

# Heart Valve project

Group F

2024-05-11

## 1. descriptive analyses of all variables in the dataset.

```
hv <- read.table("heart.valve.txt", na.strings=".",header=T,row.names=NULL)
hv$age <- floor(hv$age)
skim_without_charts(hv[-1])
```

Table 1: Data summary

Name	hv[-1]
Number of rows	255
Number of columns	9
Column type frequency:	
numeric	9
Group variables	None

### Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100
log.lvmi	0	1	5.06	0.39	3.56	4.76	5.04	5.29	6.35
fuyrs	0	1	5.32	2.52	0.02	3.39	5.03	7.38	10.74
status	0	1	0.21	0.41	0.00	0.00	0.00	0.00	1.00
sex	0	1	0.29	0.45	0.00	0.00	0.00	1.00	1.00
age	0	1	65.45	12.38	23.00	58.00	67.00	75.00	89.00
con.cabg	0	1	0.31	0.46	0.00	0.00	0.00	1.00	1.00
creat	0	1	102.13	31.18	50.00	81.00	96.00	116.50	264.00
lv	0	1	1.51	0.65	1.00	1.00	1.00	2.00	3.00
sten.reg.mix	0	1	1.56	0.79	1.00	1.00	1.00	2.00	3.00

The variable log.lvmi exhibits a mean of 5.06 and a median of 5.04, indicating a relatively symmetric distribution. The standard deviation is 0.39, suggesting low variability, with values ranging from 3.56 to 6.35.

The fuyrs variable has a mean of 5.32 and a median of 5.03, again indicating symmetry in the distribution. However, its standard deviation is 2.52, and the range extends from 0.02 to 10.74, highlighting a higher degree of variability compared to log.lvmi.

In contrast, the status variable, with a mean of 0.21 and a median of 0.00, is highly skewed towards zero. The standard deviation is 0.41, and the values span from 0.00 to 1.00. This indicates that the majority of the data points are clustered at the lower end.

Similarly, the sex variable shows a mean of 0.29 and a median of 0.00, with a standard deviation of 0.45. The distribution is skewed towards zero, suggesting that the majority of the sample is concentrated at this value.

The age variable provides a different perspective with a mean of 65.45 and a median of 67.00. The close proximity of these measures suggests a symmetric distribution. However, the standard deviation of 12.38 and a range from 23.00 to 89.00 reflect moderate variability within this variable.

The con.cabg variable, like status and sex, is skewed towards zero, with a mean of 0.31 and a median of 0.00. Its standard deviation is 0.46, and it spans the same range of 0.00 to 1.00.

The creat variable, with a mean of 102.13 and a median of 96.00, indicates a possible right skew. The standard deviation of 31.18 and the range from 50.00 to 264.00 suggest high variability, indicating significant differences in creatinine levels among the subjects.

The lv variable shows a mean of 1.51 and a median of 1.00, indicating a distribution clustered around the lower end. The standard deviation is 0.65, and the values range from 1.00 to 3.00, pointing to less variability.

Lastly, the sten.reg.mix variable has a mean of 1.56 and a median of 1.00, similar to lv. The standard deviation is 0.79, and the range is from 1.00 to 3.00, suggesting moderate variability and a distribution skewed towards the lower values.

## 2. univariate analysis (Cox model) of the association of each independent variable with the outcome under study.

```
model1 <- coxph(Surv(fuyrs,status) ~ factor(sex),data=hv)
summary(model1)
```

```
## Call:
## coxph(formula = Surv(fuyrs, status) ~ factor(sex), data = hv)
##
##    n= 255, number of events= 54
##
##              coef exp(coef) se(coef)    z Pr(>|z|)
## factor(sex)1 0.2055    1.2281  0.2936 0.7    0.484
##
##              exp(coef) exp(-coef) lower .95 upper .95
## factor(sex)1    1.228    0.8143    0.6907    2.184
##
## Concordance= 0.526 (se = 0.035 )
## Likelihood ratio test= 0.48 on 1 df,  p=0.5
## Wald test               = 0.49 on 1 df,  p=0.5
## Score (logrank) test = 0.49 on 1 df,  p=0.5
```

The hazard ratio for sex is 1.228, with a confidence interval of (0.6907, 2.184), suggesting no significant difference in hazards between sexes. The p-value for sex is 0.484, indicating the effect is not statistically significant. Model fit statistics, including the likelihood ratio test, Wald test, and score test, all have p-values around 0.5, further supporting the lack of significance.

```
model2 <- coxph(Surv(fuyrs,status) ~ factor(con.cabg),data=hv)
summary(model2)
```

```
## Call:
## coxph(formula = Surv(fuyrs, status) ~ factor(con.cabg), data = hv)
##
##      n= 255, number of events= 54
##
##              coef exp(coef) se(coef)      z Pr(>|z|)
## factor(con.cabg)1 0.9646      2.6236   0.2731 3.532 0.000413 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##              exp(coef) exp(-coef) lower .95 upper .95
## factor(con.cabg)1      2.624      0.3812      1.536      4.481
##
## Concordance= 0.602 (se = 0.036 )
## Likelihood ratio test= 12.03 on 1 df,  p=5e-04
## Wald test               = 12.47 on 1 df,  p=4e-04
## Score (logrank) test = 13.45 on 1 df,  p=2e-04
```

The hazard ratio for con.cabg is 2.624, with a confidence interval of (1.536, 4.481), indicating that the hazard for the event is significantly higher for the group where con.cabg equals 1 compared to the reference group. The coefficient for con.cabg is 0.9646 with a standard error of 0.2731. The p-value for con.cabg is 0.000413, which is highly significant ( $p < 0.001$ ), indicating a strong effect of con.cabg on the survival outcome.

```
model3 <- coxph(Surv(fuyrs,status) ~ factor(lv),data=hv)
summary(model3)
```

```
## Call:
## coxph(formula = Surv(fuyrs, status) ~ factor(lv), data = hv)
##
##      n= 255, number of events= 54
##
##              coef exp(coef) se(coef)      z Pr(>|z|)
## factor(lv)2 0.4991      1.6473   0.2972 1.679  0.0931 .
## factor(lv)3 1.0953      2.9901   0.4090 2.678  0.0074 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##              exp(coef) exp(-coef) lower .95 upper .95
## factor(lv)2      1.647      0.6071      0.920      2.950
## factor(lv)3      2.990      0.3344      1.341      6.665
```

```
##
## Concordance= 0.597 (se = 0.038 )
## Likelihood ratio test= 7.13 on 2 df, p=0.03
## Wald test = 7.85 on 2 df, p=0.02
## Score (logrank) test = 8.38 on 2 df, p=0.02
```

