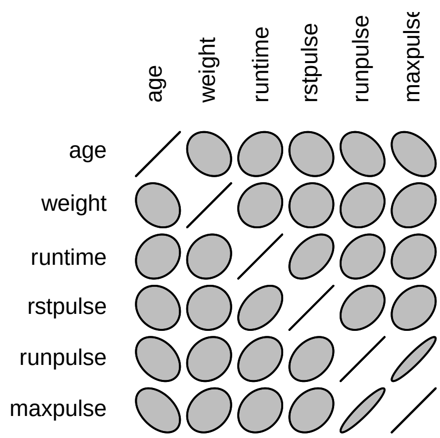
Solution to Series 6

1. a)
```
> ## load data
> load("fitness.rda")
> ## analyze the variables
> par(mfrow=c(2,4))
> for (i in 1:7) hist(fitness[,i], col="limegreen", main=names(fitness)[i])
```



As you can see from the histograms, there are no variables that are strongly skewed to the right and/or have a relative scale with a large range of values. Therefore, we will not apply any transformations and there are no other apparent issues.

```
> par(mfrow=c(1,1))
> library(ellipse)
> plotcorr(cor(fitness[,-3]), cex.lab = 0.75, mar = c(1,1,1,1))
```

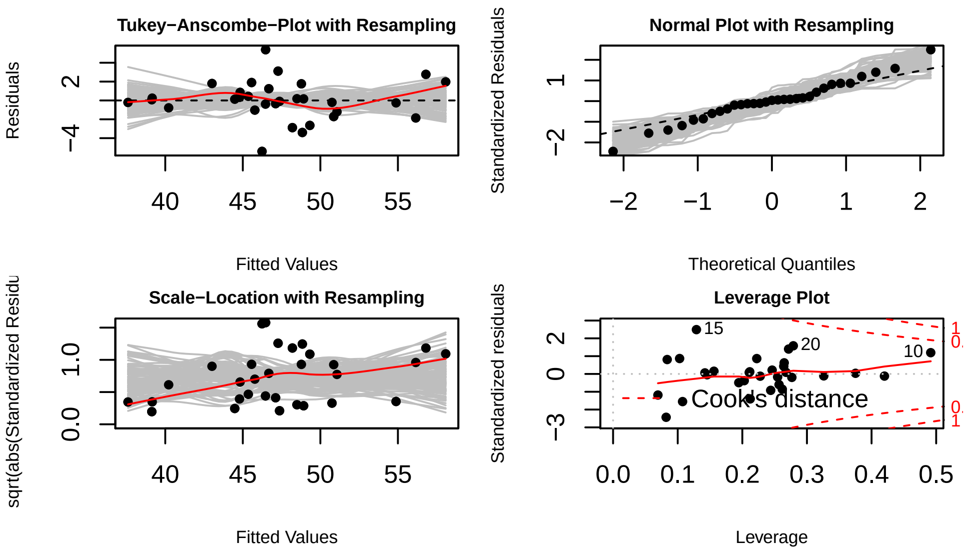


The analysis of the pairwise correlations should be done without the target variable. As we can see from the above plot, there is a strong positive correlation between the running pulse and the maximal pulse. The remaining variables do not show strong pairwise correlations.

b)
```
> ## fit model
> fit <- lm(oxy ~ ., data=fitness)
> summary(fit)
```

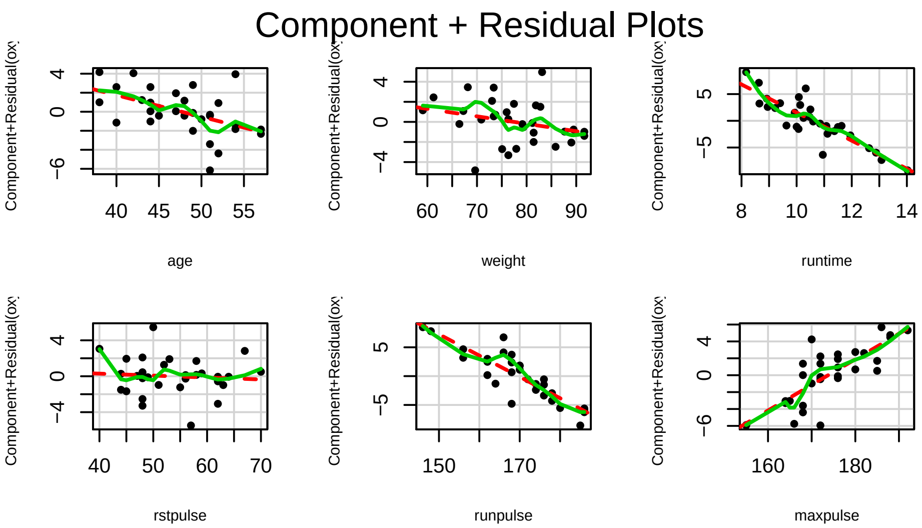
```
Call:
lm(formula = oxy ~ ., data = fitness)

Residuals:
    Min      1Q  Median      3Q     Max 
-5.4026 -0.8991  0.0706  1.0496  5.3847 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) 102.93448   12.40326   8.299 1.64e-08 ***
age          -0.22697    0.09984  -2.273  0.03224 *  
weight       -0.07418    0.05459  -1.359  0.18687    
runtime      -2.62865    0.38456  -6.835 4.54e-07 ***
rstpulse     -0.02153    0.06605  -0.326  0.74725    
runpulse     -0.36963    0.11985  -3.084  0.00508 ** 
maxpulse      0.30322    0.13650   2.221  0.03601 *  
---
Signif. codes:  
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.317 on 24 degrees of freedom
Multiple R-squared:  0.8487,	Adjusted R-squared:  0.8108 
F-statistic: 22.43 on 6 and 24 DF,  p-value: 9.715e-09
```

c)
```
> ## fit model
> fit <- lm(oxy ~ ., data=fitness)
> source("resplot.R")
> resplot(fit)
```



There are no systematic errors. There are two large residuals, one of which is positive, the other one is negative. The assumption of constant variance is at the border of being satisfied. So while the residual plots do not look perfect, the model assumptions seem to be fulfilled to a sufficient degree.

```
> ## partial residual plots
> library(car)
> crPlots(fit, pch=20, layout = c(2,3), cex.lab = .75)
```



The two observations with the large residuals cause deviations in the partial residual plots. In this case, however, we would not diagnose the presence of a systematic deviation. Therefore, we conclude that the predictors have been entered into the model in the correct form.

d)

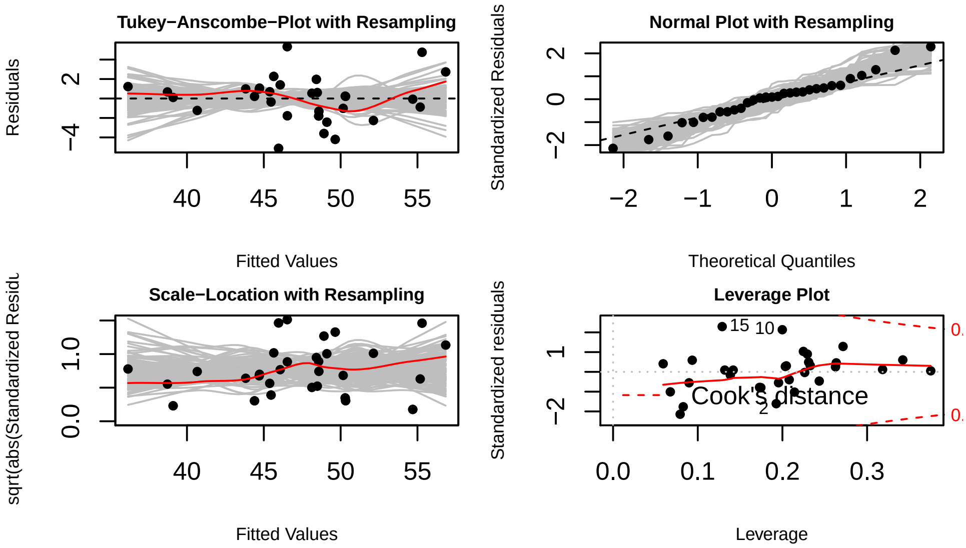
```
> ## multicollinearity
> library(faraway)
> vif(fit)
```

```
     age   weight  runtime rstpulse runpulse maxpulse
1.512836 1.155329 1.590868 1.415589 8.437274 8.743848
```

The VIFs of `runpulse` and `maxpulse` indicate the presence of critical multicollinearity. This is not surprising given the large pairwise correlation between running pulse and maximal pulse.

e) **(i) Amputation**

```
> ## fitted values
> f.o <- fitted(fit)
> ## Amputation - leave out maxpulse
> fit <- lm(oxy ~ age + weight + runtime + rstpulse + runpulse, data=fitness)
> resplot(fit)
```

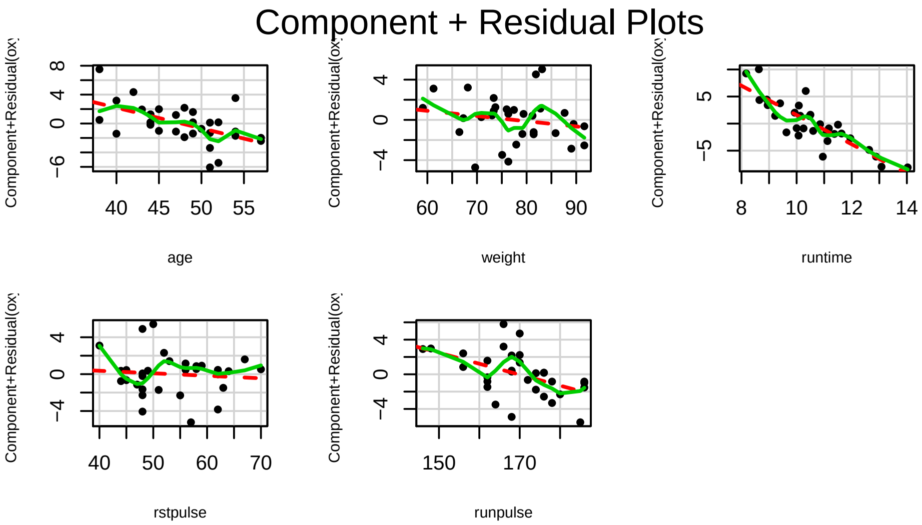


Since the high multicollinearity stems from the large pairwise correlation between running pulse and maximal pulse, one of these two variables should be excluded from the model. We recommend to leave out the maximal pulse due to background knowledge.

```
> crPlots(fit, pch=20, layout = c(2,3), cex.lab = .75)
> vif(fit)
```

```
     age   weight  runtime rstpulse runpulse
1.408289 1.116150 1.578518 1.413545 1.388799
```

