Joint models

Jose Calatayud Mateu

2025-05-16

Enunciado

Disponemos de unos datos longitudinales de un estudio sobre la detección de efectos de diferentes válvulas cardíacas, que difieren en el tipo de tejido y han sido implantadas en la posición aórtica. Los datos consisten en mediciones longitudinales de tres resultados diferentes de la función cardíaca después de la cirugía. Tenemos información sobre algunas covariables iniciales (fijas) y nos interesa estudiar el tiempo hasta la muerte tras la cirugía (variables "fuyrs" y "status"). Crea un modelo que permita predecir la muerte que dependa de alguna variable dependiente del tiempo ("grad", "lvmi", "ef") teniendo en cuenta otras variables fijas que han sido recogidas al inicio del estudio ("age", "con.cabg", "dm, "acei, "lv", "emergenc", "lvh", "creat", "dm", "hs"). Para contestar a esta pregunta usa un joint model. Ilustra de forma gráfica como este modelo se puede usar para predecir la mortalidad de los tres primeros individuos.

Los datos están disponibles en la librería "joineRML" en el dataset "heart.valve" que pueden cargarse en R mediante:

```
library(joineRML)
```

```
## Loading required package: nlme
```

```
## Loading required package: survival
```

```
library(JM)
```

```
## Loading required package: MASS
```

```
## Loading required package: splines
```

```
library(survminer)
```

```
## Loading required package: ggplot2
```

```
## Loading required package: ggpubr
```

```
## Registered S3 methods overwritten by 'broom':
##   method         from
##   augment.mjoint joineRML
##   glance.mjoint  joineRML
##   tidy.mjoint    joineRML
```

```
##
## Attaching package: 'survminer'

## The following object is masked from 'package:survival':
##
##     myeloma
```

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v lubridate 1.9.3     v tibble    3.2.1
## v purrr     1.0.2     v tidyr     1.3.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::collapse() masks nlme::collapse()
## x dplyr::filter()   masks stats::filter()
## x dplyr::lag()      masks stats::lag()
## x dplyr::select()   masks MASS::select()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
data(heart.valve)
head(heart.valve)
```

```
##   num sex      age      time    fuyrs status grad log.grad   lvmi log.lvmi ef
## 1   1   0 75.06027 0.0109589 4.956164      0   10 2.302585 118.98 4.778955 93
## 2   1   0 75.06027 3.6794520 4.956164      0   10 2.302585 118.98 4.778955 93
## 3   1   0 75.06027 4.6958900 4.956164      0   10 2.302585 137.63 4.924569 93
## 4   2   0 45.79452 6.3643840 9.663014      0   14 2.639057 114.93 4.744323 68
## 5   2   0 45.79452 7.3041100 9.663014      0    9 2.197225 109.80 4.698661 70
## 6   2   0 45.79452 8.3013700 9.663014      0   12 2.484907 157.40 5.058790 56
##    bsa lvh prenyha redo size con.cabg creat dm acei lv emergenc hc sten.reg.mix
## 1 1.77   1       3    0   27        1   103  0    1  1        0  0            1
## 2 1.77   1       3    0   27        1   103  0    1  1        0  0            1
## 3 1.77   1       3    0   27        1   103  0    1  1        0  0            1
## 4 1.92   1       1    1   22        0    76  0    0  2        0  0            1
## 5 1.92   1       1    1   22        0    76  0    0  2        0  0            1
## 6 1.92   1       1    1   22        0    76  0    0  2        0  0            1
##                hs
## 1 Stentless valve
## 2 Stentless valve
## 3 Stentless valve
## 4       Homograft
## 5       Homograft
## 6       Homograft
```

Podemos conocer el significado de cada variable ejecutando:

```
?heart.valve
```

Así el data.frame `heart.valve` está formato no balanceado, es decir, con una fila por cada observación. La base de datos consta de columnas que abarcan la identificación del paciente, el tiempo de las mediciones, múltiples mediciones longitudinales, covariables basales y datos de supervivencia.

A continuación se muestran las variables contenidas:

- **num:** Número de identificación del paciente.
- **sex:** Género del paciente (0 = Masculino y 1 = Femenino).
- **age:** Edad del paciente en el día de la cirugía (en años).
- **time:** Punto temporal observado, considerando la fecha de la cirugía como origen del tiempo (en años).
- **fuyrs:** Tiempo máximo de seguimiento, considerando la fecha de la cirugía como origen del tiempo (en años).
- **status:** Indicador de censura (1 = fallecido y 0 = perdido en el seguimiento).
- **grad:** Gradiente valvular en la visita de seguimiento.
- **log.grad:** Transformación logarítmica natural de *grad*.
- **lvmi:** Índice de masa ventricular izquierda (estandarizado) en la visita de seguimiento.
- **log.lvmi:** Transformación logarítmica natural de *lvmi*.
- **ef:** Fracción de eyección en la visita de seguimiento.
- **bsa:** Superficie corporal preoperatoria.
- **lvh:** Hipertrofia ventricular izquierda preoperatoria.
- **prenyha:** Clasificación preoperatoria de la New York Heart Association (NYHA) (1 = I/II y 3 = III/IV).
- **redo:** Cirugía cardíaca previa.
- **size:** Tamaño de la válvula (en milímetros).
- **con.cabg:** Bypass de la arteria coronaria concomitante.
- **creat:** Creatinina sérica preoperatoria (μmol/mL).
- **dm:** Diabetes preoperatoria.
- **acei:** Uso preoperatorio de inhibidores de la ECA.
- **lv:** Fracción de eyección ventricular izquierda preoperatoria (LVEF) (1 = buena, 2 = moderada, y 3 = pobre).
- **emergenc:** Urgencia operativa (0 = electiva, 1 = urgente, y 3 = emergencia).
- **hc:** Colesterol alto preoperatorio (0 = ausente, 1 = presente tratado, y 2 = presente no tratado).
- **sten.reg.mix:** Hemodinámica de la válvula aórtica (1 = estenosis, 2 = regurgitación, 3 = mixta).
- **hs:** Tipo de prótesis aórtica implantada (1 = homograft y 0 = tejido porcino sin stent).

Si observamos el data que se nos presenta, existen valores faltantes, en consecuencia, podria recurrir a imputar pero en nuestro caso, nos centramos en tan solo trabajar con datos completos:

```
heart.valve <- na.omit(heart.valve)
```

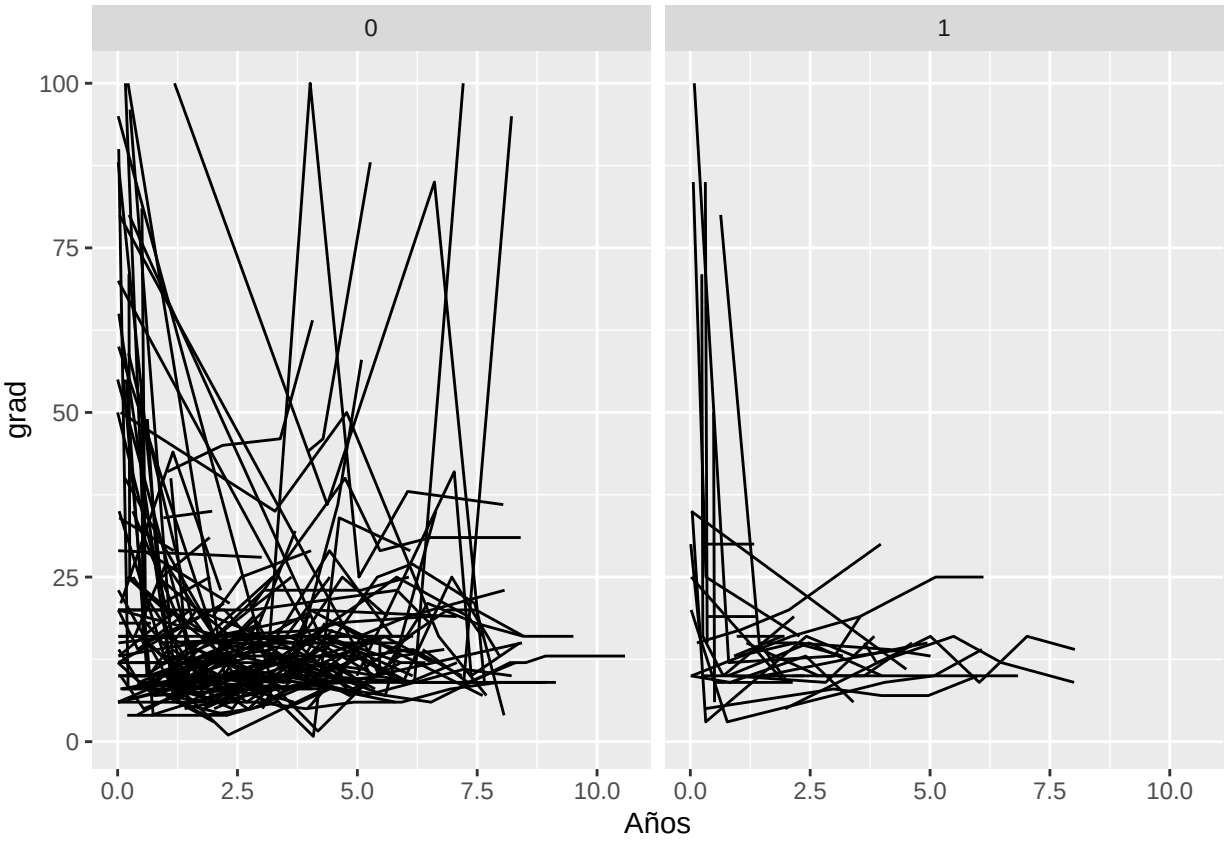
Además, disponemos de variable que deberian estar en formato factor y estan como numéricas, los modelos no pueden entender las categorias de como estan codificadas si no se les indica expresamente que se trata de una variable factor. Así:

```
# Convertir variables categóricas en factor
heart.valve <- heart.valve %>%
  mutate(
    sex = factor(sex),
    lvh = factor(lvh),
    prenyha = factor(prenyha),
    redo = factor(redo),
    con.cabg = factor(con.cabg),
    dm = factor(dm),
    acei = factor(acei),
```

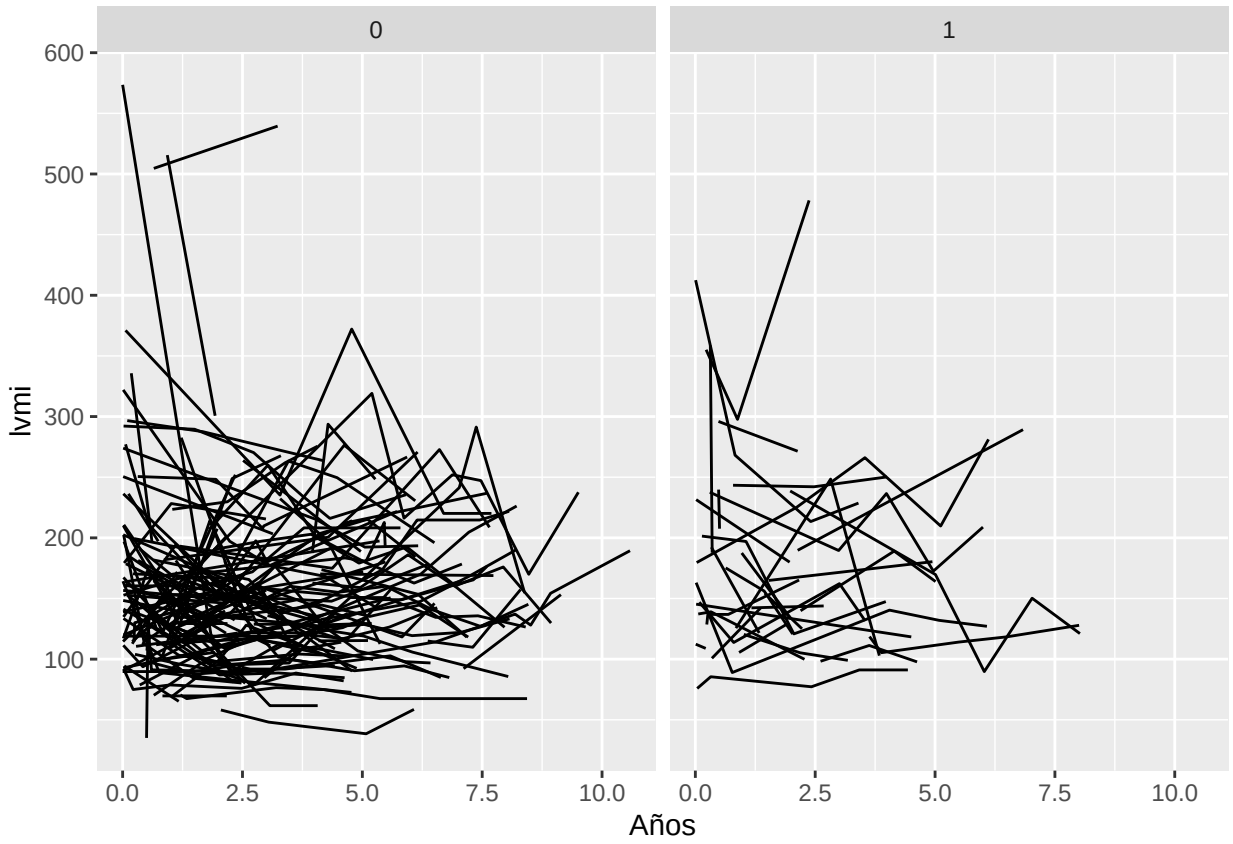
```
    lv = factor(lv),
    emergenc = factor(emergenc),
    hc = factor(hc),
    sten.reg.mix = factor(sten.reg.mix),
    hs = factor(hs)
  )
```

Y como indica el enunciado, tenemos que utilizar alguna de las variables dependientes del tiempo ("grad", "lvmi", "ef") para predecir la muerte. En primer lugar, se nos pide en nuestro modelo que variable dependiente deba introducir.

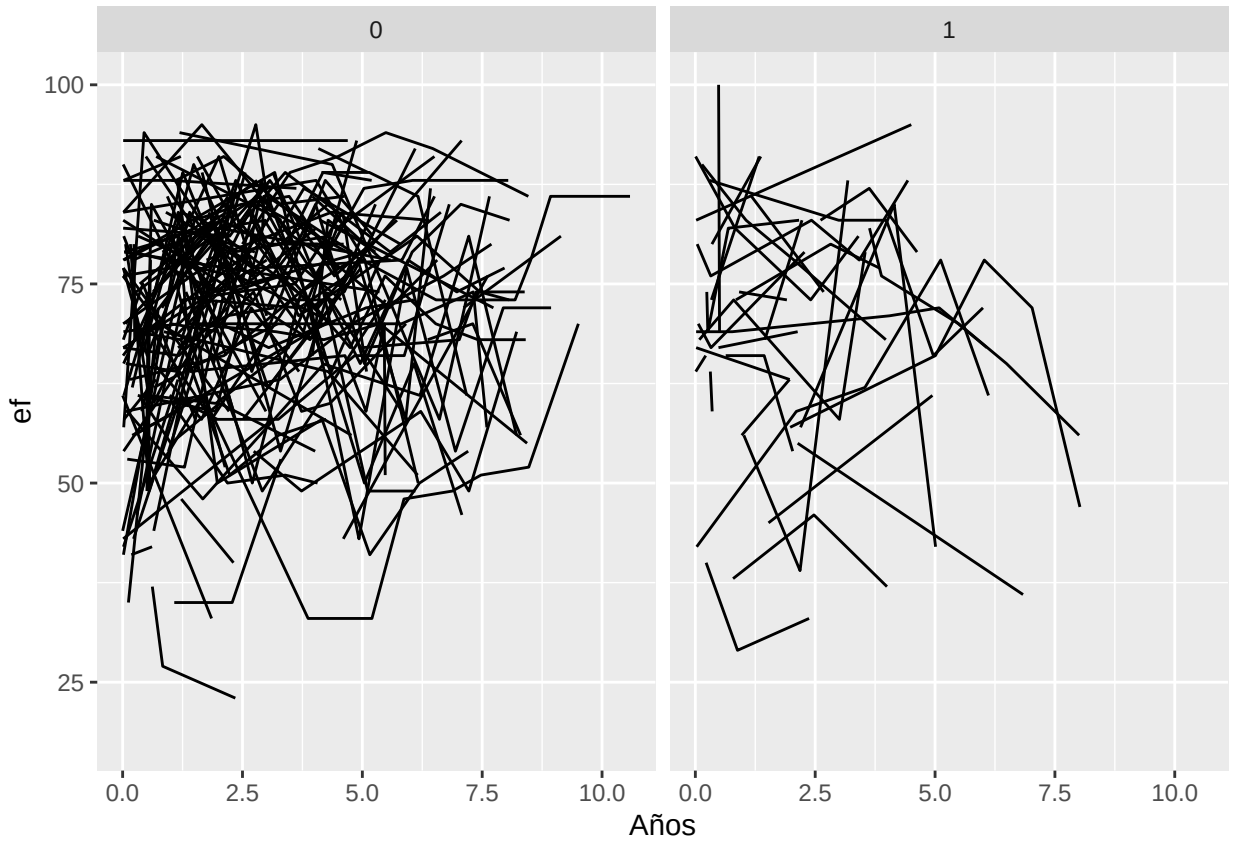
```
g1 <- ggplot(heart.valve, aes(x = time, y = grad, group = num)) +
  geom_line() + xlab("Años") + facet_wrap(heart.valve$status)

print(g1)
```




```
g2 <- ggplot(heart.valve, aes(x = time, y = lvmi, group = num)) +
  geom_line() + xlab("Años") + facet_wrap(heart.valve$status)

print(g2)
```




```
g3 <- ggplot(heart.valve, aes(x = time, y = ef, group = num)) +
  geom_line() + xlab("Años") + facet_wrap(heart.valve$status)

