```
#########################################################
##                                                     ##
##              Appendix 2 - R code                    ##
##                                                     ##
#########################################################

## Data import from SPSS   ##

> library(rms)
> library(Hmisc)
> data <- spss.get() #file path in brackets

## Basic properties, table 1 (article) ##

> names(data)

"Nr"          "age"         "sex"         "status"
"time"        "cci"         "dressing"    "eating"
"physical"    "spontaneous" "hygiene"     "toilet"
"adl"         "hb"          "gfr"         "albumin"
"bnp"         "bmi"         "control"

> describe(data)

#########################################################
## 1. outcome                                          ##
##                                                     ##
#########################################################

## Determining follow-up and censoring ##

> library (survival)
> S <- Surv(time, status)
> cens.time <- ifelse(status == ”alive”, time, NA)
summary(cens.time)

## Baseline survival plot, figure 1 (appendix 1) ##

> S.years <- Surv(time/365.25, status)

> survplot(npsurv(S.years~1), xlab = "years", xlim =
c(0,4), time.inc = 1, lwd = 1.5, n.risk = T, y.n.risk =
0.05, cex.n.risk = 1, adj.n.risk = 0.5)

## discarding cases with missing outcome ##

> missing.outcome <- is.na(data$status)
> data <- data[missing.outcome == FALSE,]
> attach(data)

#########################################################
## 2. Crude analysis                                   ##
##                                                     ##
#########################################################
```

```
## Crude analysis for all predictors, table 2 (article)##

> summary(coxph(S~age))
> summary(coxph(S~sex))
> summary(coxph(S~cci))
> summary(coxph(S~dressing))
> summary(coxph(S~eating))
> summary(coxph(S~physical))
> summary(coxph(S~spontaneous))
> summary(coxph(S~hygiene))
> summary(coxph(S~toilet))
> summary(coxph(S~adl))
> summary(coxph(S~hb))
> summary(coxph(S~gfr))
> summary(coxph(S~albumin))
> summary(coxph(S~bnp))
> summary(coxph(S~bmi))
> summary(coxph(S~control))

## Discarding the separate ADL items ##

> data <- data[,-c(7:12)]

##########################################################
## 3. Missing data                                      ##
##                                                      ##
##########################################################

## Defining covariates ##

> covs <- data[, c("age", "sex", "cci", "adl", "hb",
"clearance", "albumin", "bnp", "bmi", ”control”)]

## Plotting missing, figure 2 (appendix 1) ##

> missing <- naclus(covs)
> naplot(missing, which = ”na per var”)

## Discarding the BNP variable ##

> data <- data[,-8]

## Determining associations with missing albumin ##

> missing.albumin <- ifelse(is.na(albumin), 1, 0)
> lrm(missing.albumin~age+sex+cci+clearance+bmi+adl+hb)
> chisq.test(missing.albumin, sex)
> oneway.test(cci~missing.albumin)
> tapply (cci, missing.albumin, mean)

## Creating a transcan object for imputation ##

> trans <-transcan(~age + sex + cci + adl + hb +
clearance + albumin + bmi + control, imputed = T, data =
data)
```

```
## Imputing albumin, gfr and bmi ##

> albumin.imputed <- as.integer(impute(trans, albumin,
data = data))
> gfr.imputed <- as.integer(impute(trans, gfr, data =
data))
> bmi.imputed <- as.integer(impute(trans, bmi, data =
data))

## Testing imputed variables, table 1 (appendix 1) ##

> summary(coxph(S~albumin.imputed)
> summary(coxph(S~gfr.imputed)
> summary(coxph(S~bmi.imputed)

## Imputed variables put into original dataset ##

> data$gfr <- gfr.imputed
> data$albumin <- albumin.imputed
> data$bmi <- bmi.1
> attach(data)

#########################################################
## 4. Variable considerations                          ##
##                                                     ##
#########################################################