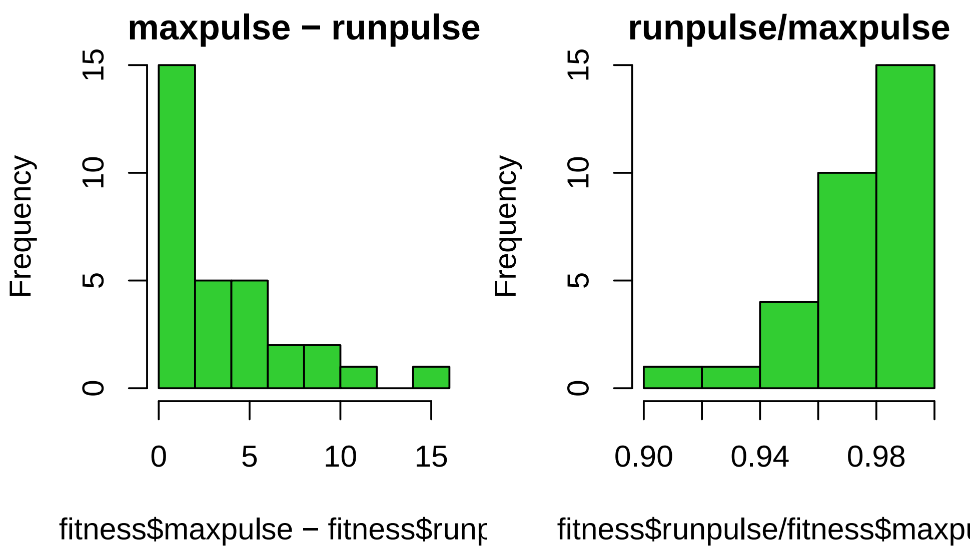
```
> f.i <- fitted(fit)
```



The resulting model does not show a systematic error, the predictors seem to be in the correct form and there is no high multicollinearity.

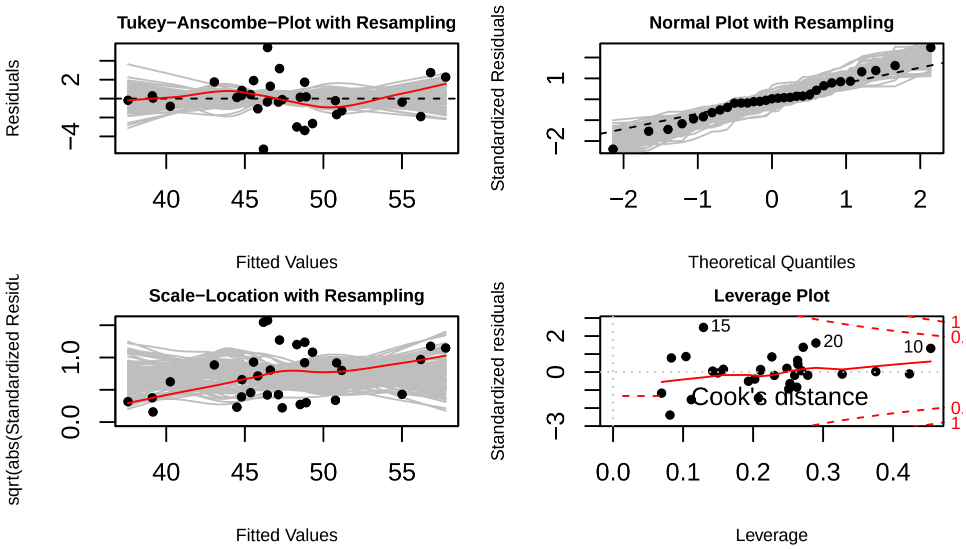
(ii) Creating new variables

```
> ## transformation
> par(mfrow=c(1,2))
> hist(fitness$maxpulse-fitness$runpulse, col="limegreen", main = "maxpulse - runpulse")
> hist(fitness$runpulse/fitness$maxpulse, col="limegreen", main = "runpulse/maxpulse")
> my.fitness <- fitness[,-7]
> my.fitness$intensity <- fitness$runpulse/fitness$maxpulse
```



Either `runpulse` or `maxpulse` needs to be adjusted. We leave the running pulse in the model and substitute the maximal pulse by either `maxpulse-runpulse` or `runpulse/maxpulse`. Since the latter shows less skew in the histogram, we choose to use the quotient.

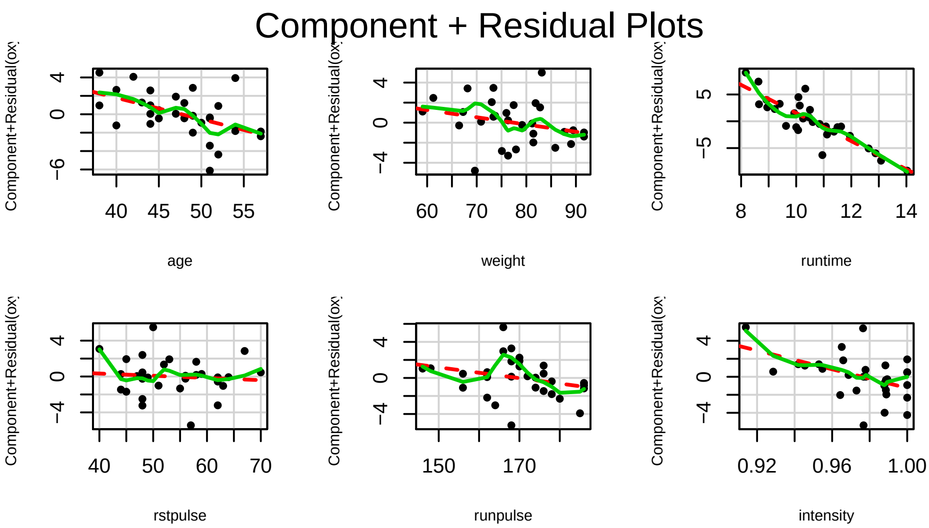
```
> fit <- lm(oxy ~ ., data=my.fitness)
> resplot(fit)
```




```
> crPlots(fit, pch=20, layout = c(2,3), cex.lab = .75)
> vif(fit)
```

```
     age    weight   runtime  rstpulse  runpulse intensity
1.500884  1.152036  1.594347  1.414005  1.961997  1.615894
```

```
> f.ii <- fitted(fit)
```



(iii) Ridge regression

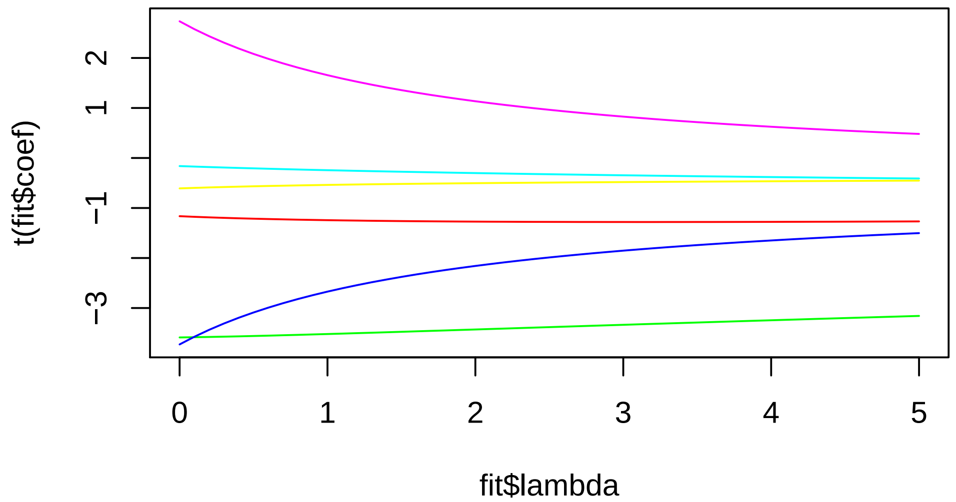
```
> ## ridge regression
> library(MASS)
> fit  <- lm.ridge(oxy ~ ., data=fitness, lambda=seq(0,5, by=0.1))
> select(fit)
```

```
modified HKB estimator is 0.5966403
modified L-W estimator is 0.9212768
smallest value of GCV  at 0.5
```

First we need to estimate the penalty parameter λ. Note that the algorithm allows to estimate the ridge regression parameters for many values of λ simultaneously. Subsequently, the optimal value is determined via Generalized Cross Validation (GCV). Here, it is 0.5.

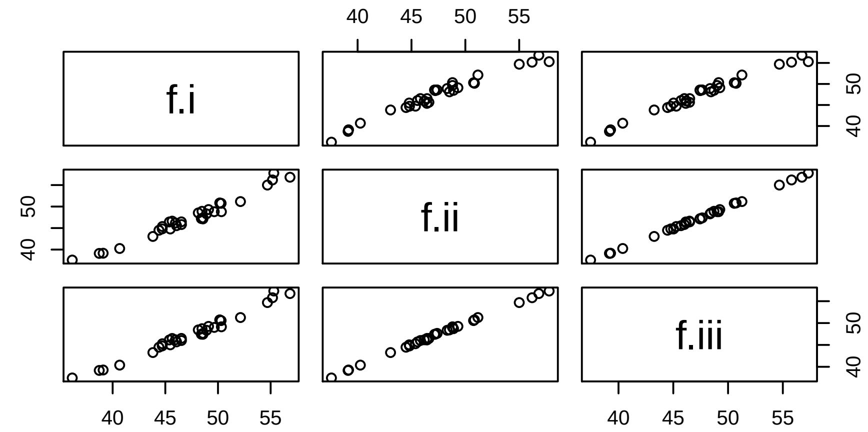
As the function does not allow us to extract the fitted values, these need to be computed from the available output. The design matrix is available with the command `model.matrix()` when performing an OLS regression. The estimated coefficients can be obtained from the ridge regression output. Finally, multiplying these two quantities yields the fitted values.

```
> matplot(fit$lambda, t(fit$coef), lty=1, type="l", col=rainbow(6))
> fit  <- lm.ridge(oxy ~ ., data=fitness, lambda=0.5)
> fit
                    age       weight      runtime
104.93110913  -0.23671702  -0.06904551  -2.60778598
    rstpulse     runpulse     maxpulse
 -0.02763130  -0.30629734   0.23089222
> mm   <- model.matrix(lm(oxy ~ ., data=fitness))
> f.iii <- mm%*%coef(fit)
```



Pairwise comparison of fitted values

```
> ## comparison of the fitted values
> df <- data.frame(f.i, f.ii, f.iii)
> pairs(df)
```



When using amputation the fitted values are notably different from the other approaches. This indicates that we do lose some precision when excluding a variable from the set of predictors. The other two approaches yield very similar fitted values – even though the approaches are very different from a theoretical perspective, they do yield almost identical results.