Patients with a moderate ejection fraction have a hazard ratio of 1.6473 (95% CI: 0.920 to 2.950), indicating a 64.7% higher hazard of the event compared to those with a high ejection fraction, though this is not statistically significant ( $p = 0.0931$ ). Patients with a low ejection fraction have a hazard ratio of 2.9901 (95% CI: 1.341 to 6.665), meaning they are almost three times more likely to experience the event compared to those with a high ejection fraction. This result is statistically significant ( $p = 0.0074$ ). Pre-operative left ventricular ejection fraction is a significant predictor of survival, with patients having a low ejection fraction showing a significantly higher hazard of the event compared to those with a high ejection fraction.

```
model4 <- coxph(Surv(fuyrs,status) ~ factor(sten.reg.mix),data=hv)
summary(model4)
```

```
## Call:
## coxph(formula = Surv(fuyrs, status) ~ factor(sten.reg.mix), data = hv)
##
## n= 255, number of events= 54
##
##              coef exp(coef) se(coef)      z Pr(>|z|)
## factor(sten.reg.mix)2 -0.8608    0.4228   0.4378 -1.966   0.0493 *
## factor(sten.reg.mix)3 -0.8453    0.4294   0.4371 -1.934   0.0531 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##              exp(coef) exp(-coef) lower .95 upper .95
## factor(sten.reg.mix)2    0.4228     2.365    0.1793    0.9972
## factor(sten.reg.mix)3    0.4294     2.329    0.1823    1.0114
##
## Concordance= 0.585 (se = 0.034 )
## Likelihood ratio test= 7.75 on 2 df, p=0.02
## Wald test = 6.75 on 2 df, p=0.03
## Score (logrank) test = 7.17 on 2 df, p=0.03
```

Regurgitation is associated with a 57.72% reduction in the hazard of the event occurring compared to stenosis. This effect is statistically significant ( $p < 0.05$ ). Mixed hemodynamics is associated with a 57.06% reduction in the hazard of the event occurring compared to stenosis. This effect is marginally significant ( $p = 0.05$ ). The Cox proportional hazards model shows that patients with regurgitation and mixed heart valve hemodynamics have significantly lower hazards compared to those with stenosis.

**3. development of the “basic” predictive model (Cox model) with covariates: sex, age, con.cabg, creat, lv, sten.reg.mix. Without including log.lvmi.**

```

coxphModel <- coxph(Surv(fuyrs,status) ~ factor(sex)+age+factor(con.cabg)+
  creat+factor(lv)+factor(sten.reg.mix),data=hv)
summary(coxphModel)

## Call:
## coxph(formula = Surv(fuyrs, status) ~ factor(sex) + age + factor(con.cabg) +
##       creat + factor(lv) + factor(sten.reg.mix), data = hv)
##
##      n= 255, number of events= 54
##
##              coef exp(coef)  se(coef)      z Pr(>|z|)
## factor(sex)1      0.274140  1.315399  0.322403  0.850 0.395157
## age                0.098615  1.103641  0.019433  5.075 3.88e-07 ***
## factor(con.cabg)1  0.663279  1.941147  0.295830  2.242 0.024955 *
## creat              0.006674  1.006696  0.004469  1.493 0.135353
## factor(lv)2        0.599564  1.821325  0.306867  1.954 0.050722 .
## factor(lv)3        1.649344  5.203566  0.446953  3.690 0.000224 ***
## factor(sten.reg.mix)2 -0.627833  0.533747  0.468134 -1.341 0.179876
## factor(sten.reg.mix)3 -0.813409  0.443344  0.448404 -1.814 0.069676 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##              exp(coef) exp(-coef) lower .95 upper .95
## factor(sex)1          1.3154      0.7602      0.6992      2.474
## age                   1.1036      0.9061      1.0624      1.146
## factor(con.cabg)1      1.9411      0.5152      1.0870      3.466
## creat                 1.0067      0.9933      0.9979      1.016
## factor(lv)2            1.8213      0.5491      0.9981      3.323
## factor(lv)3            5.2036      0.1922      2.1670     12.495
## factor(sten.reg.mix)2   0.5337      1.8735      0.2132      1.336
## factor(sten.reg.mix)3   0.4433      2.2556      0.1841      1.068
##
## Concordance= 0.784 (se = 0.034 )
## Likelihood ratio test= 70.16 on 8 df,  p=5e-12
## Wald test              = 48.23 on 8 df,  p=9e-08
## Score (logrank) test = 56.61 on 8 df,  p=2e-09

```

The concordance of 0.784 is quite high, indicating that the model has a good ability to discriminate between patients who die and those who survive.

Both the Likelihood Ratio Test and the Wald Test, as well as the Score Test, have very high values, indicating that the model is highly significant. This means that the covariates are useful in predicting the risk of death and in discriminating patients into different groups.

In particular, age and left ventricular function are the strongest predictors: these factors have a significant impact on the risk of death. The coefficient for age is 0.098615, with an  $\exp(\text{coef})$  of 1.103641, indicating that for each unit increase in age, the hazard increases by approximately 10%, which is consistent with our expectations (i.e., older patients are more likely to have cardiovascular problems). Regarding lv, the left ventricular ejection fraction has an HR of approximately 5.2 at the third level (the low one) and has high significance. Also in the case of moderate dysfunction the hazard ratio is pretty high (around 1.82). Although it is less significant than lv3, the significance is still relevant.

Less relevant factors are sex, creatinine, and the different types of hemodynamics of the heart valve. In the case of sex, for example, the hazard ratio is 1.3154, suggesting that females (coded as 1) have a 31% higher

risk than males, but this result is not statistically significant. Regarding creatinine, instead, p is around 0.135, indicating non significance. The hazard ratio is close to 1, indicating a minimal effect.

4. evaluate functional form of continuous variables and assumption “Proportional Hazards” for all covariates developing “augmented” predictive model (Cox model) with covariates: sex, age, con.cabg, creat, lv, sten.reg.mix and including log.lvmi.

```
model_spline1 <- coxph(Surv(fuyrs, status) ~ factor(sex) + age + factor(con.cabg) +
  creat + factor(lv) + factor(sten.reg.mix) + log.lvmi, data = hv)
summary(model_spline1)
```

```
## Call:
## coxph(formula = Surv(fuyrs, status) ~ factor(sex) + age + factor(con.cabg) +
##       creat + factor(lv) + factor(sten.reg.mix) + log.lvmi, data = hv)
##
##      n= 255, number of events= 54
##
##              coef exp(coef)  se(coef)      z Pr(>|z|)
## factor(sex)1      0.393916  1.482776  0.327247  1.204 0.228696
## age              0.098306  1.103300  0.019695  4.991 5.99e-07 ***
## factor(con.cabg)1  0.748436  2.113692  0.300990  2.487 0.012898 *
## creat            0.003404  1.003410  0.004717  0.722 0.470500
## factor(lv)2       0.617591  1.854456  0.311090  1.985 0.047117 *
## factor(lv)3       1.665226  5.286868  0.445590  3.737 0.000186 ***
## factor(sten.reg.mix)2 -0.690541  0.501305  0.474251 -1.456 0.145374
## factor(sten.reg.mix)3 -0.937145  0.391745  0.448879 -2.088 0.036821 *
## log.lvmi          1.229339  3.418968  0.389191  3.159 0.001585 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##              exp(coef) exp(-coef) lower .95 upper .95
## factor(sex)1      1.4828      0.6744      0.7808      2.8160
## age              1.1033      0.9064      1.0615      1.1467
## factor(con.cabg)1  2.1137      0.4731      1.1718      3.8128
## creat            1.0034      0.9966      0.9942      1.0127
## factor(lv)2       1.8545      0.5392      1.0079      3.4121
## factor(lv)3       5.2869      0.1891      2.2076     12.6615
## factor(sten.reg.mix)2  0.5013      1.9948      0.1979      1.2699
## factor(sten.reg.mix)3  0.3917      2.5527      0.1625      0.9443
## log.lvmi          3.4190      0.2925      1.5945      7.3312
##
## Concordance= 0.796 (se = 0.032 )
## Likelihood ratio test= 79.89 on 9 df,  p=2e-13
## Wald test              = 58.77 on 9 df,  p=2e-09
## Score (logrank) test = 67.03 on 9 df,  p=6e-11
```