## Screening for outliers, figure 3 (appendix 1) ##

> par(mfrow = c(3,3))
> boxplot(age)
> boxplot(cci)
> boxplot(hb)
> boxplot(gfr)
> boxplot(albumin)
> boxplot(bmi)
> boxplot(adl)

## Winsorising at 99th percentile, table 2(appendix 1) ##

> describe(gfr)
> gfr.winsorised <- ifelse(gfr > 118, 118, gfr)
> summary(coxph(S~gfr.winsorised))
> data$gfr <- gfr.winsorised
> attach(data)

#########################################################
## Transformations for haemoglobin figure 4, appendix 1##
#########################################################

## Range is obtained ##

> describe(hb)

## Linear model fitted and plotted ##
```

```
> hbfit.linear <- cph(S~hb, data = data)
> plot(Predict(hbfit.linear, hb = seq(53, 179, by = 1)),
xlab = "Haemoglobin g/L" , anova = anova(hbfit.linear),
pval = T, data = llist(hb))

## Dichotomous model##

> data$anemia <- ifelse(sex == ”male”, ifelse(hb < 130,
”yes”, ”no”), ifelse (hb < 120, ”yes”, ”no”))
> dd <- datadist (data)
> options(datadist = ”dd”)
> hbfit.dichotomous <- cph(S~anemia, data = data)
> plot(Predict(hbfit.dichotomous), xlab = "Anemia", anova
= anova(hbfit.dichotomous), pval = T)

## Categorical model ##

> data$hbcat <- ifelse(hb < 92.25, 1, ifelse(hb < 118, 2,
ifelse(hb < 130, 3, ifelse (hb < 148, 4, 5))))
> dd <- datadist (data)
> options(datadist = ”dd”)
> hbfit.categorical <- cph(S~as.factor(hbcat), data =
data)
> plot(Predict(hbfit.categorical), anova =
anova(hbfit.categorical), xlab = ”Haemoglobin g/L”, pval
= T)

## Restricted cubic spline ##

> hbfit.spline <- cph(S~rcs(hb,4), data = data)
> plot(Predict(hbfit.spline, hb = seq(53,179, by = 1)),
anova = anova(hbfit.spline), pval = T, xlab =
"Haemoglobin g/L", data = llist(hb))

## Testing other continuous variables ##

> agefit <- cph(S~rcs(age,4), data = data)
> anova(agefit)

> ccifit <- cph(S~rcs(cci,4), data = data)
> anova(ccifit)

> albuminfit <- cph(S~rcs(albumin,4), data = data)
> anova(albuminfit)

> bmifit <- cph(S~rcs(bmi,4), data = data)
> anova(bmifit)

> gfrfit <- cph(S~rcs(gfr,4), data = data)
> anova(gfrfit)

> adlfit <- cph(S~rcs(adl,4), data = data)
> anova(adlfit)
```

```
#########################################################
## Transformations for ADL figure 5, appendix 1        ##
#########################################################

> describe(adl)

## Linear fit ##

> adlfit.linear <- cph(S~adl, data = data)
> plot(Predict(adlfit.linear, adl = seq(0,25, by = 1)),
anova = anova(adlfit.linear), pval = T, data =
llist(adl), xlab = "GBS-ADL")

## Dichotomised at median ##

> data$adl.dichotomised <- ifelse(adl<5,0,1)
> dd <- datadist(data)
> options(datadist = "dd")
> adlfit.dichotomous <- cph(S~adl.dichotomised, data =
data)
> plot(Predict(adlfit.dichotomous), anova =
anova(adlfit.dichotomous), pval = T)

## Categorised at quartiles ##

> data$adl.quartiles <- ifelse(adl<3, 1, ifelse(adl<6,2,
ifelse(adl<10, 3,4)))
> dd <- datadist(data)
> options(datadist = "dd")
> adlfit.quartiles <- cph(S~as.factor(adl.quartiles),
data = data)
> plot(Predict(adlfit.quartiles), anova =
anova(adlfit.quartiles), pval = T)

## Two-degree polynomial ##

> adlfit.poly <- cph(S~pol(adl,2), data = data)
> plot(Predict(adlfit.poly, adl = seq(0,25,by=1)), anova
= anova(adlfit.poly), pval = T, xlab = "GBS-ADL", data =
llist(adl))

## four-knot spline ##

> adlfit.spline <- cph(S~rcs(adl,4), data = data)
> plot(Predict(adlfit.spline, adl = seq(0,25,by=1)),
anova = anova(adlfit.spline), pval = T, xlab = "GBS-ADL",
data = llist(adl))

## Log fit ##

> adlfit.log <- cph(S~log(adl+1), data = data)
> plot(Predict(adlfit.log, adl = seq(0,25,by=1)), anova =
anova(adlfit.log), pval = T, xlab = "GBS-ADL", data =
llist(adl))
```

```
##########################################################
## 5. Fitting the multivariate models                  ##
##                                                     ##
##########################################################