f)
```
> ## backward elimination using the p-values
> fit <- lm(oxy ~ ., data=my.fitness)
> drop1(fit, test="F")
Single term deletions
```

```
Model:
oxy ~ age + weight + runtime + rstpulse + runpulse + intensity
          Df Sum of Sq    RSS    AIC F value    Pr(>F)
<none>                 131.09 58.698
age        1    29.417 160.50 62.974  5.3860   0.02912 *
weight     1     9.440 140.53 58.853  1.7283   0.20105
runtime    1   249.086 380.17 89.706 45.6045 5.516e-07 ***
rstpulse   1     0.744 131.83 56.873  0.1362   0.71530
runpulse   1     5.764 136.85 58.032  1.0554   0.31452
intensity  1    24.244 155.33 61.958  4.4388   0.04577 *
---
Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
> ## remove rstpulse
> fit <- update(fit, .~.-rstpulse)
> drop1(fit, test="F")
```

```
Single term deletions

Model:
oxy ~ age + weight + runtime + runpulse + intensity
          Df Sum of Sq    RSS    AIC F value   Pr(>F)
<none>                 131.83 56.873
age        1     28.79 160.62 60.997  5.4604  0.02776 *
weight     1      8.98 140.81 56.915  1.7023  0.20387
runtime    1    320.50 452.33 93.093 60.7798 3.75e-08 ***
runpulse   1      6.36 138.18 56.333  1.2052  0.28275
intensity  1     24.41 156.23 60.139  4.6283  0.04130 *
---
Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
> ## remove runpulse
> fit <- update(fit, .~.-runpulse)
> drop1(fit, test="F")
```

```
Single term deletions

Model:
oxy ~ age + weight + runtime + intensity
          Df Sum of Sq    RSS    AIC F value    Pr(>F)
<none>                 138.18 56.333
age        1     22.44 160.62 58.997  4.2220  0.050081 .
weight     1     11.19 149.37 56.746  2.1051  0.158771
runtime    1    370.78 508.97 94.750 69.7648 7.806e-09 ***
intensity  1     56.93 195.11 65.027 10.7109  0.003006 **
---
Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
> ## remove weight
> fit <- update(fit, .~.-weight)
> drop1(fit, test="F")
```

```
Single term deletions

Model:
oxy ~ age + runtime + intensity
          Df Sum of Sq    RSS    AIC F value    Pr(>F)
<none>                 149.37 56.746
age        1     15.92 165.29 57.885  2.8773  0.101336
runtime    1    422.45 571.82 96.360 76.3609 2.357e-09 ***
```

```
intensity  1     51.34 200.72 63.905  9.2807  0.005126 **
---
Signif. codes:
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

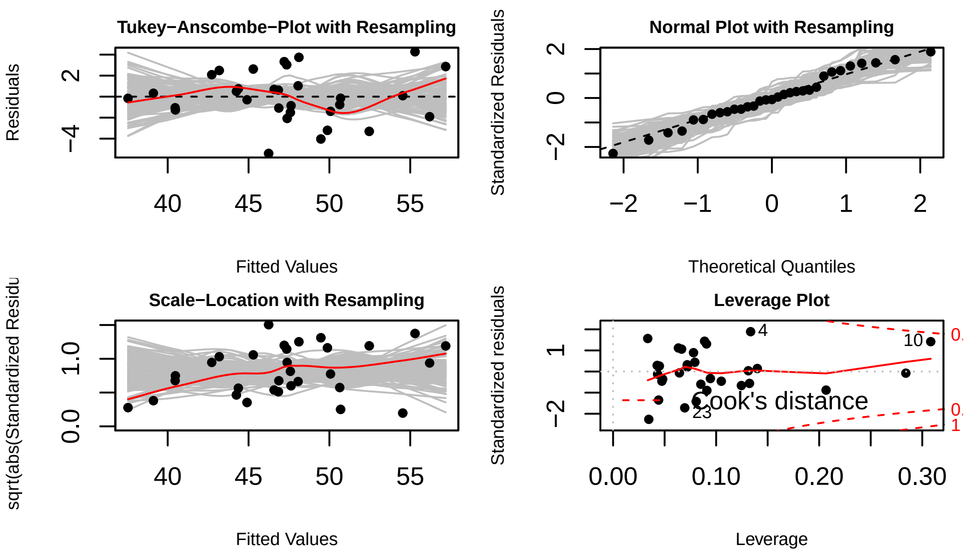
```
> ## remove age
> fit <- update(fit, .~.-age)
> drop1(fit, test="F")
```

```
Single term deletions

Model:
oxy ~ runtime + intensity
          Df Sum of Sq    RSS    AIC F value    Pr(>F)
<none>                 165.29 57.885
runtime    1    464.20 629.49 97.339 78.6348 1.274e-09 ***
intensity  1     53.19 218.48 64.534  9.0105  0.005593 **
---
Signif. codes:
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> ## final model
> summary(fit)
```

```
Call:
lm(formula = oxy ~ runtime + intensity, data = my.fitness)

Residuals:
   Min     1Q Median     3Q    Max
-5.407 -1.334 -0.148  1.557  4.273

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 141.4228    19.9496   7.089 1.03e-07 ***
runtime      -2.9896     0.3371  -8.868 1.27e-09 ***
intensity   -63.9305    21.2978  -3.002  0.00559 **
---
Signif. codes:
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.43 on 28 degrees of freedom
Multiple R-squared:  0.8059,	Adjusted R-squared:  0.792
F-statistic: 58.11 on 2 and 28 DF,  p-value: 1.081e-10
```

```
> resplot(fit)
```



Sequentially, the variables `rstpulse`, `runpulse`, `weight` and `age` are excluded from the model. Finally, only `runtime` and `intensity` (the quotient of `runpulse` and `maxpulse`) are left in the model.

g) According to our results, the rate of oxygen consumption could be modeled with the variables `running time` and `intensity`. This yields an R-squared value around 0.8, i.e. the rate of oxygen consumption can be explained well but not perfectly. It is difficult to tell whether this is sufficient for practical purposes and cannot be concluded based on our results only. The trade-off between costs and loss of precision would need to be assessed further.

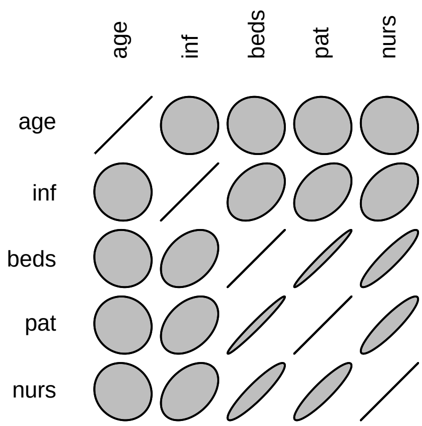
2.
```
> load("senic.rda")
> senic <- senic[,c("length", "age", "inf", "region", "beds", "pat", "nurs")]
```

a) We check the correlations between the continuous predictors:

```
> indices_categorical_vars <- which(is.element(colnames(senic), c("length", "region")))
> cor(senic[, -indices_categorical_vars])
              age          inf        beds         pat
age   1.000000000 -0.006266807 -0.05882316 -0.05477467
inf  -0.006266807  1.000000000  0.36917855  0.39070521
beds -0.058823160  0.369178549  1.00000000  0.98099774
pat  -0.054774667  0.390705214  0.98099774  1.00000000
nurs -0.082944616  0.402911390  0.91550415  0.90789698
            nurs
age  -0.08294462
inf   0.40291139
beds  0.91550415
pat   0.90789698
nurs  1.00000000
```

Graphical illustration of the correlations:

```
>      library(ellipse)
>      plotcorr(cor(senic[, -indices_categorical_vars]), cex.lab = 0.75, mar = c(1,1,1,1))
```



We see that `beds`, `pat` and `nurs` are strongly correlated. We expected this because they all can be seen as measures of the size of a hospital. We will leave the variable `pat` unmodified because it is definitely a key factor to take into account when `length` is the response variable. We change the others to solve the high-correlation problem without having to take them out of the model. For this, we will substitute `beds` by `pat/beds` and `nurs` by `pat/nurs`.
Before combining the variables, we check if `beds` and `nurs` contain zeroes:

```
> any(senic$beds == 0)
[1] FALSE
> any(senic$nurs == 0)
```

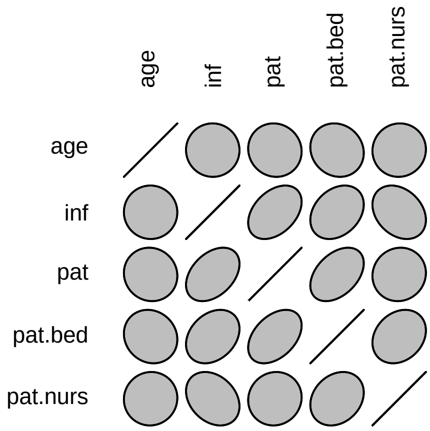
```
[1] FALSE
```

Now we combine the variables and check the correlations again.

```
> senic.02 <- data.frame(length=senic$length, age=senic$age, inf=senic$inf,
 region=senic$region, pat=senic$pat, pat.bed=senic$pat/senic$beds,
 pat.nurs=senic$pat/senic$nurs)
> cor(senic.02[,-indices_categorical_vars])
                  age          inf         pat    pat.bed
age       1.000000000 -0.006266807 -0.05477467 -0.1096058
inf      -0.006266807  1.000000000  0.39070521  0.2897338
pat      -0.054774667  0.390705214  1.00000000  0.4151079
pat.bed  -0.109605797  0.289733778  0.41510791  1.0000000
pat.nurs  0.026954588 -0.285984796  0.05659985  0.2289331
            pat.nurs
age       0.02695459
inf      -0.28598480
pat       0.05659985
pat.bed   0.22893307
pat.nurs  1.00000000
```

Graphical illustration of the correlations after modifying some variables:

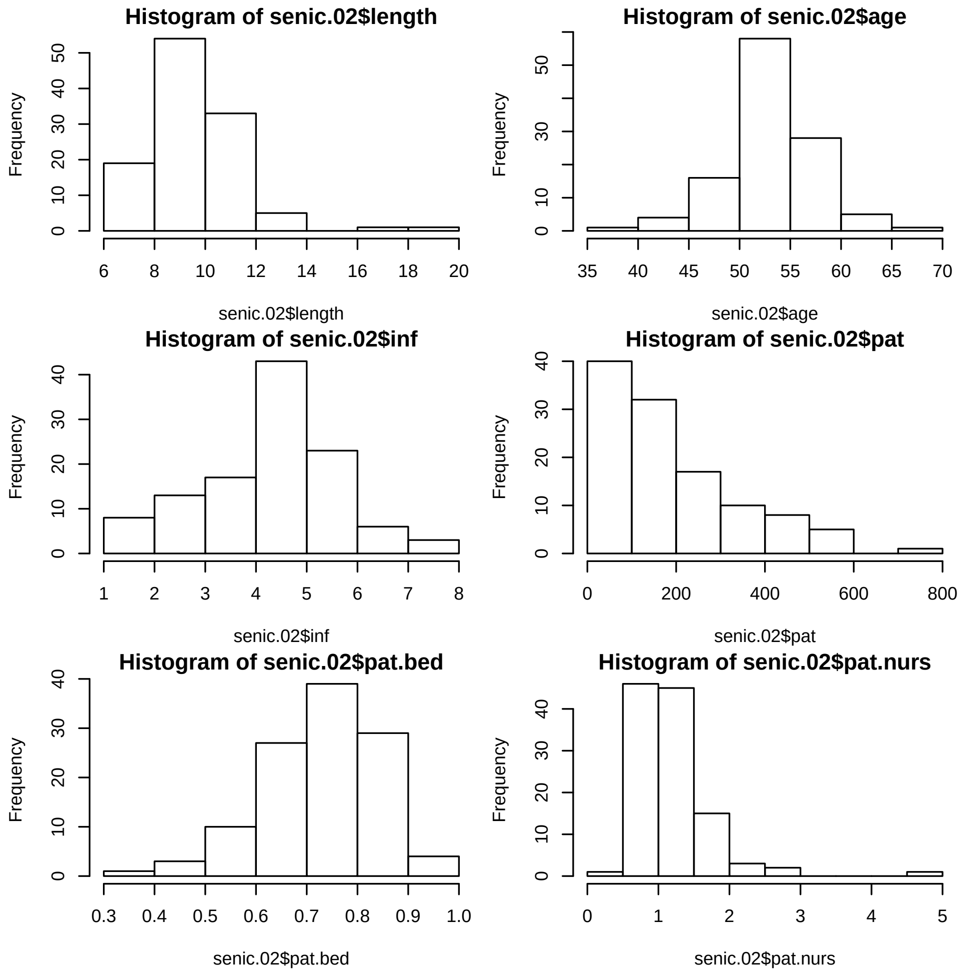
```
>     plotcorr(cor(senic.02[,-indices_categorical_vars]), cex.lab = 0.75, mar = c(1,1,1,1))
```



The correlations were strongly reduced and we still have some information about the variables `beds` and `nurs`.

b) First, we take a look at the histogram of the predictors before doing transformations:

```
>     par(mfrow=c(3,2))
>     hist(senic.02$length)
>     hist(senic.02$age)
>     hist(senic.02$inf)
>     hist(senic.02$pat)
>     hist(senic.02$pat.bed)
>     hist(senic.02$pat.nurs)
```



The variables `length`, `pat` and `pat.nurs` need to be transformed. Moreover, we see that `pat.bed` is slightly left skewed. In this case, one would try to square or cube the variable to improve the situation. However, for the purpose of this question, we will not do it here and leave this as an exercise.

We check for zeroes in `pat` and `length`:

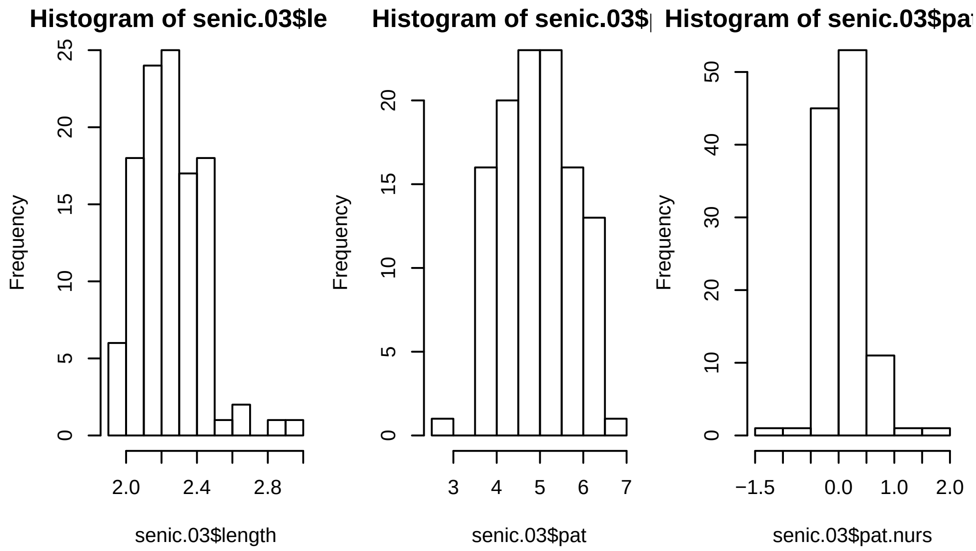
```
> any(senic.02$length == 0)
[1] FALSE
> any(senic.02$pat == 0)
[1] FALSE
```

Given that there are no zeroes in these variables, we are free to transform the predictors:

```
> senic.03 <- senic.02
> senic.03$length <- log(senic.02$length)
> senic.03$pat <- log(senic.02$pat)
> senic.03$pat.nurs <- log(senic.02$pat.nurs)
```

We look at the histograms again after applying the necessary transformations.

```
>     par(mfrow=c(1,3))
>     hist(senic.03$length)
>     hist(senic.03$pat)
>     hist(senic.03$pat.nurs)
```



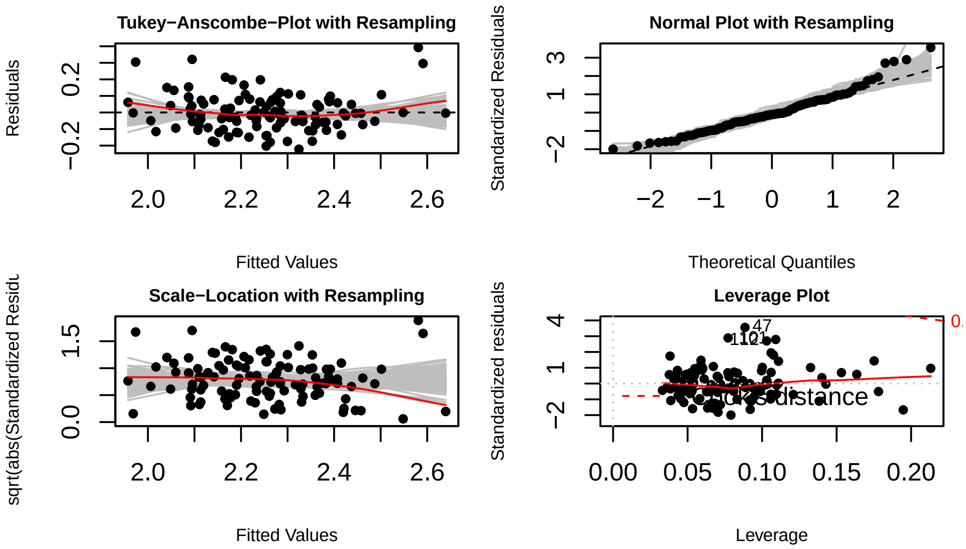
We see that the transformations improved the histograms.

c) We fit a linear regression:

```
>     fit.03 <- lm(length ~ age + inf + region + pat + pat.bed + pat.nurs, data=senic.03)
>     summary(fit.03)
```

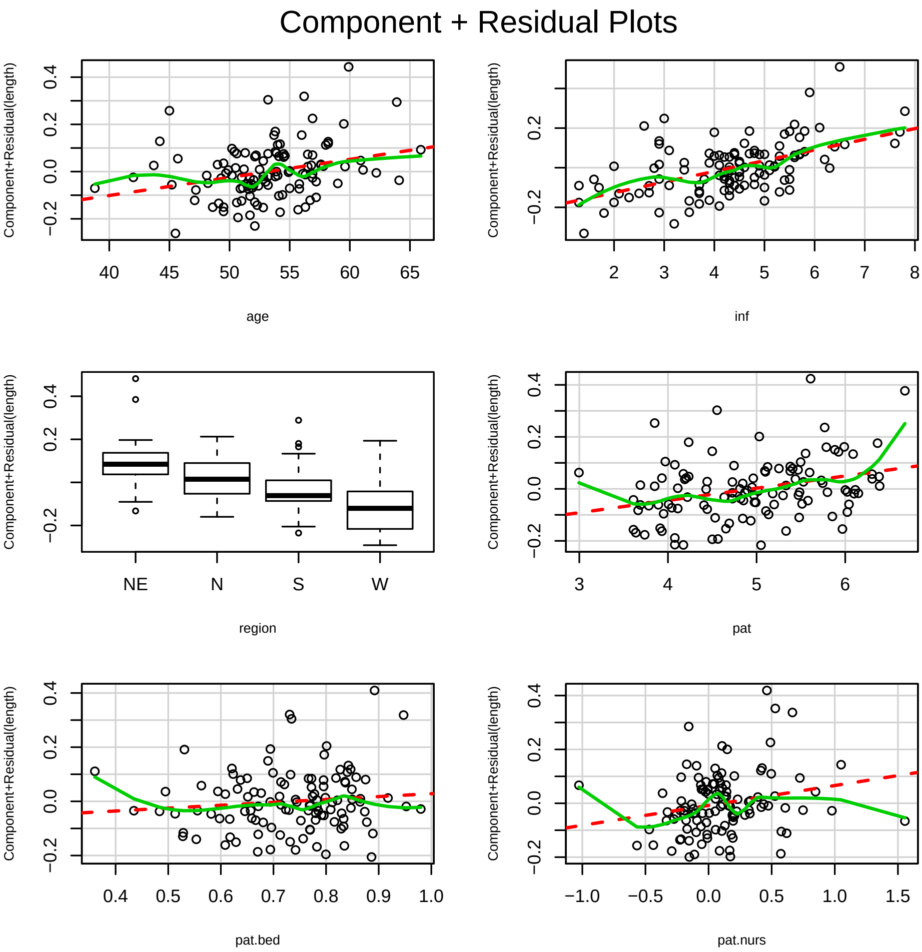
```
Call:
lm(formula = length ~ age + inf + region + pat + pat.bed + pat.nurs,
    data = senic.03)

Residuals:
     Min       1Q   Median       3Q      Max
-0.22160 -0.07198 -0.01166  0.06382  0.39264

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.379579   0.178646   7.722 7.39e-12 ***
age          0.007645   0.002551   2.997 0.003412 **
inf          0.053916   0.010312   5.228 8.88e-07 ***
regionN     -0.074073   0.031132  -2.379 0.019168 *
regionS     -0.121379   0.030443  -3.987 0.000125 ***
regionW     -0.200437   0.039882  -5.026 2.10e-06 ***
pat          0.047034   0.017795   2.643 0.009485 **
pat.bed      0.106392   0.124304   0.856 0.394020
pat.nurs     0.073836   0.037202   1.985 0.049808 *
---
Signif. codes:
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.1156 on 104 degrees of freedom
Multiple R-squared:  0.6081,       Adjusted R-squared:  0.578
F-statistic: 20.17 on 8 and 104 DF,  p-value: < 2.2e-16
```

```
>     par(mfrow=c(2,2))
>     source("../ex1/resplot.R")
>     resplot(fit.03)
```



From the summary we see that pat.bed is not statistically significant and a variable selection is necessary (see next question).
From the model diagnostics plots we note that there are three outliers, i.e. observations 47, 101, and 112. However, since their Cook's distance is below 0.5, they don't significantly influence our fit and we proceed with our analysis. The assumptions of linearity and constant variance seem to be satisfied. The QQ-plot does not look perfect but we can also accept the normality assumption.
Now we visualise our model with partial residual plots.