As before, the covariates that mostly impact and discriminate the model are age and left ventricular function. Focusing on age, an HR of 2.33 indicates that with increase in age, the risk of death more than doubles. But

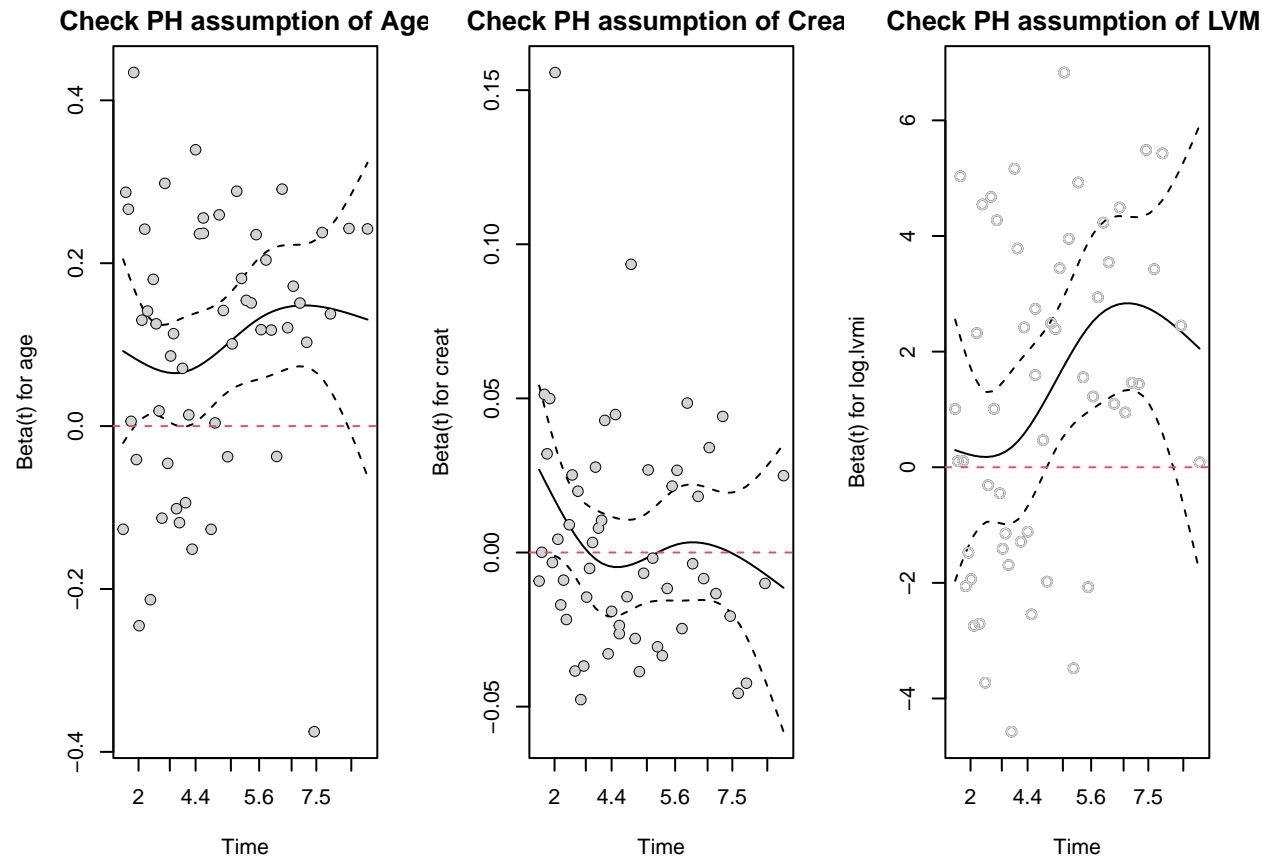
now we consider also the natural logarithm of the Left Ventricular Mass Index, which has relevance as well, having an hazard ratio of 3.37. As before, sex and creatinine result to be not significant.

Regarding Concordance Index and the test values, they are still high and significant, suggesting that the model is a good fit for the data and has a good discriminative ability between patients who die and those who survive

```
par(mfrow=c(1,3),mar=c(4,4,2,2))
checkPH.age<-cox.zph(model_spline1)[2]
plot(checkPH.age,main="Check PH assumption of Age")
points(checkPH.age$x,checkPH.age$y,pch=16,col="lightgray")
abline(h=0,lty=2,col=2)

checkPH.creat<-cox.zph(model_spline1)[4]
plot(checkPH.creat,main="Check PH assumption of Creat")
points(checkPH.creat$x,checkPH.creat$y,pch=16,col="lightgray")
abline(h=0,lty=2,col=2)

checkPH.log.lvmi<-cox.zph(model_spline1)[7]
plot(checkPH.log.lvmi,main="Check PH assumption of LVMI")
points(checkPH.log.lvmi$x,checkPH.log.lvmi$y,col="lightgray")
abline(h=0,lty=2,col=2)
```



These plots are Schoenfeld residual plots, and they assess the PH assumption checking if the covariate effects are time-independent, i.e if the coefficients for each covariate are constant over time.

In this case, all the plots show some deviation from the horizontal line, suggesting potential violations of the proportional hazards assumption. This means that those variables may have a nonconstant effect across time, and that can cause problems due to the fact that Cox assumes the HR to be constant over time

```
model_spline2 <- coxph(Surv(fuyrs, status) ~ sex + age + con.cabg + creat + lv +
                        sten.reg.mix + log.lvmi, data = hv)
summary(model_spline2)
```

```
## Call:
## coxph(formula = Surv(fuyrs, status) ~ sex + age + con.cabg +
##       creat + lv + sten.reg.mix + log.lvmi, data = hv)
##
##      n= 255, number of events= 54
##
##              coef exp(coef)    se(coef)      z Pr(>|z|)
## sex           0.336916  1.400622  0.317502  1.061 0.288624
## age           0.097647  1.102573  0.019109  5.110 3.22e-07 ***
## con.cabg      0.714460  2.043083  0.291454  2.451 0.014232 *
## creat         0.003279  1.003285  0.004686  0.700 0.483988
## lv            0.758255  2.134548  0.210514  3.602 0.000316 ***
## sten.reg.mix -0.519170  0.595014  0.216555 -2.397 0.016512 *
## log.lvmi      1.227546  3.412844  0.388375  3.161 0.001574 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##              exp(coef) exp(-coef) lower .95 upper .95
## sex              1.401      0.7140    0.7517    2.6096
## age              1.103      0.9070    1.0620    1.1446
## con.cabg         2.043      0.4895    1.1540    3.6172
## creat            1.003      0.9967    0.9941    1.0125
## lv               2.135      0.4685    1.4129    3.2247
## sten.reg.mix     0.595      1.6806    0.3892    0.9096
## log.lvmi         3.413      0.2930    1.5942    7.3064
##
## Concordance= 0.798 (se = 0.031 )
## Likelihood ratio test= 79.25 on 7 df,  p=2e-14
## Wald test              = 59.14 on 7 df,  p=2e-10
## Score (logrank) test = 66.74 on 7 df,  p=7e-12
```