## The model without ADL ##

> model1 <- cph(S~age + sex + cci + rcs (hb, 4) + albumin
+ bmi + control + gfr, x = T, y = T, surv = T, data =
data)

## The full model ##

> model2 <- cph(S~age + sex + cci + rcs(hb, 4) + albumin
+ bmi + control + gfr + log (adl + 1), x = T, y = T, surv
= T, data = data)

##########################################################
## 6. Multicolinearity                                 ##
##                                                     ##
##########################################################

> cor(data)
> vif(model1)
> vif(model2)

##########################################################
## 7. Interactions                                     ##
##                                                     ##
##########################################################

## Global tests for the model without ADL ##

> z1 <- predict(model1, type = ”terms”)

> age.ia <- z1[,”age”]
> all.others <- z1[,-1]
> anova(cph(S~age.ia*all.others))

> sex.ia <- z1[,”sex”]
> all.others <- z1[,-2]
> anova(cph(S~sex.ia*all.others))
> cph(S~sex.ia*all.others)

> cci.ia <- z1[,"cci"]
> all.others <- z1[,-3]
> anova(cph(S~cci.ia*all.others))

> hb.ia <- z1[,"hb"]
> all.others <- z1[,-4]
> anova(cph(S~hb.ia*all.others))

> albumin.ia <- z1[,"albumin"]
> all.others <- z1[,-5]
> anova(cph(S~albumin.ia*all.others))
> bmi.ia <- z1[,"bmi"]
> all.others <- z1[,-6]
> anova(cph(S~bmi.ia*all.others))
> cph(S~bmi.ia*all.others)

> control.ia <- z1[,"control"]
> all.others <- z1[,-7]
> anova(cph(S~control.ia*all.others))

> gfr.ia <- z1[,"gfr"]
> all.others <- z1[,-8]
> anova(cph(S~gfr.ia*all.others))

## The full model ##

> z2 <- predict(model2, type = ”terms”)

> age.ia2 <- z2[,"age"]
> all.others.2 <- z2[,-1]
> anova(cph(S~age.ia2*all.others.2))

> sex.ia2 <- z2[,"sex"]
> all.others.2 <- z2[,-2]
> anova(cph(S~sex.ia2*all.others.2))

> cci.ia2 <- z2[,"cci"]
> all.others.2 <- z2[,-3]
> anova(cph(S~cci.ia2*all.others.2))

> hb.ia2 <- z2[,"hb"]
> all.others.2 <- z2[,-4]
> anova(cph(S~hb.ia2*all.others.2))

> albumin.ia2 <- z2[,"albumin"]
> all.others.2 <- z2[,-5]
> anova(cph(S~albumin.ia2*all.others.2))

> bmi.ia2 <- z2[,"bmi"]
> all.others.2 <- z2[,-6]
> anova(cph(S~bmi.ia2*all.others.2))

> control.ia2 <- z2[,"control"]
> all.others.2 <- z2[,-7]
> anova(cph(S~control.ia2*all.others.2))

> gfr.ia2 <- z2[,"gfr"]
> all.others.2 <- z2[,-8]
> anova(cph(S~gfr.ia2*all.others.2))
> cph(S~gfr.ia2*all.others.2)

> adl.ia <- z2[,"adl"]
> all.others <- z2[,-9]
> anova(cph(S~adl.ia*all.others))
> cph(S~adl.ia*all.others)
```

```
## Updating the models with the interactions ##

> model1 <- cph(S~age + sex * bmi + cci + rcs (hb, 4) +
albumin + control + gfr, data = data, x = T, y = T, surv
= T)

> model2 <- cph(S~age + sex * bmi + cci + rcs (hb, 4) +
albumin + control + gfr * log (adl + 1), data = data, x =
T, y = T, surv = T)

#########################################################
## 8. Proportional hazards assumption                  ##
##                                                     ##
#########################################################

> z3 <- predict(model1, type = "terms")
> model1.short <- cph(S~z3, x = T, y = T)
> ph1 <- cox.zph(model1.short, transform = "identity")
> ph1

> z4 <- predict(model2, type = ”terms”)
> model2.short <- cph(S~z4, x = T, y = T)
> ph2 <- cox.zph(model2.short, transform = ”identity”)
> ph2
> plot(ph2, var = ”gfr”) ##figure 6 (appendix 1) ##

#########################################################
## 9. Influential observations                         ##
##                                                     ##
#########################################################

> inf1 <- which.influence(model1)
> show.influence(inf1, dframe = data)

> inf2 <- which.influence(model2)
> show.influence(inf2, dframe = data)
> inf2

## Sensitivity analysis without influential for ADL ##

> subset <- data[-c(3,25,38,56,67,69,95,108,161),]
> attach(subset)
> S.sens <- Surv (time, status)