```
>    library(car)
>    crPlots(fit.03, cex.lab = .75)
```



As it can be seen in the plots, the predictor pat.bed don't have much explanatory power, and indeed, its p-value is also large.

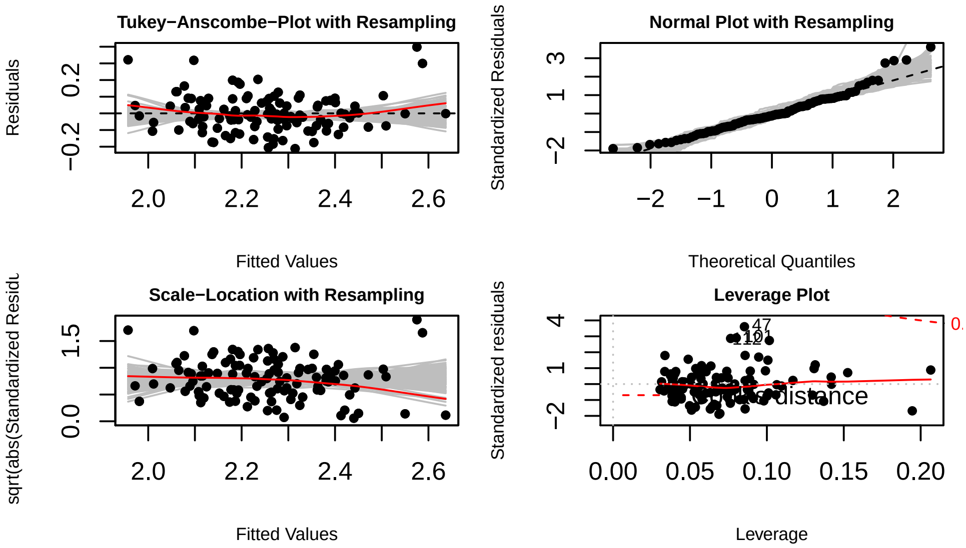
Now, we perform backwards elimination using `fit.03` as our starting model. We remove the variable `pat.bed`:

```
> fit.P1 <-  lm(length ~ age + inf + region + pat + pat.nurs, data=senic.03)
> summary(fit.P1)
```

```
Call:
lm(formula = length ~ age + inf + region + pat + pat.nurs, data = senic.03)

Residuals:
     Min       1Q   Median       3Q      Max
-0.21159 -0.07408 -0.01331  0.06479  0.39816

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.438983   0.164404   8.753 3.80e-14 ***
age          0.007404   0.002532   2.924  0.00424 **
inf          0.055360   0.010160   5.449 3.37e-07 ***
regionN     -0.078067   0.030741  -2.540  0.01257 *
regionS     -0.123516   0.030302  -4.076 8.92e-05 ***
regionW     -0.209690   0.038340  -5.469 3.08e-07 ***
pat          0.052614   0.016537   3.182  0.00193 **
pat.nurs     0.081985   0.035917   2.283  0.02447 *
---
Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1155 on 105 degrees of freedom
Multiple R-squared:  0.6053,     Adjusted R-squared:  0.579
F-statistic: 23.01 on 7 and 105 DF,  p-value: < 2.2e-16
```

Note that the F-statistic increased. Since the rest of the variables are statistically significant, `pat.bed` is the only predictor that is left out of the model.
Now we look at the residuals of model `fit.P1`:

```
> par(mfrow=c(2,2))
> resplot(fit.P1)
```



Model diagnostics plots look similar to the ones of the fit containing all predictors. The assumptions of linearity, constant variance and normality of the errors seem to be fine.

d) Backward elimination:

```
> fit.B <- step(fit.03, direction="backward")
```

```
Start:  AIC=-478.96
length ~ age + inf + region + pat + pat.bed + pat.nurs
```

```
           Df Sum of Sq    RSS     AIC
- pat.bed   1   0.00979 1.4000 -480.17
<none>                  1.3902 -478.96
- pat.nurs  1   0.05266 1.4429 -476.76
- pat       1   0.09339 1.4836 -473.62
- age       1   0.12006 1.5103 -471.60
- region    3   0.41062 1.8009 -455.72
- inf       1   0.36544 1.7557 -454.59

Step:  AIC=-480.17
length ~ age + inf + region + pat + pat.nurs

           Df Sum of Sq    RSS     AIC
<none>                  1.4000 -480.17
- pat.nurs  1   0.06947 1.4695 -476.70
- age       1   0.11399 1.5140 -473.33
- pat       1   0.13498 1.5350 -471.77
- inf       1   0.39587 1.7959 -454.03
- region    3   0.46502 1.8651 -453.76
> summary(fit.B)
Call:
lm(formula = length ~ age + inf + region + pat + pat.nurs, data = senic.03)

Residuals:
     Min       1Q   Median       3Q      Max
-0.21159 -0.07408 -0.01331  0.06479  0.39816

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.438983   0.164404   8.753 3.80e-14 ***
age          0.007404   0.002532   2.924  0.00424 **
inf          0.055360   0.010160   5.449 3.37e-07 ***
regionN     -0.078067   0.030741  -2.540  0.01257 *
regionS     -0.123516   0.030302  -4.076 8.92e-05 ***
regionW     -0.209690   0.038340  -5.469 3.08e-07 ***
pat          0.052614   0.016537   3.182  0.00193 **
pat.nurs     0.081985   0.035917   2.283  0.02447 *
---
Signif. codes:
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.1155 on 105 degrees of freedom
Multiple R-squared:  0.6053,       Adjusted R-squared:  0.579
F-statistic: 23.01 on 7 and 105 DF,  p-value: < 2.2e-16
```

The backward elimination using AIC only removes the variable pat.bed from the model, just as the backward elimination using the p-values did.

e) Forward selection:

```
> fit.empty <- lm(length ~ 1, data=senic.03)
>     scp   <- list(lower=~1, upper=~age + inf + region + pat + pat.bed + pat.nurs)
>     fit.F <- step(fit.empty, scope=scp, direction="forward")
Start:  AIC=-389.11
length ~ 1

           Df Sum of Sq    RSS     AIC
+ inf       1   1.08286 2.4646 -428.27
+ pat       1   0.94180 2.6057 -421.98
```

```
+ region    3   0.98268 2.5648 -419.76
+ pat.bed   1   0.69376 2.8537 -411.70
+ age       1   0.10368 3.4438 -390.46
+ pat.nurs  1   0.07906 3.4684 -389.66
<none>                  3.5475 -389.11

Step:  AIC=-428.27
length ~ inf

           Df Sum of Sq    RSS     AIC
+ region    3   0.71923 1.7454 -461.26
+ pat.bed   1   0.30829 2.1563 -441.37
+ pat       1   0.29591 2.1687 -440.72
+ pat.nurs  1   0.28973 2.1749 -440.40
+ age       1   0.10793 2.3567 -431.33
<none>                  2.4646 -428.27

Step:  AIC=-461.26
length ~ inf + region

           Df Sum of Sq    RSS     AIC
+ pat       1  0.151470 1.5939 -469.52
+ pat.nurs  1  0.128904 1.6165 -467.93
+ age       1  0.086145 1.6592 -464.98
+ pat.bed   1  0.079078 1.6663 -464.50
<none>                  1.7454 -461.26

Step:  AIC=-469.52
length ~ inf + region + pat

           Df Sum of Sq    RSS     AIC
+ age       1  0.124380 1.4695 -476.70
+ pat.nurs  1  0.079866 1.5140 -473.33
<none>                  1.5939 -469.52
+ pat.bed   1  0.016785 1.5771 -468.71

Step:  AIC=-476.7
length ~ inf + region + pat + age

           Df Sum of Sq    RSS     AIC
+ pat.nurs  1  0.069473 1.4000 -480.17
+ pat.bed   1  0.026608 1.4429 -476.76
<none>                  1.4695 -476.70

Step:  AIC=-480.17
length ~ inf + region + pat + age + pat.nurs

          Df Sum of Sq    RSS     AIC
<none>                 1.4000 -480.17
+ pat.bed  1 0.0097928 1.3902 -478.96
>     summary(fit.F)
Call:
lm(formula = length ~ inf + region + pat + age + pat.nurs, data = senic.03)

Residuals:
     Min       1Q   Median       3Q      Max
-0.21159 -0.07408 -0.01331  0.06479  0.39816

Coefficients:
```

```
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.438983   0.164404   8.753 3.80e-14 ***
inf          0.055360   0.010160   5.449 3.37e-07 ***
regionN     -0.078067   0.030741  -2.540  0.01257 *
regionS     -0.123516   0.030302  -4.076 8.92e-05 ***
regionW     -0.209690   0.038340  -5.469 3.08e-07 ***
pat          0.052614   0.016537   3.182  0.00193 **
age          0.007404   0.002532   2.924  0.00424 **
pat.nurs     0.081985   0.035917   2.283  0.02447 *
---
Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1155 on 105 degrees of freedom
Multiple R-squared:  0.6053,       Adjusted R-squared:  0.579
F-statistic: 23.01 on 7 and 105 DF,  p-value: < 2.2e-16
```

We get the same result as when we performed a backward elimination using AIC and when using p-values (only predictor `pat.bed` has been taken out of the model). Note that this happened in our particular example and is not always the case.

f)
```
>     step(fit.03, direction="both")
```

```
Start:  AIC=-478.96
length ~ age + inf + region + pat + pat.bed + pat.nurs

           Df Sum of Sq    RSS     AIC
- pat.bed   1   0.00979 1.4000 -480.17
<none>                  1.3902 -478.96
- pat.nurs  1   0.05266 1.4429 -476.76
- pat       1   0.09339 1.4836 -473.62
- age       1   0.12006 1.5103 -471.60
- region    3   0.41062 1.8009 -455.72
- inf       1   0.36544 1.7557 -454.59

Step:  AIC=-480.17
length ~ age + inf + region + pat + pat.nurs

           Df Sum of Sq    RSS     AIC
<none>                  1.4000 -480.17
+ pat.bed   1   0.00979 1.3902 -478.96
- pat.nurs  1   0.06947 1.4695 -476.70
- age       1   0.11399 1.5140 -473.33
- pat       1   0.13498 1.5350 -471.77
- inf       1   0.39587 1.7959 -454.03
- region    3   0.46502 1.8651 -453.76

Call:
lm(formula = length ~ age + inf + region + pat + pat.nurs, data = senic.03)

Coefficients:
(Intercept)          age          inf      regionN
   1.438983     0.007404     0.055360    -0.078067
    regionS      regionW          pat     pat.nurs
  -0.123516    -0.209690     0.052614     0.081985
```

Starting with the full model leaves `pat.bed` out the model. Therefore, this method yields to the same result as models `fit.P1`, `fit.B`, and `fit.F`.