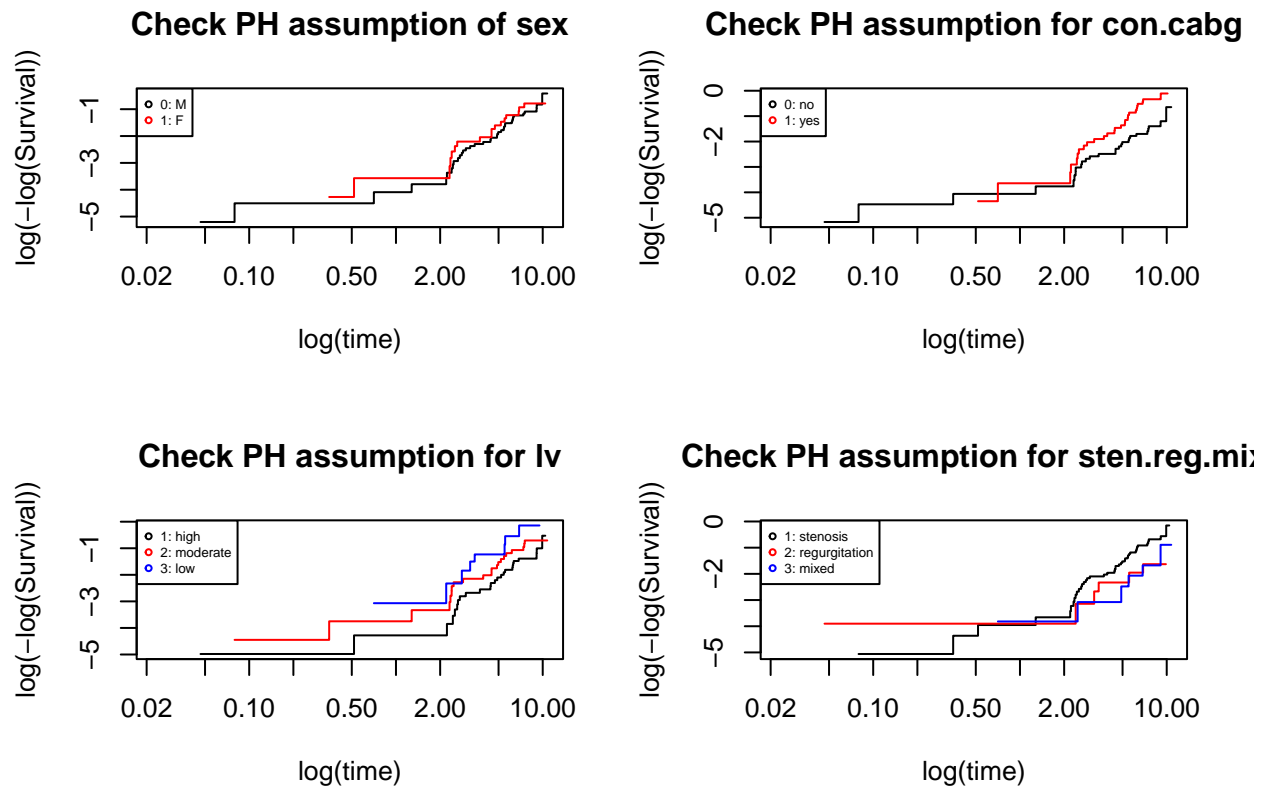
```
km.sex <- survfit(Surv(fuyrs, status) ~ factor(sex), data = hv)
km.con.cabg <- survfit(Surv(fuyrs, status) ~ factor(con.cabg), data = hv)
km.lv <- survfit(Surv(fuyrs, status) ~ factor(lv), data = hv)
km.sten.reg.mix <- survfit(Surv(fuyrs, status) ~ factor(sten.reg.mix), data = hv)
par(mfrow = c(2, 2))
plot(km.sex, col=c("black", "red"), fun="cloglog", ylab="log(-log(Survival))",
     xlab="log(time)", main="Check PH assumption of sex")
legend("topleft", legend = c("0: M", "1: F"), col = c("black", "red"), pch = 1, cex = 0.5)
plot(km.con.cabg, col = c("black", "red"), fun = "cloglog", ylab = "log(-log(Survival))",
     xlab = "log(time)", main = "Check PH assumption for con.cabg")
```



```

legend("topleft", legend = c("0: no", "1: yes"), col = c("black", "red"), pch = 1,
      cex = 0.5)
plot(km.lv, col = c("black", "red", "blue"), fun = "cloglog", ylab = "log(-log(Survival))",
      xlab = "log(time)", main = "Check PH assumption for lv")
legend("topleft", legend = c("1: high", "2: moderate", "3: low"),
      col = c("black", "red", "blue"), pch = 1, cex = 0.5)
plot(km.sten.reg.mix, col = c("black", "red", "blue"), fun = "cloglog",
      ylab = "log(-log(Survival))", xlab = "log(time)",
      main = "Check PH assumption for sten.reg.mix")
legend("topleft", legend = c("1: stenosis", "2: regurgitation", "3: mixed"),
      col = c("black", "red", "blue"), pch = 1, cex = 0.5)

```



In this case, the plots represent cloglog curves of variables sex, con.cabg and lv and sten.reg.mix. These graphs are used to verify the proportional hazards (PH) assumption in the Cox model. If the curves for different levels of a covariate are parallel or tend to be so, the PH assumption is satisfied.

Thus I have to check if the survival curves for different categories of a predictor variable do not intersect or significantly overlap, because this suggests that the effect of that variable on the logarithm of the instantaneous hazard is constant over time, supporting the assumption of proportional hazards. On the other hand, if the curves overlap significantly, it may indicate that the effect of that variable is not constant over time, and thus the assumption of proportional hazards may not be met for that variable.

In the first two cases we have just two curves, because there are two possible outcomes for the variables, in the plots below we can have three outcomes so we have three curves, indicated with black, red and blue colors

By checking for potential overlapping, I can conclude that in the case of the covariates sex and lv, there is no significant overlapping, and if present, it's always towards the extremes: the function that is greater at one

point tends to be so almost throughout the graph. On the other hand, `con.cabg` shows no overlapping except for some overlap of the functions for a certain stretch. `sten.reg.mix`, on the other hand, is the covariate that behaves “worse”, exhibiting significant overlapping among all three of its curves, hence among all three possible outcomes of the variable.

In conclusion, despite some covariates showing potential violations of the proportional hazards assumption, the overall model appears to be robust, and for this model the predictors displayed are highly significant.

## 5. evaluate functional form of continuous variables and “Proportional Hazards” assumption (especially for `log.lvmi`).

We are going now to fit the Cox proportional hazards model to evaluate the association between `log.lvmi` (natural logarithm of left ventricular mass index) and the survival outcome.

```
model_spline3 <- coxph(Surv(fuyrs, status) ~ log.lvmi, data = hv)
summary(model_spline3)
```

```
## Call:
## coxph(formula = Surv(fuyrs, status) ~ log.lvmi, data = hv)
##
##    n= 255, number of events= 54
##
##              coef exp(coef) se(coef)      z Pr(>|z|)
## log.lvmi 0.9090      2.4818   0.3514 2.587  0.00968 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##              exp(coef) exp(-coef) lower .95 upper .95
## log.lvmi      2.482      0.4029      1.246      4.942
##
## Concordance= 0.585  (se = 0.041 )
## Likelihood ratio test= 6.64  on 1 df,   p=0.01
## Wald test              = 6.69  on 1 df,   p=0.01
## Score (logrank) test = 6.6  on 1 df,   p=0.01
```

The coefficient for `log.lvmi` is 0.9090, indicating that for each unit increase in the logarithm of left ventricular mass index, the hazard of experiencing the event, in this case mortality, increases by 90.9%.