print(g3)
```



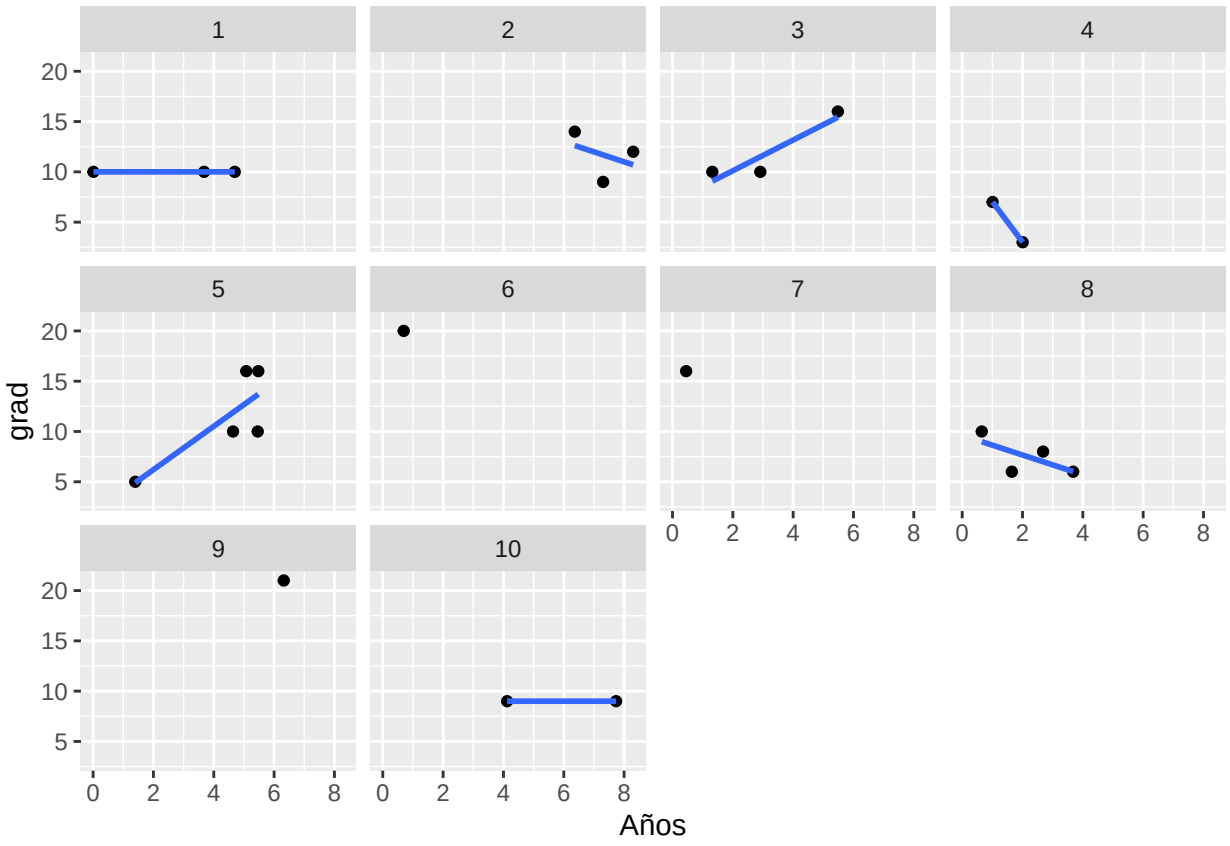
Aunque las representaciones anteriores no muestran suficiente información sobre cual de ellas podria ser una mejor variable depediente para predecir la muerte, si que podemos observar que en nuestras representaciones estratificando aquellos pacientes que han muerto podemos ver que, en cuento a la variable `grad` (gradiente valvular en la visita de seguimiento), vemos que en los casos de seguiento se mantiene que han fallecido se mantienen a lo largo de los años en niveles bajos y la variable `lvmi` (índice de masa ventricular izquierda estandarizado en la visita de seguimiento) parece presentar como valores superiores que en la mayoria de los casos que no han fallecido. En cambio, la variable `ef` (fracción de eyección en la visita de seguimiento) parece presentar el mismo comportamiento en ambos casos.

Aunque esto solo no dé una intuición de como deberian de comportarse los datos, no debemos de perder de vista que solo es una mirada general y que en ningún caso sera suficiente para determinar si incluimos o no una de las anteriores variables dependientes dentro de nuestro modelo.Así pues, podriamos plantear el hecho de trabajar de forma independiente con los tres modelos hasta llegar a una solución final donde contrastar los resultados entre ellos y decidir que variable dependiente és mejor para la predicción de muerte.

Así pues, para estas dos variables que gráficamente presenta mejores diferenciación entre los casos de muerte y no, visualicemos los 10 primeros individuos para ver si tenemos que usar un modelo mixto con intercept o pendiente aleatoria.

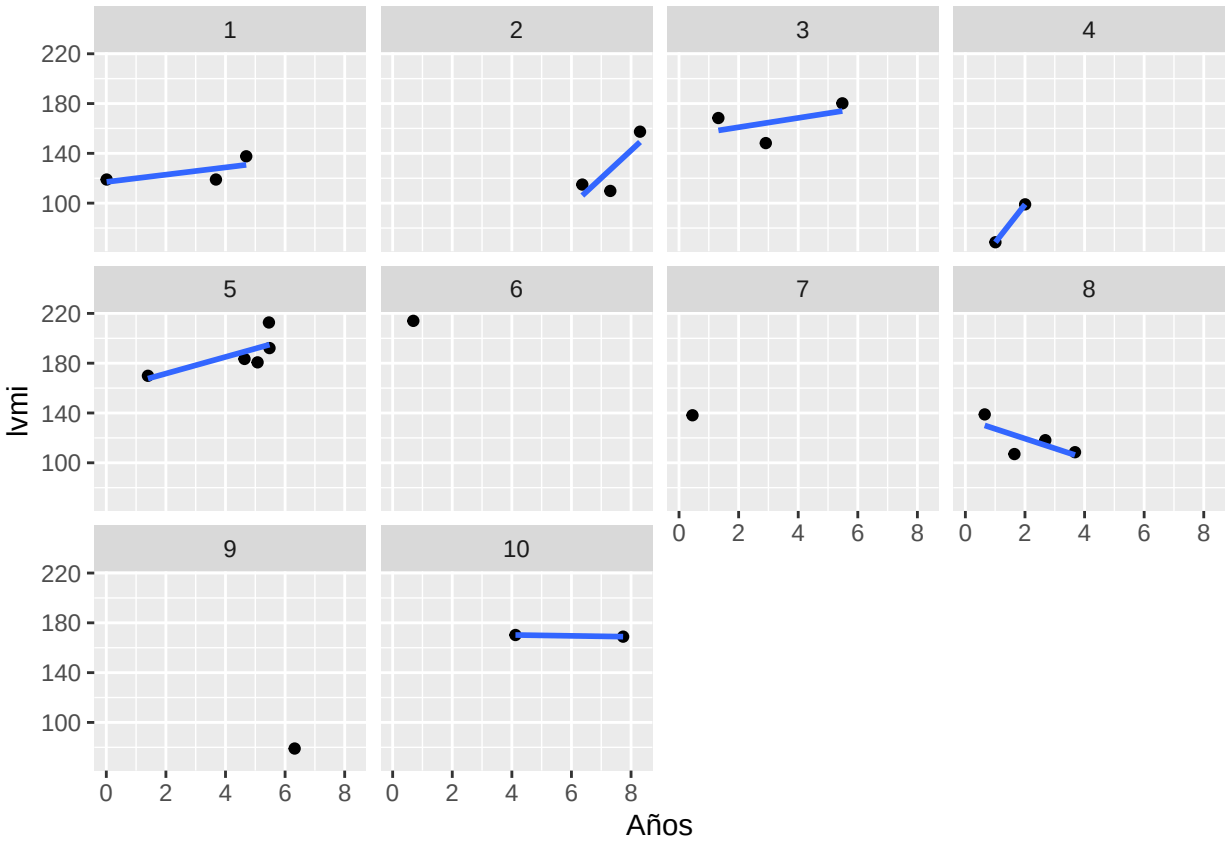
```
heart10 <- heart.valve %>% filter(num %in% 1:10)

ggplot(heart10, aes(x = time, y = grad)) +
  geom_point() +  stat_smooth(method = "lm", se = FALSE) +
  xlab("Años") + facet_wrap(~num)
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