```
>     step(fit.empty, scope=scp, direction="both")
```

```
Start:  AIC=-389.11
length ~ 1
```

```
           Df Sum of Sq    RSS     AIC
+ inf       1   1.08286 2.4646 -428.27
+ pat       1   0.94180 2.6057 -421.98
+ region    3   0.98268 2.5648 -419.76
+ pat.bed   1   0.69376 2.8537 -411.70
+ age       1   0.10368 3.4438 -390.46
+ pat.nurs  1   0.07906 3.4684 -389.66
<none>                  3.5475 -389.11

Step:  AIC=-428.27
length ~ inf

           Df Sum of Sq    RSS     AIC
+ region    3   0.71923 1.7454 -461.26
+ pat.bed   1   0.30829 2.1563 -441.37
+ pat       1   0.29591 2.1687 -440.72
+ pat.nurs  1   0.28973 2.1749 -440.40
+ age       1   0.10793 2.3567 -431.33
<none>                  2.4646 -428.27
- inf       1   1.08286 3.5475 -389.11

Step:  AIC=-461.26
length ~ inf + region

           Df Sum of Sq    RSS     AIC
+ pat       1   0.15147 1.5939 -469.52
+ pat.nurs  1   0.12890 1.6165 -467.93
+ age       1   0.08614 1.6592 -464.98
+ pat.bed   1   0.07908 1.6663 -464.50
<none>                  1.7454 -461.26
- region    3   0.71923 2.4646 -428.27
- inf       1   0.81941 2.5648 -419.76

Step:  AIC=-469.52
length ~ inf + region + pat

           Df Sum of Sq    RSS     AIC
+ age       1   0.12438 1.4695 -476.70
+ pat.nurs  1   0.07987 1.5140 -473.33
<none>                  1.5939 -469.52
+ pat.bed   1   0.01678 1.5771 -468.71
- pat       1   0.15147 1.7454 -461.26
- inf       1   0.35905 1.9529 -448.56
- region    3   0.57478 2.1687 -440.72

Step:  AIC=-476.7
length ~ inf + region + pat + age

           Df Sum of Sq    RSS     AIC
+ pat.nurs  1   0.06947 1.4000 -480.17
+ pat.bed   1   0.02661 1.4429 -476.76
<none>                  1.4695 -476.70
- age       1   0.12438 1.5939 -469.52
- pat       1   0.18970 1.6592 -464.98
- inf       1   0.33436 1.8039 -455.53
- region    3   0.53057 2.0001 -447.86

Step:  AIC=-480.17
length ~ inf + region + pat + age + pat.nurs
```

```
           Df Sum of Sq    RSS     AIC
<none>                  1.4000 -480.17
+ pat.bed   1   0.00979 1.3902 -478.96
- pat.nurs  1   0.06947 1.4695 -476.70
- age       1   0.11399 1.5140 -473.33
- pat       1   0.13498 1.5350 -471.77
- inf       1   0.39587 1.7959 -454.03
- region    3   0.46502 1.8651 -453.76

Call:
lm(formula = length ~ inf + region + pat + age + pat.nurs, data = senic.03)

Coefficients:
(Intercept)          inf      regionN      regionS
   1.438983     0.055360    -0.078067    -0.123516
    regionW          pat          age     pat.nurs
  -0.209690     0.052614     0.007404     0.081985
```

Doing stepwise starting with the empty model yields the same result than doing stepwise starting with the full model, backward elimination and forward elimination. Note that this is not always the case: applying these methods with different data could give us different results.

3. a)

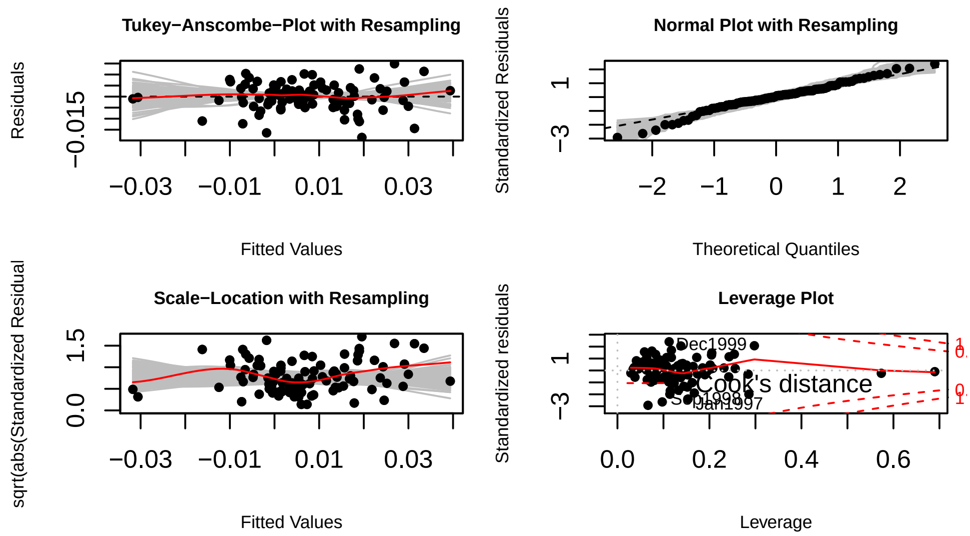
```
> ## load data
> load("FoHF.rda")
> ## fit model with all variables
> fit <- lm(FoHF ~ ., data=FoHF)
> summary(fit)
```

```
Call:
lm(formula = FoHF ~ ., data = FoHF)

Residuals:
       Min         1Q     Median         3Q        Max
-0.0185186 -0.0031189  0.0004069  0.0035469  0.0148925

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.002256   0.001299  -1.736   0.0862 .
RV          -0.388854   0.171151  -2.272   0.0257 *
CA           0.238653   0.104522   2.283   0.0250 *
FIA          0.363010   0.087832   4.133 8.51e-05 ***
EMN          0.184766   0.197475   0.936   0.3522
ED           0.314914   0.215792   1.459   0.1482
DS          -0.007699   0.124324  -0.062   0.9508
MA          -0.028413   0.169406  -0.168   0.8672
LSE          0.153636   0.099548   1.543   0.1266
GM           0.127093   0.086897   1.463   0.1474
EM           0.049183   0.035065   1.403   0.1645
CTA          0.159225   0.037304   4.268 5.20e-05 ***
SS           0.032630   0.023424   1.393   0.1673
---
Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.006563 on 83 degrees of freedom
Multiple R-squared:  0.8076,       Adjusted R-squared:  0.7798
F-statistic: 29.03 on 12 and 83 DF,  p-value: < 2.2e-16
```

Only four variables are significant at the 5% level in the summary output; two variables are associated with a very small p-value. The multiple R-squared is large with a value of more than 80% and also the global F-test is highly significant. This means that the return of the FoHF is explained very well and not all subindices are necessary for this. Therefore, we can assume that the FoHF does not invest in all subindices.

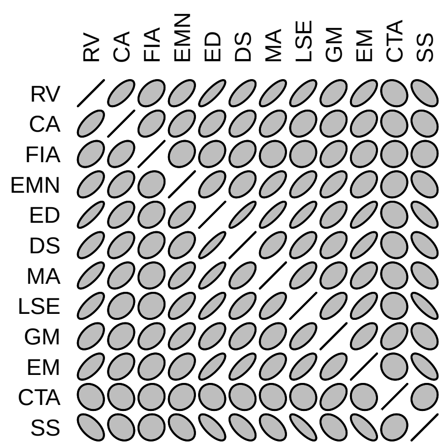
b)

```
> par(mfrow=c(2,2))
> source("../ex1/resplot.R")
> resplot(fit)
```



The residual plots show that there is no systematic error in the model. The normality assumption seems to be satisfied. The smoother in the scale location plot does show some deviations from the horizon. It is generally known that finance data shows volatility, i.e. the (conditional) variance is not necessarily constant over time. However, in this case these deviations are not very strong, so that the constant variance assumption seems to be satisfied and we can proceed with the analysis. Lastly, there are two points with high leverage but since their Cook's distance is small, they can be tolerated.

```
> ## check for large multicollinearity
> library(ellipse)
> par(mfrow=c(1,1))
> plotcorr(cor(FoHF[,-1]), cex.lab = 0.75, mar = c(1,1,1,1))
```



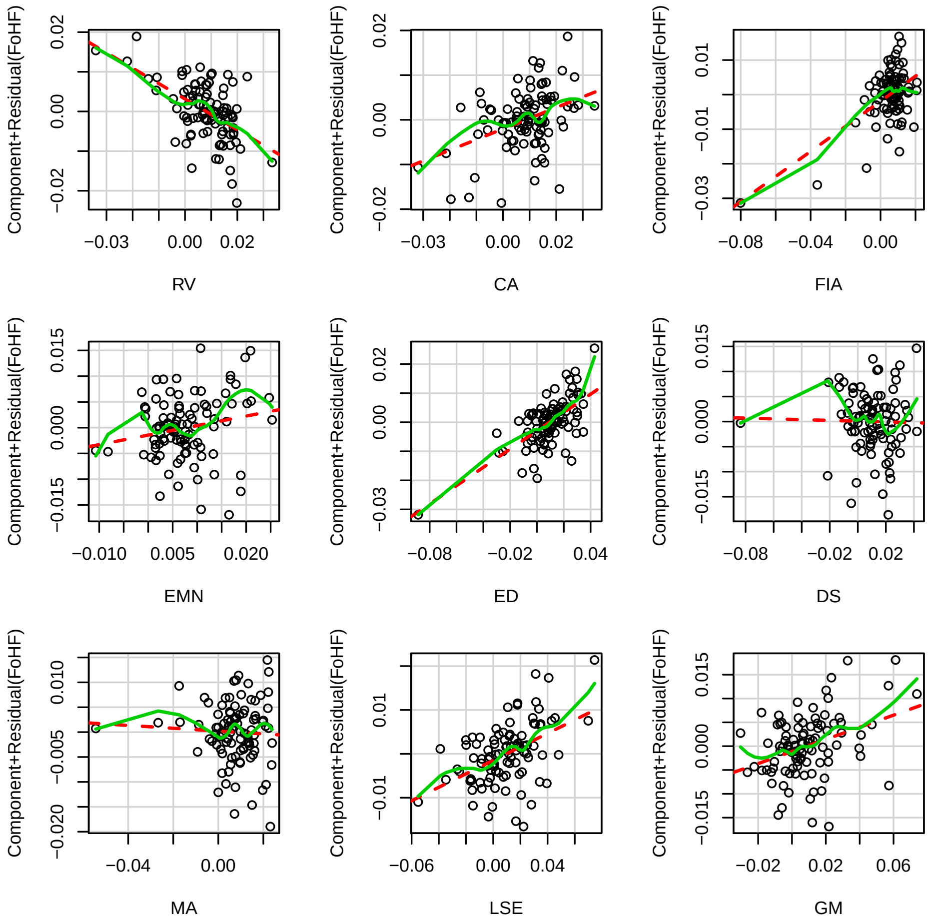
We check for high multicollinearity of the predictors by plotting the pairwise correlations and by computing the VIFs.