The hazard ratio (`exp(coef)`) associated with `log.lvmi` is 2.482, suggesting that people with higher left ventricular mass indices are 2.482 times more likely to experience the event compared to those with lower indices, having other variables constant.

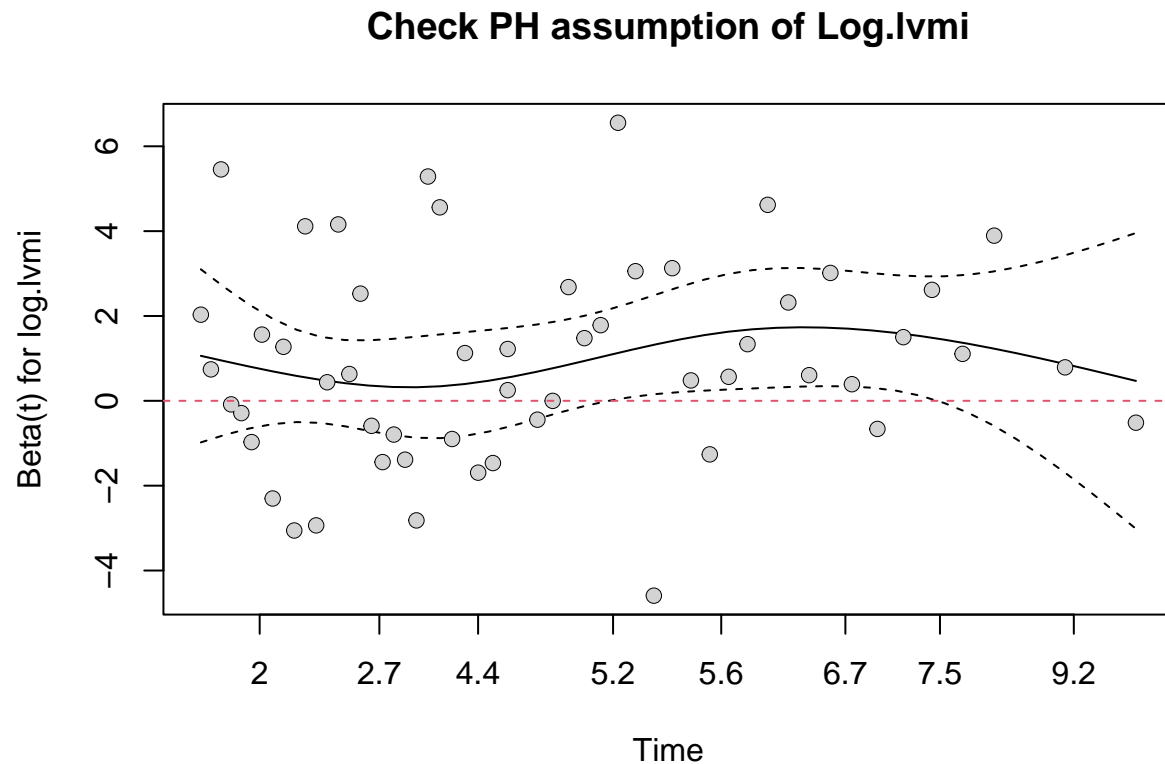
The significance of `log.lvmi` is demonstrated by a p-value of 0.00968, indicating that it is statistically significant at the alpha level of 0.05.

Next, we look at the concordance index, which measures the model’s ability to correctly rank the relative risk of individuals. The model achieved a concordance index of 0.585, indicating moderate predictive discrimination.

Both the likelihood ratio test and the Wald test have significant p-values ( $p=0.01$ ), suggesting that the model significantly improves the fit compared to a null model where `log.lvmi` is not included.

The significant association between log.lvmi and the survival outcome suggests that left ventricular mass index is a relevant predictor of mortality risk in the studied population. Clinically, this highlights the importance of assessing left ventricular mass as a potential marker for adverse cardiovascular outcomes.

```
checkPH.lvmi<-cox.zph(model_spline3)
plot(checkPH.lvmi,main="Check PH assumption of Log.lvmi")
points(checkPH.lvmi$x,checkPH.lvmi$y,pch=16,col="lightgray")
abline(h=0,lty=2,col=2)
```



To check the proportional hazards assumption for this model for the variable log.lvmi, we use the Schoenfeld plot as done previously. Here too, the solid line deviates significantly from the horizontal dashed red line, indicating potential violations of the PH assumption for this variable. Therefore, the effect of the variable on the risk varies over time, making the model inappropriate.

## 6. evaluate and compare the performance of the two models (calibration, discrimination, Net Benefit) for predicting event risk at a fixed time-point (e.g., 5 years).

We are going now to compare the performance of the model we have made in the previous exercises on three factors: Calibration, Discrimination and Net Benefit.

In particular, we have taken in account two Cox proportional hazards models to predict event risk at a fixed time-point of 5 years. Each model includes different combinations of predictor variables, such as demographic factors (sex, age), medical history (con.cabg, creat), and cardiac measures (lv, sten.reg.mix, log.lvmi).

```

# Fit Cox proportional hazards model 1
model_spline1_1 <- coxph(Surv(fuyrs, status) ~ factor(sex) + age + factor(con.cabg) +
  creat + factor(lv) + factor(sten.reg.mix), data = hv, x=T)

# Generate survival probabilities based on model 1
fit1 <- survfit(model_spline1_1, newdata = hv)

# Calculate the estimated risk of death at a specific time point (e.g., 5 units of time)
#for model 1
hv$riskdeath1 <- 1 - as.numeric(summary(fit1, times = 5)$surv)

# Fit Cox proportional hazards model 2 (extended model with an additional
#continuous predictor)
model_spline2_1 <- coxph(Surv(fuyrs, status) ~ sex + age + con.cabg + creat + lv +
  sten.reg.mix + log.lvmi, data = hv, x=T)

# Generate survival probabilities based on model 2
fit2 <- survfit(model_spline2_1, newdata = hv)

# Calculate the estimated risk of death at the same specific time point for model 2
hv$riskdeath2 <- 1 - as.numeric(summary(fit2, times = 5)$surv)

```

We compare now the two models

```

# Risk quantiles of 1
q1 <- quantile(hv$riskdeath1, probs = seq(0.1, 0.9, 0.1))

# Risk classes of 1:
riskclass1 <- ifelse(hv$riskdeath1 < q1[1], 1,
  ifelse(hv$riskdeath1 < q1[2], 2,
    ifelse(hv$riskdeath1 < q1[3], 3,
      ifelse(hv$riskdeath1 < q1[4], 4,
        ifelse(hv$riskdeath1 < q1[5], 5,
          ifelse(hv$riskdeath1 < q1[6], 6,
            ifelse(hv$riskdeath1 < q1[7], 7,
              ifelse(hv$riskdeath1 < q1[8], 8,
                ifelse(hv$riskdeath1 < q1[9], 9,
                  10))))))))))

# Observed proportion of events in risk classes of 1:
obsclass1 <- c()
for (i in 1:10) {
  group <- i
  ratio <- sum(hv$status[riskclass1 == group]) / sum(riskclass1 == group)
  obsclass1 <- c(obsclass1, ratio)
}

# Average predicted risk in classes of 1:
meanriskclass1 <- c()
for (i in 1:10) {
  group <- i
  meanrisk <- mean(hv$riskdeath1[riskclass1 == group])
}