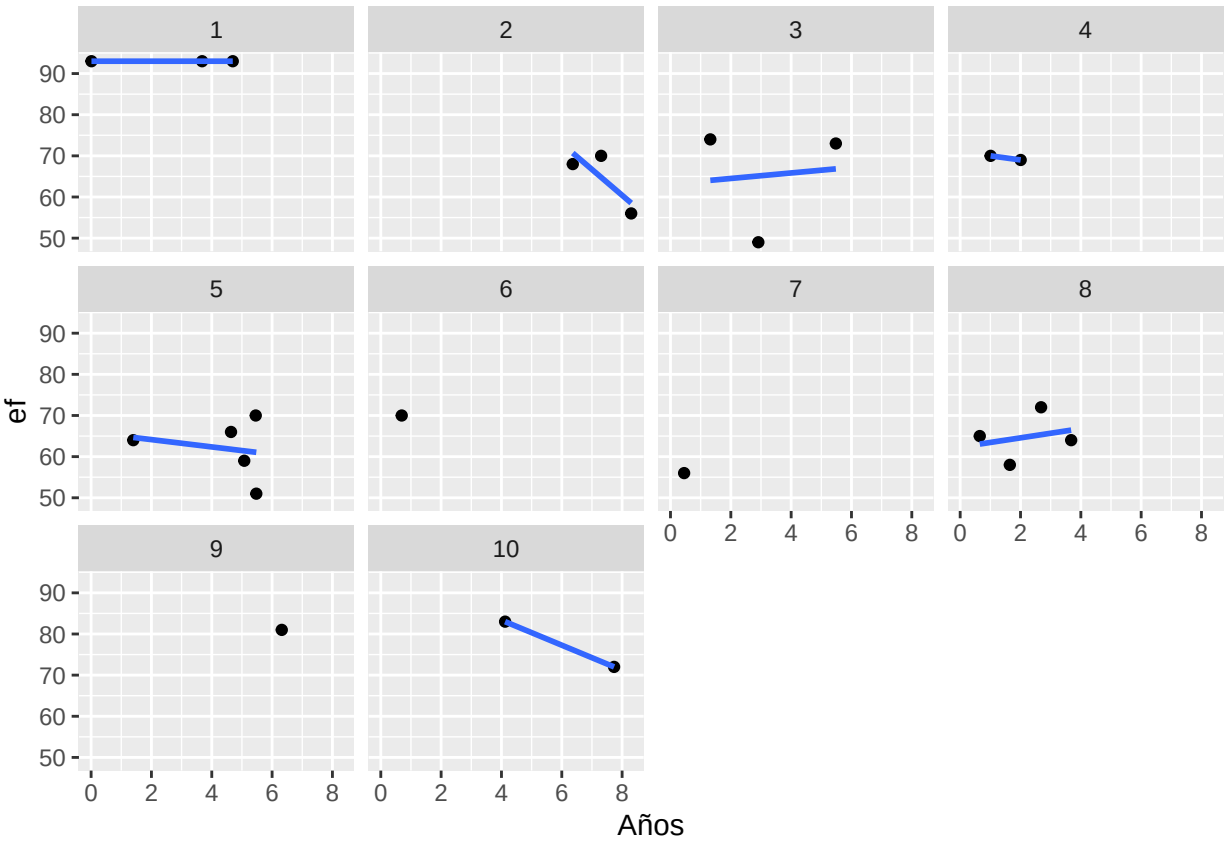

```
ggplot(heart10, aes(x = time, y = lvmi)) +
  geom_point() +  stat_smooth(method = "lm", se = FALSE) +
  xlab("Años") + facet_wrap(~num)
```

```
## `geom_smooth()` using formula = 'y ~ x'
```




```
ggplot(heart10, aes(x = time, y = ef)) +
  geom_point() +  stat_smooth(method = "lm", se = FALSE) +
  xlab("Años") + facet_wrap(~num)
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



Ahora vamos a especificar y ajustar los modelos para cada uno de nuestros outcomes. El modelo lineal de efectos mixtos para el gradiente valvular en la visita de seguimientoincluye:

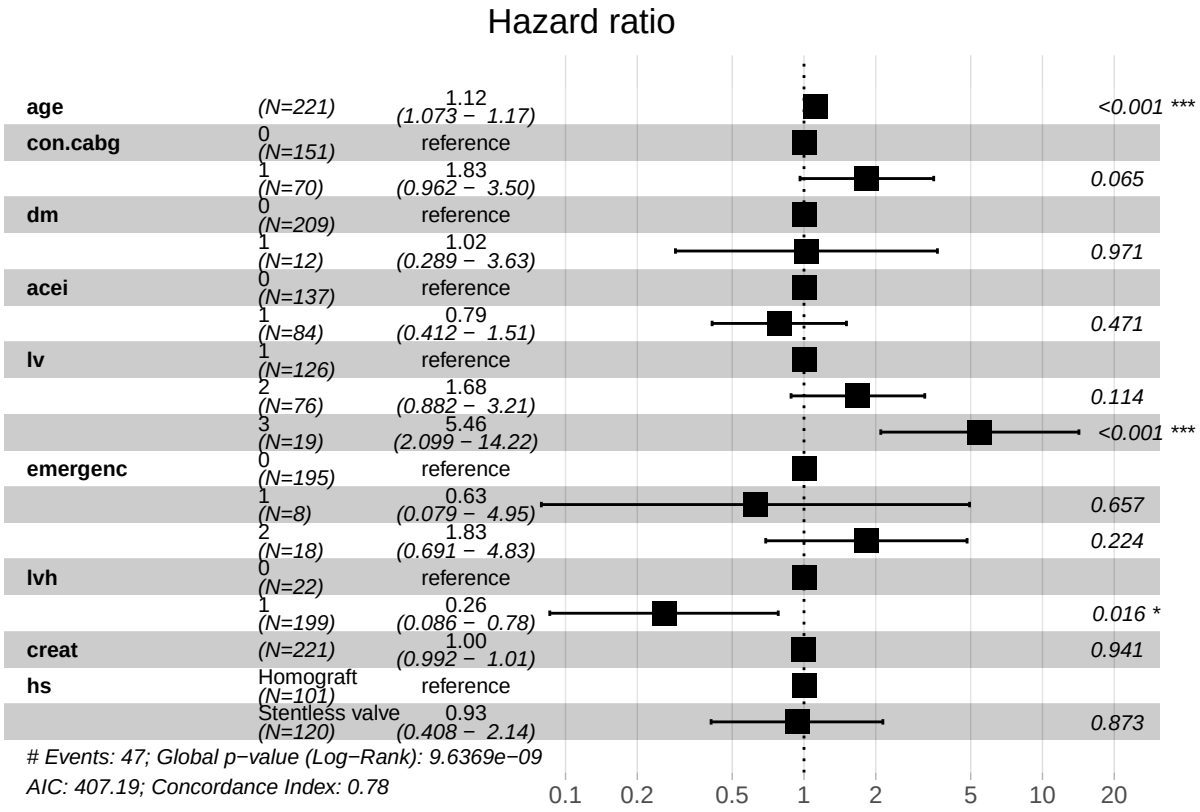
- Parte de efectos fijos: efecto principal del tiempo
- Matriz de diseño de efectos aleatorios: el intercept y un término de tiempo, ya que vemos en la gráfica anterior que ambos son aleatorios, es decir hay intercepts y pendientes distintas para cada individuo.

Y parece que este mismo modelo resulta interesante para el resto de las variable que tambien se estan estudiando, aunque con `lvmi` y con `ef` la variación entre individuos es menos pronunciada, se podria considerar que esta varición también se puede incorporar.