```
> library(faraway)
> vif(fit)
       RV        CA       FIA       EMN        ED        DS
 6.387024  2.982646  2.271113  3.672017 29.973694  9.404810
```

```
      MA       LSE        GM        EM       CTA        SS
 8.001994 10.046374  5.699120  4.255477  2.232320  4.972861
```

The VIF of the variables ED and LSE are larger than 10 and therefore larger than the threshold of what can be tolerated.

```
> library(car)
> crPlots(fit, layout = c(3,3))
```



Lastly, we look at the partial residual plots to check whether all predictors have been entered into the model in the correct form. The plots show some deviations but these are neither heavy nor systematic.

c) We saw that there is a multicollinearity problem. The possible remedies are limited in this case: since the FoHF can invest in all of the given subindices, we cannot amputate some of them. Similarly, creating new variables in this context does not make sense and we cannot transform the predictors either – the FoHF invests in the subindices which contribute directly and linearly to the return of the FoHF. The multicollinearity problem is caused by the fact that the performance of certain subindices are indeed highly correlated.

It is possible, however, to perform a variable selection which will hopefully alleviate the multicollinearity problem. From the summary output, it is plausible that the FoHF does not invest in all subindices. In the following subtask, we shall try to achieve a reduction of the model size so that the final model will only contain those subindices the FoHF does in fact invest in.

d) **(i) Stepwise variable selection, starting with the full model.**

```
> ## variable selection with BIC, starting with the full model
> fit.bic.01 <- step(fit, k=log(nrow(FoHF)))
```

```
Start:  AIC=-919.68
FoHF ~ RV + CA + FIA + EMN + ED + DS + MA + LSE + GM + EM + CTA +
    SS

       Df  Sum of Sq       RSS     AIC
- DS    1 0.00000017 0.0035753 -924.24
```

```
- MA    1 0.00000121 0.0035763 -924.21
- EMN   1 0.00003771 0.0036128 -923.24
- SS    1 0.00008358 0.0036587 -922.03
- EM    1 0.00008474 0.0036598 -922.00
- ED    1 0.00009173 0.0036668 -921.81
- GM    1 0.00009214 0.0036672 -921.80
- LSE   1 0.00010260 0.0036777 -921.53
<none>               0.0035751 -919.68
- RV    1 0.00022234 0.0037974 -918.45
- CA    1 0.00022456 0.0037996 -918.40
- FIA   1 0.00073576 0.0043108 -906.28
- CTA   1 0.00078472 0.0043598 -905.20

Step:  AIC=-924.24
FoHF ~ RV + CA + FIA + EMN + ED + MA + LSE + GM + EM + CTA +
    SS

       Df  Sum of Sq       RSS     AIC
- MA    1 0.00000108 0.0035763 -928.78
- EMN   1 0.00003761 0.0036129 -927.80
- SS    1 0.00008344 0.0036587 -926.59
- EM    1 0.00008500 0.0036603 -926.55
- GM    1 0.00009213 0.0036674 -926.36
- LSE   1 0.00010811 0.0036834 -925.95
<none>               0.0035753 -924.24
- RV    1 0.00022710 0.0038024 -922.89
- CA    1 0.00022724 0.0038025 -922.89
- ED    1 0.00023020 0.0038055 -922.82
- FIA   1 0.00073934 0.0043146 -910.76
- CTA   1 0.00079410 0.0043693 -909.55

Step:  AIC=-928.78
FoHF ~ RV + CA + FIA + EMN + ED + LSE + GM + EM + CTA + SS

       Df  Sum of Sq       RSS     AIC
- EMN   1 0.00003909 0.0036154 -932.30
- SS    1 0.00008398 0.0036603 -931.11
- EM    1 0.00008759 0.0036639 -931.02
- GM    1 0.00010058 0.0036769 -930.68
- LSE   1 0.00010832 0.0036846 -930.48
<none>               0.0035763 -928.78
- CA    1 0.00024101 0.0038173 -927.08
- RV    1 0.00026057 0.0038369 -926.59
- ED    1 0.00035144 0.0039278 -924.34
- CTA   1 0.00079349 0.0043698 -914.11
- FIA   1 0.00079685 0.0043732 -914.03

Step:  AIC=-932.3
FoHF ~ RV + CA + FIA + ED + LSE + GM + EM + CTA + SS

       Df  Sum of Sq       RSS     AIC
- EM    1 0.00007430 0.0036897 -934.91
- SS    1 0.00010742 0.0037228 -934.05
- GM    1 0.00012492 0.0037403 -933.60
<none>               0.0036154 -932.30
- LSE   1 0.00018537 0.0038008 -932.06
- RV    1 0.00025580 0.0038712 -930.30
- ED    1 0.00035729 0.0039727 -927.82
- CA    1 0.00040461 0.0040200 -926.68
```

```
- FIA   1 0.00075873 0.0043741 -918.57
- CTA   1 0.00088516 0.0045006 -915.84

Step:  AIC=-934.91
FoHF ~ RV + CA + FIA + ED + LSE + GM + CTA + SS

       Df  Sum of Sq       RSS     AIC
- SS    1 0.00005369 0.0037434 -938.09
- LSE   1 0.00014269 0.0038324 -935.83
<none>               0.0036897 -934.91
- GM    1 0.00024280 0.0039325 -933.36
- RV    1 0.00026091 0.0039506 -932.92
- CA    1 0.00037752 0.0040672 -930.12
- ED    1 0.00058092 0.0042706 -925.44
- CTA   1 0.00081755 0.0045073 -920.26
- FIA   1 0.00083132 0.0045210 -919.97

Step:  AIC=-938.09
FoHF ~ RV + CA + FIA + ED + LSE + GM + CTA

       Df  Sum of Sq       RSS     AIC
- LSE   1 0.00009111 0.0038345 -940.34
<none>               0.0037434 -938.09
- RV    1 0.00024437 0.0039878 -936.58
- GM    1 0.00027617 0.0040196 -935.82
- CA    1 0.00038539 0.0041288 -933.24
- ED    1 0.00052919 0.0042726 -929.96
- CTA   1 0.00083822 0.0045816 -923.25
- FIA   1 0.00092714 0.0046706 -921.41

Step:  AIC=-940.34
FoHF ~ RV + CA + FIA + ED + GM + CTA

       Df  Sum of Sq       RSS     AIC
- RV    1 0.00016854 0.0040031 -940.78
<none>               0.0038345 -940.34
- CA    1 0.00031176 0.0041463 -937.40
- GM    1 0.00072123 0.0045558 -928.36
- CTA   1 0.00074886 0.0045834 -927.78
- ED    1 0.00083966 0.0046742 -925.90
- FIA   1 0.00085130 0.0046858 -925.66

Step:  AIC=-940.78
FoHF ~ CA + FIA + ED + GM + CTA

       Df  Sum of Sq       RSS     AIC
<none>               0.0040031 -940.78
- CA    1 0.00019591 0.0041990 -940.76
- GM    1 0.00066084 0.0046639 -930.67
- ED    1 0.00071917 0.0047222 -929.48
- FIA   1 0.00073423 0.0047373 -929.18
- CTA   1 0.00089226 0.0048953 -926.03
> summary(fit.bic.01)
Call:
lm(formula = FoHF ~ CA + FIA + ED + GM + CTA, data = FoHF)

Residuals:
      Min        1Q    Median        3Q       Max
-0.017656 -0.003736  0.000617  0.003476  0.016531
```

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.0018567  0.0009089  -2.043 0.043984 *
CA           0.1756651  0.0837020   2.099 0.038645 *
FIA          0.2984918  0.0734666   4.063 0.000103 ***
ED           0.2654424  0.0660132   4.021 0.000120 ***
GM           0.2469422  0.0640654   3.855 0.000217 ***
CTA          0.1535033  0.0342726   4.479  2.2e-05 ***
---
Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.006669 on 90 degrees of freedom
Multiple R-squared:  0.7846,       Adjusted R-squared:  0.7726
F-statistic: 65.55 on 5 and 90 DF,  p-value: < 2.2e-16
```