```

```

    meanriskclass1 <- c(meanriskclass1, meanrisk)
  }

  # Risk quantiles of 2
  q2 <- quantile(hv$riskdeath2, probs = seq(0.1, 0.9, 0.1))

  # Risk classes of 2:
  riskclass2 <- ifelse(hv$riskdeath2 < q2[1], 1,
    ifelse(hv$riskdeath2 < q2[2], 2,
      ifelse(hv$riskdeath2 < q2[3], 3,
        ifelse(hv$riskdeath2 < q2[4], 4,
          ifelse(hv$riskdeath2 < q2[5], 5,
            ifelse(hv$riskdeath2 < q2[6], 6,
              ifelse(hv$riskdeath2 < q2[7], 7,
                ifelse(hv$riskdeath2 < q2[8], 8,
                  ifelse(hv$riskdeath2 < q2[9], 9,
                    10))))))))))
  # Observed proportion of events in risk classes of 2:
  obsclass2 <- c()
  for (i in 1:10) {
    group <- i
    ratio <- sum(hv$status[riskclass2 == group]) / sum(riskclass2 == group)
    obsclass2 <- c(obsclass2, ratio)
  }

  # Average predicted risk in classes of 2:
  meanriskclass2 <- c()
  for (i in 1:10) {
    group <- i
    meanrisk <- mean(hv$riskdeath2[riskclass2 == group])
    meanriskclass2 <- c(meanriskclass2, meanrisk)
  }

  # Calibration plot
  plot(meanriskclass1, obsclass1, main = '', pch = 1, xlim = c(0,1), cex = 1.5,
    ylim = c(0,1), lwd = 3, ylab = 'Observed event rate',
    xlab = 'Average risk within decile', xaxt = "n", yaxt = "n", frame = F)
  axis(2, at = c(0,0.2,0.4,0.6,0.8,1), labels = NA, pos = 0)
  axis(2, at = c(0,0.2,0.4,0.6,0.8,1), labels = c(0,0.2,0.4,0.6,0.8,1), pos = 0)
  axis(1, at = c(0,0.2,0.4,0.6,0.8,1), labels = c(0,0.2,0.4,0.6,0.8,1), pos = 0)

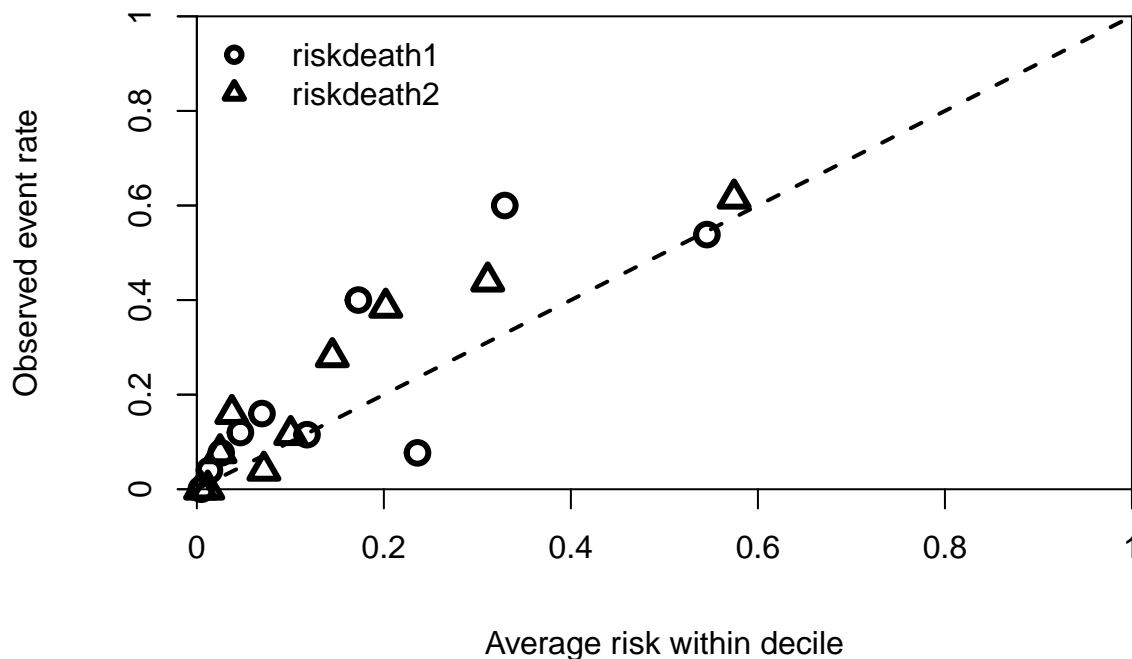
  points(meanriskclass2, obsclass2, lwd = 3, pch = 2, cex = 1.5)

  lines(c(0, 1), c(0, 1), lty = 2, lwd = 2)

  lines(c(0,1), c(1,1), lty = 1)
  lines(c(1,1), c(0,1), lty = 1)

  legend(x = 0, y = 1, c("riskdeath1", "riskdeath2"), lwd = c(3, 3), pch = c(1, 2),
    bty = 'n', lty = c(NA, NA))

```



As we can see, the calibration of the two models is far from perfect, but it is acceptable. There are, of course, some problems with the predicted risks of the two models, like for instance in correspondence to the average risk equal to 20%, we have that the event is going to have a higher percentage than 20% (for a data point of the second model we have an event probability equal to 40%).

We now analyze the discrimination (C-index). Discrimination, in the context of predictive modeling and ROC curves, refers to the ability of a model to distinguish between individuals who experience an event (in this case death) and those who do not.

We use a ROC curve plot. The ROC curve plots the true positive rate (sensitivity) against the false positive rate (1 - specificity) for different threshold values of a model's predicted probabilities.

```
roc1<-roc(hv$status, hv$riskdeath1)

## Setting levels: control = 0, case = 1

## Setting direction: controls < cases

plot(1 - roc1$specificities, roc1$sensitivities,
     type = 'l', ylab = 'TPF', xlab = 'FPF', lwd = 3, xaxt = "n", yaxt = "n",
     xlim = c(0,1), frame = F)
axis(1, at = c(0,0.25,0.5,0.75,1), labels = NA, pos = 0)
axis(1, at = c(0,0.25,0.5,0.75,1), labels = c(0,0.25,0.5,0.75,1), pos = 0)
axis(2, at = c(0,0.25,0.5,0.75,1), labels = c(0,0.25,0.5,0.75,1), pos = 0)
Youden1<-roc1$sensitivities+roc1$specificities-1
optimal.cut.off1<-roc1$thresholds[Youden1==max(Youden1)]
cbind(optimal.cut.off1,Youden1=max(Youden1))
```

```
##      optimal.cut.off1      Youden
## [1,]      0.1587072 0.4806523
```

```
points(1-roc1$specificities[roc1$thresholds==optimal.cut.off1],
       roc1$sensitivities[roc1$thresholds==optimal.cut.off1],pch=0)
```

```
roc2<-roc(hv$status, hv$riskdeath2)
```

```
## Setting levels: control = 0, case = 1
## Setting direction: controls < cases
```

```
lines(1 - roc2$specificities, roc2$sensitivities,
      lwd = 3, lty = 3)
Youden2<-roc2$sensitivities+roc2$specificities-1
optimal.cut.off2<-roc2$thresholds[Youden2==max(Youden2)]
cbind(optimal.cut.off2,Youden=max(Youden2))
```