```
fitLME1 <- lme(grad ~ time, random = ~ time | num, data = heart.valve)
fitLME2 <- lme(lvmi ~ time, random = ~ time | num, data = heart.valve)
fitLME3 <- lme(ef ~ time, random = ~ time | num, data = heart.valve)
```

Por otra parte, el submodelo de supervivencia incluye: incluimos ciertas variable fijas del modelo que ajudaran a ajustar mejores las curvas de supervivencia. Además, a partir de los datos utilizamos B-splines que nos dará una estimación sueva de la función de riesgo basal. Así, la el estructura del modelo de supervivencia queda de la siguiente forma pero antes de realizar eso al modelo de Cox tengo que construir un data `heart.valve` agrupando por pacientes y esto funciona porque cada paciente tiene múltiples mediciones (`time`) pero sus variables de supervivencia (`fuyrs`, `age`, `status`...) no cambian entre mediciones.

```
heart.valve.num <- heart.valve[!duplicated(heart.valve$num), ]

fitSURV <- coxph(Surv(fuyrs, status) ~ age + con.cabg + dm + acei + lv + emergenc + lvh + creat + hs, da

ggforest(fitSURV, data = heart.valve.num)
```



Antes de continuar procedemos por extraer aquellas variables no significativas dentro del modelo para mejorar su ajuste, no aplicamos ningún método de extracción de variable como top-down o :

```
sum.fitSURV<- summary(fitSURV)
mat.surv <- sum.fitSURV$coefficients
aux <- which(mat.surv[,"Pr(>|z|)"]<0.05) %>% names
print(aux)
```

```
## [1] "age"  "lv3"  "lvh1"
```

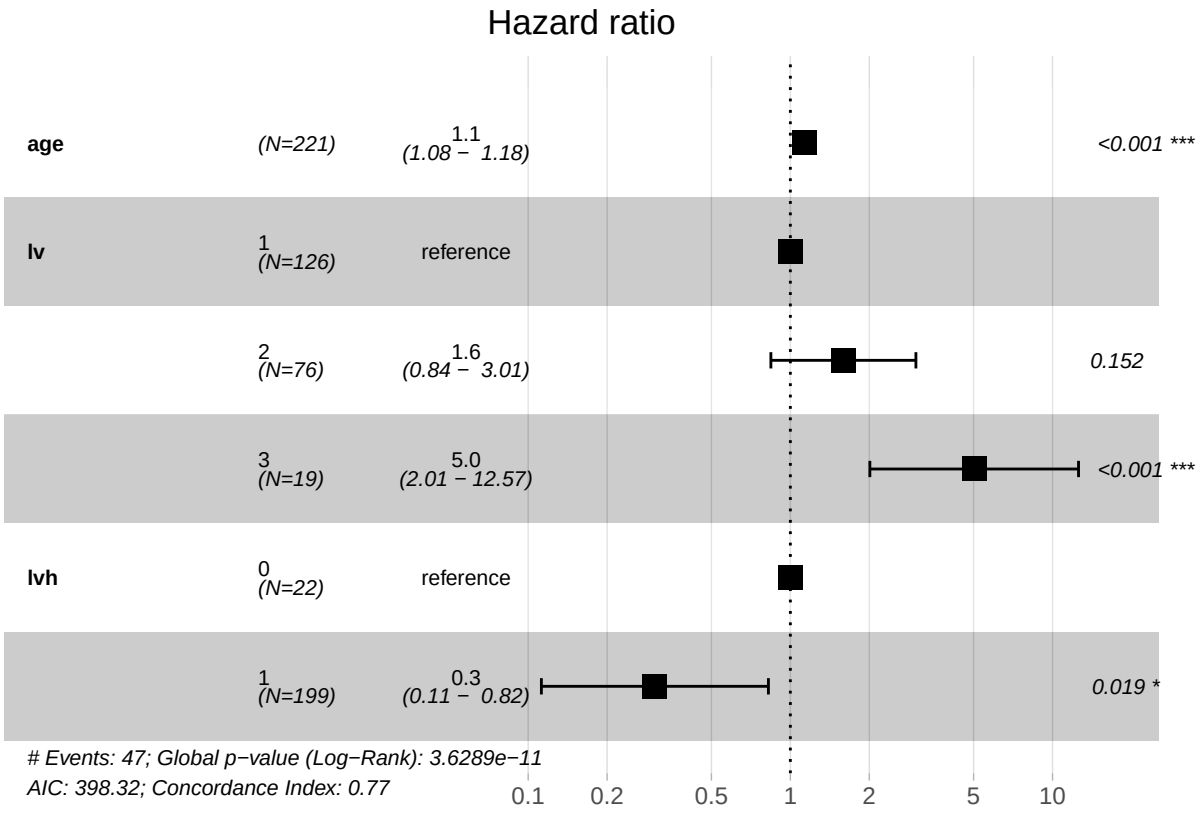
Entonces, para nuestro modelo solo nos quedamos con las variable que son significativas. En consecuencia,

```
fitSURV.nova <- coxph(Surv(fuyrs, status) ~ age + lv + lvh, data = heart.valve.num, x=TRUE)
summary(fitSURV.nova)
```

```
## Call:
## coxph(formula = Surv(fuyrs, status) ~ age + lv + lvh, data = heart.valve.num,
##     x = TRUE)
##
##   n= 221, number of events= 47
##
##          coef exp(coef) se(coef)      z Pr(>|z|)
## age   0.12419   1.13223  0.02183  5.688 1.28e-08 ***
## lv2   0.46586   1.59339  0.32492  1.434 0.151643
## lv3   1.61469   5.02631  0.46753  3.454 0.000553 ***
## lvh1 -1.18962   0.30434  0.50862 -2.339 0.019339 *
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##      exp(coef) exp(-coef) lower .95 upper .95
## age     1.1322     0.8832    1.0848    1.1817
## lv2     1.5934     0.6276    0.8428    3.0123
## lv3     5.0263     0.1990    2.0104   12.5664
## lvh1    0.3043     3.2858    0.1123    0.8247
##
## Concordance= 0.771  (se = 0.039 )
## Likelihood ratio test= 54.77  on 4 df,   p=4e-11
## Wald test            = 36.11  on 4 df,   p=3e-07
## Score (logrank) test = 38.22  on 4 df,   p=1e-07
```