(ii) Stepwise variable selection, starting with the empty model.

```
> ## variable selection with BIC, starting with the empty model
> fit.null   <- lm(FoHF ~ 1, data=FoHF)
> scopi      <- list(lower=formula(fit.null), upper=formula(fit))
> fit.bic.02 <- step(fit.null, scope=scopi, k=log(nrow(FoHF)))
```

```
Start:  AIC=-816.23
FoHF ~ 1

       Df Sum of Sq       RSS     AIC
+ GM    1 0.0116463 0.0069343 -906.29
+ ED    1 0.0072492 0.0113314 -859.15
+ EMN   1 0.0071500 0.0114306 -858.31
+ DS    1 0.0066117 0.0119689 -853.89
+ EM    1 0.0065705 0.0120101 -853.56
+ FIA   1 0.0059473 0.0126333 -848.70
+ LSE   1 0.0058787 0.0127020 -848.18
+ RV    1 0.0055859 0.0129947 -846.00
+ CA    1 0.0047059 0.0138748 -839.71
+ MA    1 0.0043282 0.0142525 -837.13
+ CTA   1 0.0027357 0.0158449 -826.96
+ SS    1 0.0020455 0.0165351 -822.87
<none>              0.0185806 -816.23

Step:  AIC=-906.29
FoHF ~ GM

       Df Sum of Sq       RSS     AIC
+ CA    1 0.0013440 0.0055904 -922.41
+ FIA   1 0.0012867 0.0056477 -921.43
+ DS    1 0.0008103 0.0061241 -913.66
+ ED    1 0.0007262 0.0062081 -912.35
+ RV    1 0.0004910 0.0064434 -908.78
+ MA    1 0.0004643 0.0064700 -908.38
+ EMN   1 0.0004195 0.0065148 -907.72
<none>              0.0069343 -906.29
+ EM    1 0.0002708 0.0066636 -905.55
+ CTA   1 0.0000318 0.0069025 -902.17
+ LSE   1 0.0000280 0.0069064 -902.11
+ SS    1 0.0000009 0.0069334 -901.74
- GM    1 0.0116463 0.0185806 -816.23

Step:  AIC=-922.41
```

```
FoHF ~ GM + CA

       Df Sum of Sq       RSS     AIC
+ FIA   1 0.0004944 0.0050959 -926.73
+ CTA   1 0.0003730 0.0052174 -924.47
<none>              0.0055904 -922.41
+ DS    1 0.0001376 0.0054527 -920.24
+ ED    1 0.0001021 0.0054882 -919.61
+ EM    1 0.0000447 0.0055457 -918.61
+ MA    1 0.0000393 0.0055511 -918.52
+ SS    1 0.0000330 0.0055574 -918.41
+ EMN   1 0.0000191 0.0055712 -918.17
+ RV    1 0.0000025 0.0055878 -917.89
+ LSE   1 0.0000024 0.0055880 -917.88
- CA    1 0.0013440 0.0069343 -906.29
- GM    1 0.0082844 0.0138748 -839.71

Step:  AIC=-926.73
FoHF ~ GM + CA + FIA

       Df Sum of Sq       RSS     AIC
+ CTA   1 0.0003737 0.0047222 -929.48
<none>              0.0050959 -926.73
+ ED    1 0.0002006 0.0048953 -926.03
+ MA    1 0.0001505 0.0049454 -925.05
+ DS    1 0.0001452 0.0049507 -924.95
+ EMN   1 0.0001411 0.0049548 -924.87
+ EM    1 0.0000704 0.0050255 -923.51
+ LSE   1 0.0000383 0.0050577 -922.89
- FIA   1 0.0004944 0.0055904 -922.41
+ SS    1 0.0000056 0.0050904 -922.27
+ RV    1 0.0000055 0.0050904 -922.27
- CA    1 0.0005517 0.0056477 -921.43
- GM    1 0.0063437 0.0114396 -853.67

Step:  AIC=-929.48
FoHF ~ GM + CA + FIA + CTA

       Df Sum of Sq       RSS     AIC
+ ED    1 0.0007192 0.0040031 -940.78
+ DS    1 0.0004934 0.0042289 -935.51
+ LSE   1 0.0003982 0.0043240 -933.37
+ EM    1 0.0003709 0.0043513 -932.77
+ MA    1 0.0003524 0.0043698 -932.36
<none>              0.0047222 -929.48
+ SS    1 0.0001590 0.0045633 -928.20
+ EMN   1 0.0001447 0.0045775 -927.91
- CTA   1 0.0003737 0.0050959 -926.73
+ RV    1 0.0000481 0.0046742 -925.90
- FIA   1 0.0004951 0.0052174 -924.47
- CA    1 0.0008029 0.0055252 -918.97
- GM    1 0.0035251 0.0082473 -880.52

Step:  AIC=-940.78
FoHF ~ GM + CA + FIA + CTA + ED

       Df  Sum of Sq       RSS     AIC
<none>               0.0040031 -940.78
- CA    1 0.00019591 0.0041990 -940.76
```

```
+ RV    1 0.00016854 0.0038345 -940.34
+ EMN   1 0.00004912 0.0039539 -937.40
+ EM    1 0.00002789 0.0039752 -936.88
+ LSE   1 0.00001528 0.0039878 -936.58
+ SS    1 0.00001019 0.0039929 -936.46
+ DS    1 0.00000805 0.0039950 -936.41
+ MA    1 0.00000154 0.0040015 -936.25
- GM    1 0.00066084 0.0046639 -930.67
- ED    1 0.00071917 0.0047222 -929.48
- FIA   1 0.00073423 0.0047373 -929.18
- CTA   1 0.00089226 0.0048953 -926.03
```

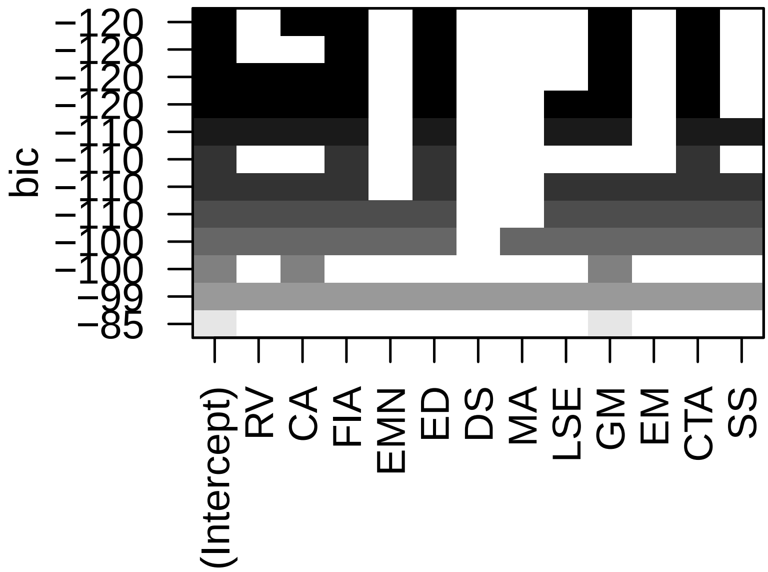
```
> summary(fit.bic.02)
```

```
Call:
lm(formula = FoHF ~ GM + CA + FIA + CTA + ED, data = FoHF)

Residuals:
      Min        1Q    Median        3Q       Max
-0.017656 -0.003736  0.000617  0.003476  0.016531

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.0018567  0.0009089  -2.043 0.043984 *
GM           0.2469422  0.0640654   3.855 0.000217 ***
CA           0.1756651  0.0837020   2.099 0.038645 *
FIA          0.2984918  0.0734666   4.063 0.000103 ***
CTA          0.1535033  0.0342726   4.479  2.2e-05 ***
ED           0.2654424  0.0660132   4.021 0.000120 ***
---
Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.006669 on 90 degrees of freedom
Multiple R-squared:  0.7846,       Adjusted R-squared:  0.7726
F-statistic: 65.55 on 5 and 90 DF,  p-value: < 2.2e-16
```

(iii) All Subsets variable selection.

```
> ## All Subsets Search
> library(leaps)
> out <- regsubsets(FoHF~., nvmax=12, data=FoHF)
> plot(out)
> coef(out,5)
```

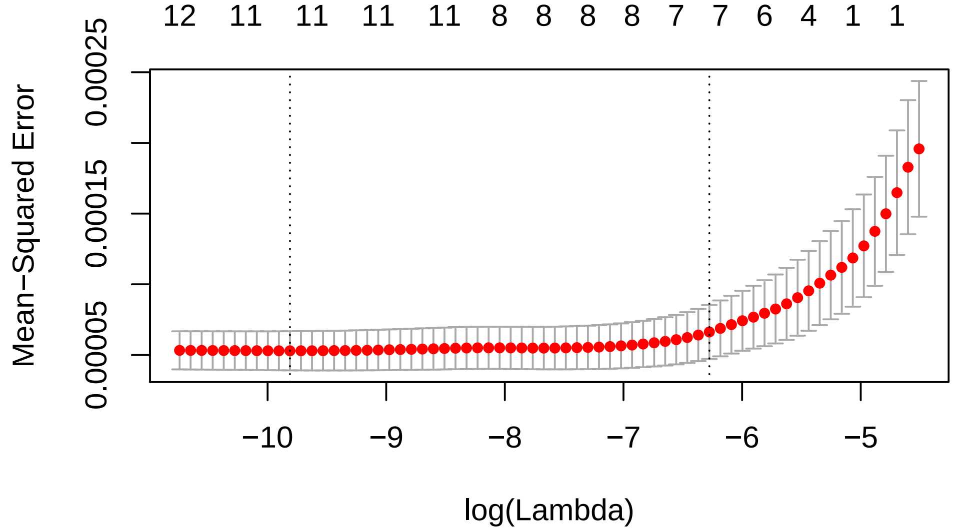
```
 (Intercept)           CA          FIA           ED
-0.001856729  0.175665074  0.298491812  0.265442447
          GM          CTA
 0.246942244  0.153503336
```



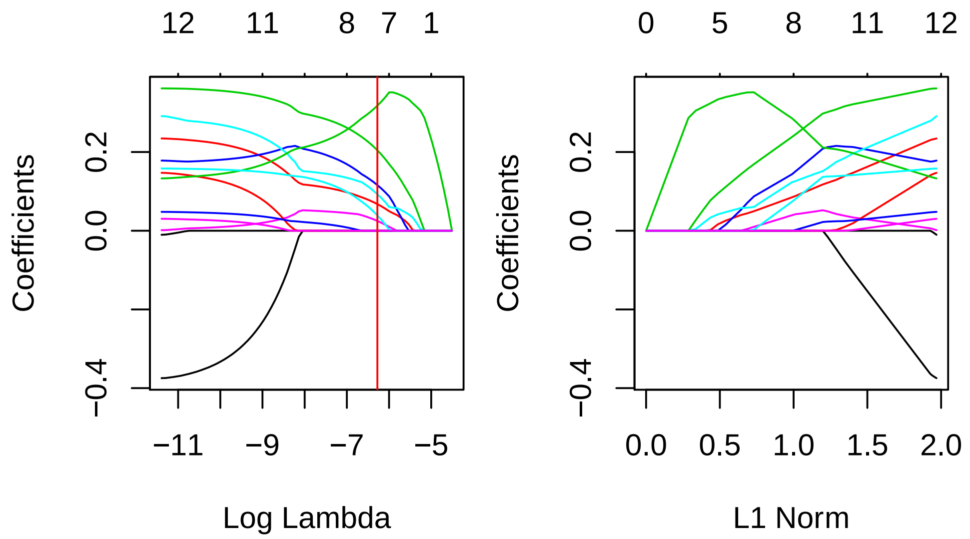
All three variable selection methods yield the same model. It contains the subindices `CA`, `FIA`, `ED`, `GM` and `CTA`. All Subsets search shows that there are three alternative models with almost identical BIC values – they contain 4, 6 and 7 predictors respectively.

e)
```
> ## Lasso
> library(glmnet)
> xx        <- model.matrix(FoHF~., data=FoHF)
> yy        <- FoHF$FoHF
> cvfit     <- cv.glmnet(xx,yy)
> plot(cvfit)
> coef(cvfit, s = "lambda.1se")
```

```
14 x 1 sparse Matrix of class "dgCMatrix"
                       1
(Intercept) 0.0002196824
(Intercept) .
RV          .
CA          0.0676192479
FIA         0.2044571894
EMN         0.1160512738
ED          0.0927171171
DS          0.0249532901
MA          .
LSE         .
GM          0.3178869629
EM          .
CTA         0.0390495344
SS          .
```




```
> fit.lasso <- glmnet(xx,yy)
> par(mfrow = c(1,2))
> plot(fit.lasso, label = TRUE, xvar = "lambda")
> abline(v = log(cvfit$lambda.1se), col = 2)
> plot(fit.lasso, label = TRUE)
```



Cross validation yields a model with 7 predictors. However, these are not identical to the ones chosen by the best BIC fit with 7 predictors. In general, the Lasso is a suitable tool as it can handle the multicollinearity of the predictors and perform variable selection. Both of these aspects are necessary here since the subindices are collinear and we know that the FoHF is not invested in all possible subindices. Therefore, the Lasso solution should also be considered next to the one from the variable selection with BIC.