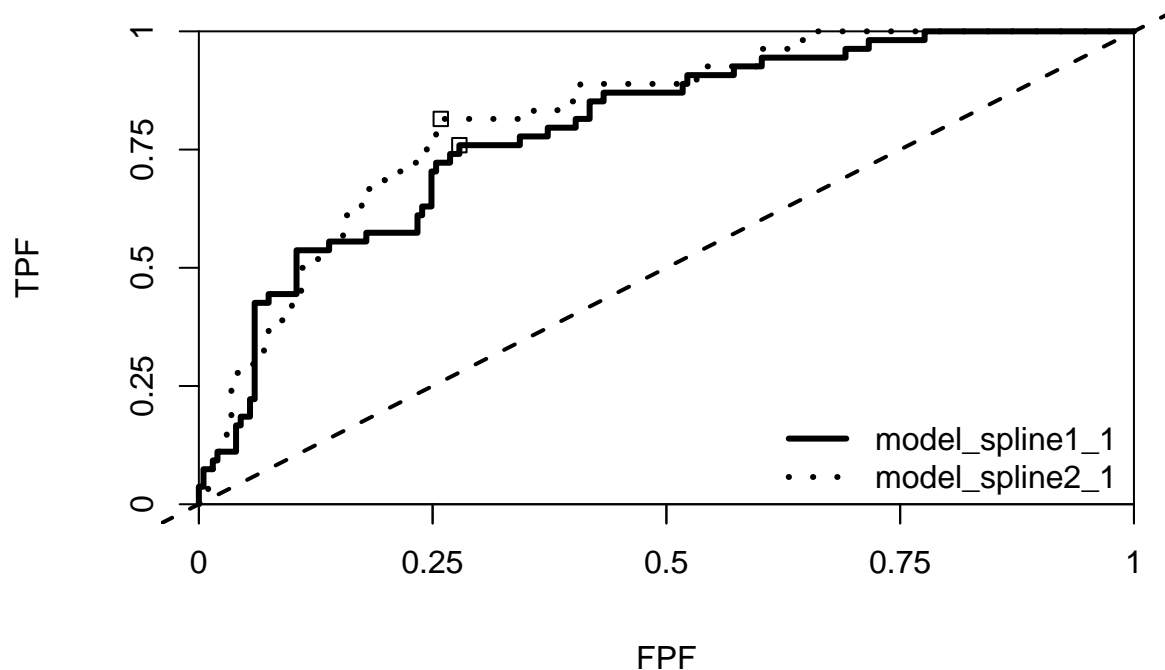
```
##      optimal.cut.off2      Youden
## [1,]      0.1344332 0.5561083
```

```
points(1-roc2$specificities[roc2$thresholds==optimal.cut.off2],
       roc2$sensitivities[roc2$thresholds==optimal.cut.off2],pch=0)
```

```
# Chance line:
```

```
abline(a=0, b=1, lty = 2, lwd = 2)
lines(c(0,1), c(1,1), lty = 1)
lines(c(1,1), c(0,1), lty = 1)
```

```
legend(x = 0.6, y = 0.22, c("model_spline1_1","model_spline2_1"), lwd = c(3,3),
      lty = c(1,3), bty = 'n', )
```



As it is possible to notice, the second model manages to have a better discrimination than the first model and we can tell this by simply looking at the curves' heights: in particular, we have that the second model is slightly higher than the first one. The optimal cut-off are basically around 25% of false positive rates. We can also have a look at the area under the curve to see whether our initial assumption was right:

```
(AUC1 <- roc1$auc)
```

```
## Area under the curve: 0.793
```

```
(AUC2 <- roc2$auc)
```

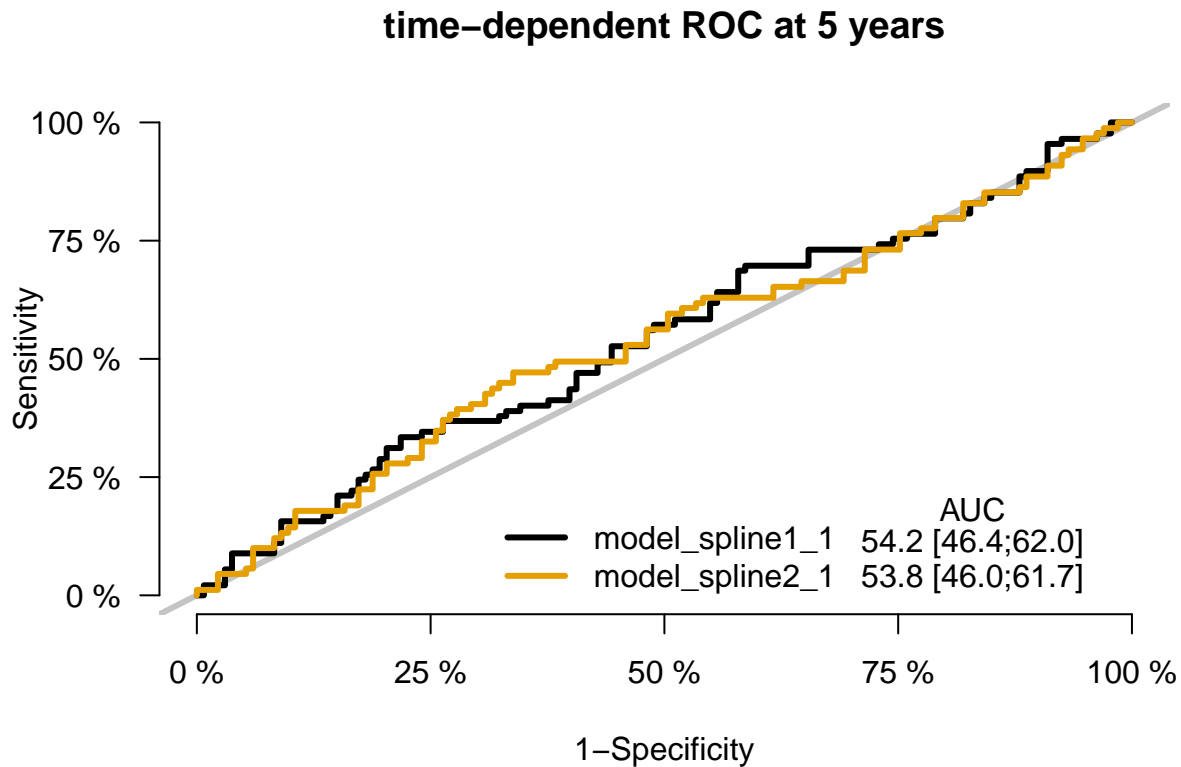
```
## Area under the curve: 0.8188
```

As we can notice, we were right: the AUC of the second model is slightly higher than the first model's, meaning that the model with `log.lvmi` manages to discriminate better between individuals who will experience death and those who will not in a span of 5 years.

```
score<- Score(list("model_spline1_1"=model_spline1_1,"model_spline2_1"=model_spline2_1),
  formula=Surv(fuyrs, status==0)~1,
  data=hv,conf.int=T,
  times=seq(1,5,1),
  plots=c("calibration","ROC"))
```