> sensitivity.model <- cph(S.sens~age + sex * bmi + cci+
rcs (hb, 4) + albumin + control + gfr * log(adl + 1), x=
T, y = T, surv = T, data = subset)
> sensitivity.model
> anova(sensitivity.model)
> detach(subset)
```

```
##########################################################
## 10. Relative contribution of ADL, figure 1(article) ##
##     figure 7(appendix)                               ##
##                                                      ##
##########################################################

> plot(anova(model1), margin = "P", rm.ia = TRUE)
> plot(Predict(model1), anova = anova(model1), pval = T)

> plot(anova(model2), margin = "P", rm.ia = TRUE)
> plot(Predict(model2), anova = anova(model1), pval = T)

##########################################################
## 11. Added value of ADL, table 3 (article)            ##
##                                                      ##
##########################################################

## Likelihood ratio $\chi^2$ test ##

> lrtest(model1, model2)

## Discrimination ##

> library (survC1)
> mydata <- as.matrix(data[,c(”time”, ”status”)])
> Inf.Cval.Delta(mydata, model1$x, model2$x, tau = 1428)

## NRI>0 and IDI ##

> library(survIDINRI)
> i <- IDI.INF(mydata, model1$x, model2$x, t0 = 1428)
> IDI.INF.OUT(i)

##########################################################
## 12. Internal validation                              ##
##                                                      ##
##########################################################

> validate(model1, B = 1000)
> validate(model2, B = 1000)

##########################################################
##                                                      ##
## 13. Updating the model                               ##
##                                                      ##
##########################################################

>library(glmpath)
> mydata <- list(x = predict(model2, type = "ccterms"),
time = data$time, status = data$status)
> path <- coxpath(data = mydata)

##creating figure 8 appendix ##
```

```
> plot(path)
> plot(path, type = ”aic”)

## Determining the shrinkage factors ##

> lasso.factors <- path$b.predictor[path$aic ==
min(path$aic),]

## Shrinking the lasso.coefs ##

> lasso.coefs <- model2$coef

> lasso.coefs[”age”] <- lasso.coefs[”age”] *
lasso.factors[1]
> lasso.coefs[”sex”] <- lasso.coefs[”sex”] *
lasso.factors[2]
> lasso.coefs[”bmi”] <- lasso.coefs[”bmi”] *
lasso.factors[2]
> lasso.coefs[”cci”] <- lasso.coefs[”cci”] *
lasso.factors[3]
> lasso.coefs[”hb”] <- lasso.coefs[”hb”] *
lasso.factors[4]
> lasso.coefs[”hb’”] <- lasso.coefs[”hb’”] *
lasso.factors[4]
> lasso.coefs[”hb’’”] <- lasso.coefs[”hb’’”] *
lasso.factors[4]
> lasso.coefs[”albumin”] <- lasso.coefs[”albumin”] *
lasso.factors[5]
> lasso.coefs[”control”] <- lasso.coefs[”control”] * 0
> lasso.coefs[”gfr”] <- lasso.coefs[”gfr”] *
lasso.factors[7]
> lasso.coefs[”adl”] <- lasso.coefs[”adl”] *
lasso.factors[7]
> lasso.coefs[”sex * bmi”] <- lasso.coefs[”sex * bmi”] *
lasso.factors[2] * lasso.factors[2]
> lasso.coefs[”gfr * adl”] <- lasso.coefs[”gfr * adl”] *
lasso.factors[7] * lasso.factors[7]

## Updating the model ””

> lassomodel <- model2
> lassomodel$coefficients <- lasso.coefs

## Plotting nomogram, figure 2 (article) ##

> plot(nomogram(lassomodel, age = c(60,80,100), albumin =
c(15,20,30,40,45), bmi = c(15,20,25,30,35), hb =
c(50,70,90,110,150,175), interact = list(gfr =
c(27,36,51), adl, bmi, sex), lp = T, lp.at = c(-4,-
2,0,2), nint = 5, maxscale = 50))

## creating four risk groups ##
```

```
> risk.group <- cut2(as.numeric
(lassomodel$linear.predictor), g = 4)
> levels(risk.group) <- as.character(1:4)

## Kaplan-Meier plot, figure 3 (article)

> survplot(npsurv(S~risk.group, data = data), xlim =
c(0,1318), label.curves = FALSE, conf = "none", n.risk =
T, xlab = "follow-up (days)", cex.nrisk = 0.8, ylab =
"Fraction survivors", time.inc = 364, sep.n.risk = 0.03,
y.n.risk = 0, col = c(1,2,3,4), lty = 1)


### END ###
```