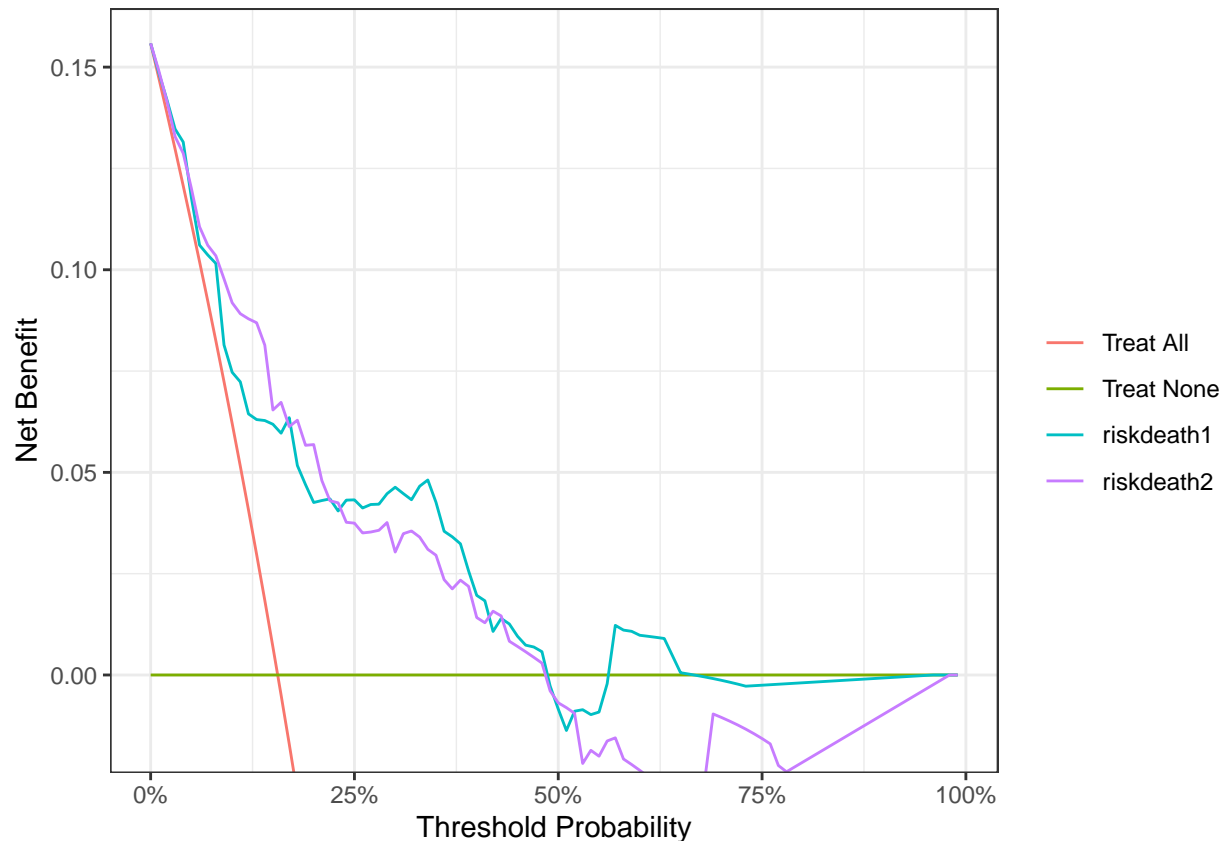
```
plotROC(score,times=5,cens.method="local")
title(main="time-dependent ROC at 5 years")
```



The net benefit is used in decision curve analysis to evaluate the usefulness of a certain predictive model compared to others (e.g., treating all or none of the patients). It takes into account both the true positives and false positives at different threshold probabilities for a binary outcome. It uses the package `dcurves` so let's load it and let's plot the decision curve analysis (DCA)

```
dca(Surv(fuyrs, status==1) ~ riskdeath1 + riskdeath2, data = hv, time = 5)
```

```
## Printing with `plot(x, type = 'net_benefit', smooth = FALSE, show_ggplot_code = FALSE)`
```



The red curve is the strategy in which we treat every patient so basically here the cost of treatment is equal to 0. The other two curves mean that the cost of treatment are dependent on the threshold. In particular, we can see that the two models are particular useful between about 15% and 50% of the threshold probability, otherwise the net benefit of the two models is basically equal to zero, therefore we do not gain any benefit from them.

**7. prediction of event risk at a fixed time-point (e.g., 5 years) for 3 “type” subjects (randomly chosen in the dataset or new hypothetical subjects) based on the base model and the augmented model.**

```
set.seed(123) # we settle the seed for the reproducibility
subjects1 <- sample_n(hv,3) #here we choose a sample of
subjects <- data.frame(
  paz.id = c(1001, 1002, 1004), log.lvmi = c(2.3, 2.8, 2.1),
  fuysr = c(5, 5, 5), status = c(0, 1, 0),
  sex = c(1, 0, 1), age = c(30, 50, 70),
  con.cabg = c(1, 1, 0), creat = c(55, 65, 150),
  lv = c(2, 1, 3), sten.reg.mix = c(1, 2, 2)
)

# function to predict the risk event
predict_event_risk <- function(model, subjects, time_point = 5) {
```

```

surv_prob <- survfit(model, newdata = subjects)
surv_at_time <- summary(surv_prob, times = time_point)$surv
event_risk <- 1 - surv_at_time
return(event_risk)
}

# Prediction of the event based on a fake data
event_risk_base_fake <- predict_event_risk(model_spline1, subjects)
event_risk_augmented_fake <- predict_event_risk(model_spline2, subjects)

#Prediction of the event based on a sample
event_risk_base_sample <- predict_event_risk(model_spline1, subjects1)
event_risk_augmented_sample <- predict_event_risk(model_spline2, subjects1)

fakeMatrix <- matrix(c(round(event_risk_base_fake, digits = 9),
                        round(event_risk_augmented_fake, digits = 9),
                        round((event_risk_base_fake-event_risk_augmented_fake),
                              digits = 9)), nrow = 3, byrow =TRUE)
rownames(fakeMatrix) <- c("Base", "Augmented", "Difference")
cat("Fake:\n"); fakeMatrix

```

## Fake:

```

##           [,1]      [,2]      [,3]
## Base      0.000240513  0.000599070  0.008926958
## Augmented  0.000259460  0.000694742  0.008526894
## Difference -0.000018946 -0.000095672  0.000400064

```

```

sampleMatrix <- matrix(c(round(event_risk_base_sample, digits = 9),
                        round(event_risk_augmented_sample, digits = 9),
                        round((event_risk_base_sample-event_risk_augmented_sample),
                              digits = 9)), nrow = 3, byrow =TRUE)
rownames(sampleMatrix) <- c("Base", "Augmented", "Difference")
cat("\nSample:\n"); sampleMatrix

```

##

## Sample:

```

##           [,1]      [,2]      [,3]
## Base      0.133387276  0.011433018  0.28106367
## Augmented  0.127543018  0.011454535  0.30319726
## Difference 0.005844258 -0.000021518 -0.02213359

```

Predictions are nearly identical between the base and augmented models. Differences are minuscule, indicating the augmentation did not significantly affect the predictions.

The augmented model predicts slightly lower risks compared to the base model for the sample data. Differences are small but noticeable, indicating potential improvements from the augmentation. Overall, predictions are consistent, with minor adjustments from the augmented model that could have clinical significance.