```
# Dibuja el forest plot de los coeficientes
ggforest(fitSURV.nova, data = heart.valve.num)
```



A partir de los resultados del modelo de Cox donde se ha estimado la curva de supervivencia vemos que la edad augmenta el riesgo por cada año más mayor de Fracción de eyección ventricular izquierda preoperatoria

Por tanto, se crea el joint model juntando los dos modelos anteriores de la siguiente forma:

```
fitJM1 <- jointModel(fitLME1, fitSURV.nova, timeVar = "time", method = "weibull-PH-GH")
fitJM1 %>% summary
```

```
##
## Call:
## jointModel(lmeObject = fitLME1, survObject = fitSURV.nova, timeVar = "time",
##     method = "weibull-PH-GH")
```

```
##
## Data Descriptives:
## Longitudinal Process     Event Process
## Number of Observations: 629  Number of Events: 47 (21.3%)
## Number of Groups: 221
##
## Joint Model Summary:
## Longitudinal Process: Linear mixed-effects model
## Event Process: Weibull relative risk model
## Parameterization: Time-dependent
##
##    log.Lik      AIC     BIC
##  -2840.867 5707.734 5751.91
##
## Variance Components:
##               StdDev    Corr
## (Intercept)  12.6861  (Intr)
## time          2.2977 -0.9114
## Residual     15.0064
##
## Coefficients:
## Longitudinal Process
##               Value Std.Err z-value p-value
## (Intercept) 20.4903  1.3649 15.0127 <0.0001
## time        -0.8620  0.3863 -2.2317  0.0256
##
## Event Process
##                Value Std.Err z-value p-value
## (Intercept) -13.4960  1.9500 -6.9212 <0.0001
## age           0.1205  0.0213  5.6630 <0.0001
## lv2           0.4024  0.3258  1.2352  0.2168
## lv3           1.5958  0.4687  3.4045  0.0007
## lvh1         -1.1161  0.5096 -2.1903  0.0285
## Assoct        0.0287  0.0386  0.7425  0.4578
## log(shape)    0.7420  0.1396  5.3168 <0.0001
##
## Scale: 2.1002
##
## Integration:
## method: Gauss-Hermite
## quadrature points: 15
##
## Optimization:
## Convergence: 0
```

Vemos que si hacemos el jointModel con el modelo longitudinal donde se precide la variable `grad` presenta un valor negativo de la variable longitudinal al largo del tiempo queriendo decir que la variable longitudinal `grad` tiende a disminuir con el tiempo. Por lo que respecta a modelo de surpervivencia de Cox construido vemos que la variable edad y lv tienden a aumentar el riesgo. En cambio lvh tiende a dismiuir el riesgo de muerte

La variable Assoct representa la asociación entre la evolución de la variable longitudinal y el riesgo de muerte (en nuestro caso és positivo lo que un mayor valor aumenta el risgo pero NO es significativa). Si exponenciamos su valor $exp(0.029) \approx 1.03$ lo que indica que un aumento de una unidad en el gradiente valvular hace aumentar el riesgo en aprox. 3%.

```r
fitJM2 <- jointModel(fitLME2, fitSURV.nova, timeVar = "time", method = "weibull-PH-GH")
summary(fitJM2)
```

```
##
## Call:
## jointModel(lmeObject = fitLME2, survObject = fitSURV.nova, timeVar = "time",
##     method = "weibull-PH-GH")
##
## Data Descriptives:
## Longitudinal Process     Event Process
## Number of Observations: 629  Number of Events: 47 (21.3%)
## Number of Groups: 221
##
## Joint Model Summary:
## Longitudinal Process: Linear mixed-effects model
## Event Process: Weibull relative risk model
## Parameterization: Time-dependent
##
##    log.Lik      AIC      BIC
##  -3562.256 7150.511 7194.687
##
## Variance Components:
##               StdDev    Corr
## (Intercept)  68.1215  (Intr)
## time          8.4317 -0.6017
## Residual     35.7675
##
## Coefficients:
## Longitudinal Process
##                Value Std.Err z-value p-value
## (Intercept) 165.5576  5.3841 30.7495 <0.0001
## time         -1.5334  1.1649 -1.3163  0.1881
##
## Event Process
##                Value Std.Err z-value p-value
## (Intercept) -13.6005  1.8442 -7.3746 <0.0001
## age           0.1211  0.0214  5.6629 <0.0001
## lv2           0.3482  0.3226  1.0792  0.2805
## lv3           1.4745  0.4745  3.1077  0.0019
## lvh1         -1.0503  0.5050 -2.0798  0.0375
## Assoct        0.0040  0.0031  1.3066  0.1913
## log(shape)    0.7081  0.1209  5.8576 <0.0001
##
## Scale: 2.0301
##
## Integration:
## method: Gauss-Hermite
## quadrature points: 15
##
## Optimization:
## Convergence: 0
```

Presenta resultados diferentes pero interpretaciones análogas al modelo conjunto anterior. En este caso,

un aumento de una unidad en la variable longitudinal `lvmi` hace aumentar el riesgo en un 0.4% ya que $exp(0.004) \approx 1.004$ Por último, el modelo conjunto con el submodelo longitudinal `ef`:

```
fitJM3 <- jointModel(fitLME3, fitSURV.nova, timeVar = "time", method = "weibull-PH-GH")
summary(fitJM3)
```

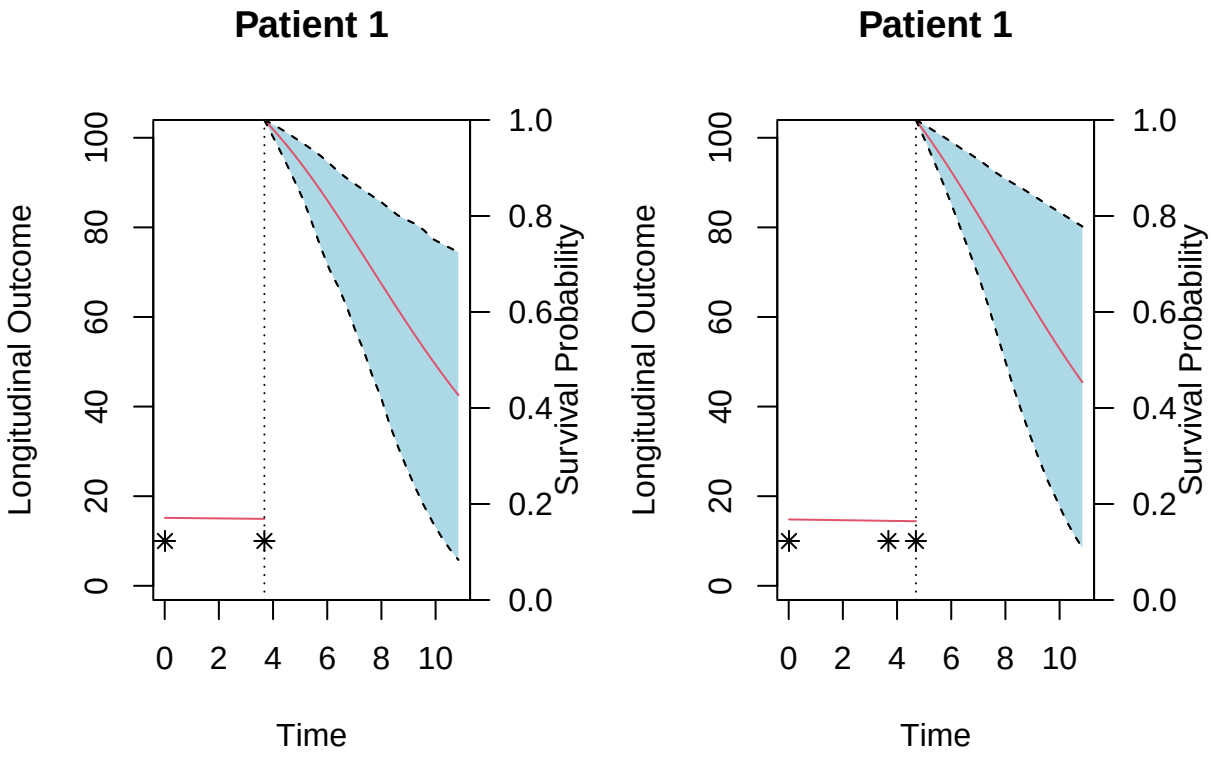
```
##
## Call:
## jointModel(lmeObject = fitLME3, survObject = fitSURV.nova, timeVar = "time",
##     method = "weibull-PH-GH")
##
## Data Descriptives:
## Longitudinal Process     Event Process
## Number of Observations: 629  Number of Events: 47 (21.3%)
## Number of Groups: 221
##
## Joint Model Summary:
## Longitudinal Process: Linear mixed-effects model
## Event Process: Weibull relative risk model
## Parameterization: Time-dependent
##
##    log.Lik      AIC      BIC
##  -2647.388 5320.777 5364.953
##
## Variance Components:
##               StdDev    Corr
## (Intercept)  11.0057  (Intr)
## time          1.2687 -0.5676
## Residual      9.8824
##
## Coefficients:
## Longitudinal Process
##               Value Std.Err z-value p-value
## (Intercept) 70.0576  1.0401 67.3595 <0.0001
## time         0.3628  0.2610  1.3901  0.1645
##
## Event Process
##                Value Std.Err z-value p-value
## (Intercept) -10.7306  2.1168 -5.0693 <0.0001
## age           0.1156  0.0208  5.5543 <0.0001
## lv2           0.3270  0.3237  1.0104  0.3123
## lv3           1.3014  0.5115  2.5445  0.0109
## lvh1         -1.0650  0.5028 -2.1181  0.0342
## Assoct       -0.0251  0.0196 -1.2813  0.2001
## log(shape)    0.7090  0.1215  5.8358 <0.0001
##
## Scale: 2.0319
##
## Integration:
## method: Gauss-Hermite
## quadrature points: 15
##
## Optimization:
## Convergence: 0
```

Al igual que en los modelos anteriores, los valores cambian pero las interpretaciones son las mismas, excepto que en este último modelo conjunto el valor de la asociación es negativo, lo que indica que un mayor valor de la vatiable longitudinal está asociado a un menor riesgo, así un aumento de una unidad en la fracción de eyección hace disminuir el riesgo en 2.5% ya que $exp(-0.0251) \approx 0.975$

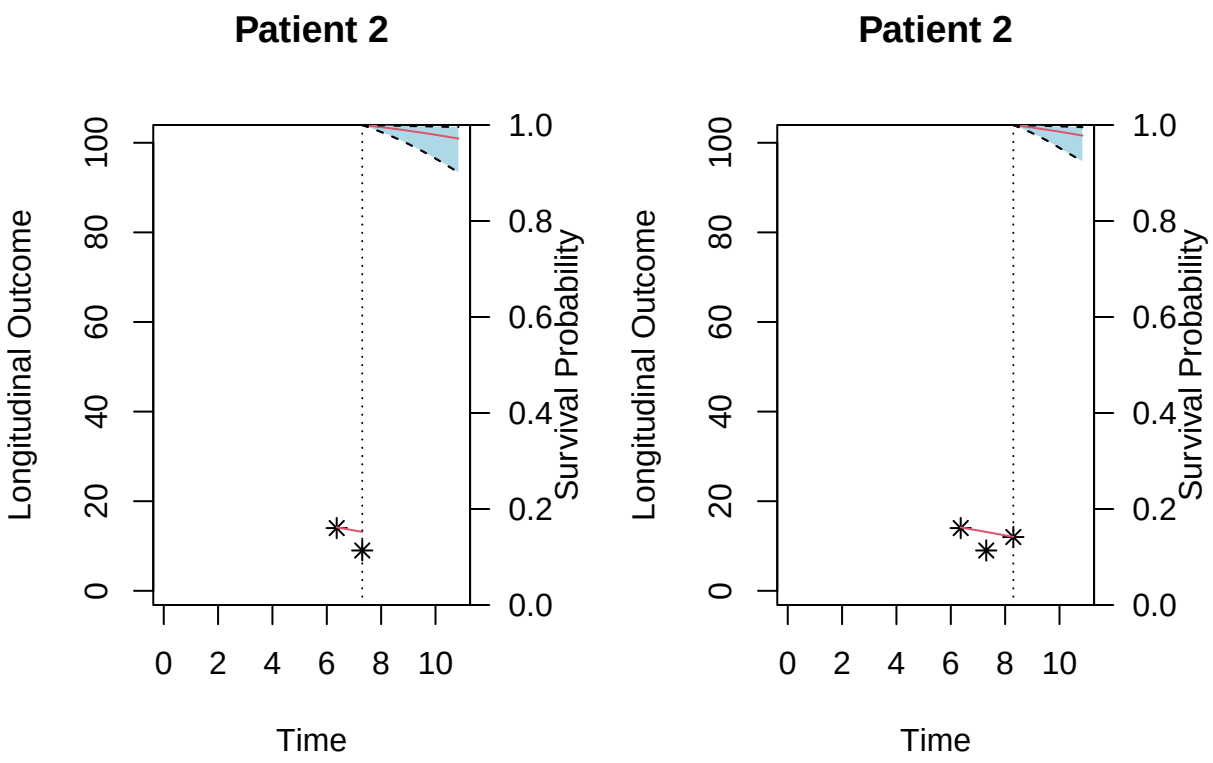
Vistos los resultados, el modelo que más nos interesa es el que incorpora el submodelo longitudinal `grad`, que presenta el mayor valores de asociación absoluto aunque en ninguno de los modelos anteriores presenten significancia.

Además, ahora vamos a graficar las probabilidades de mortalidad para los dos primeros individuos:

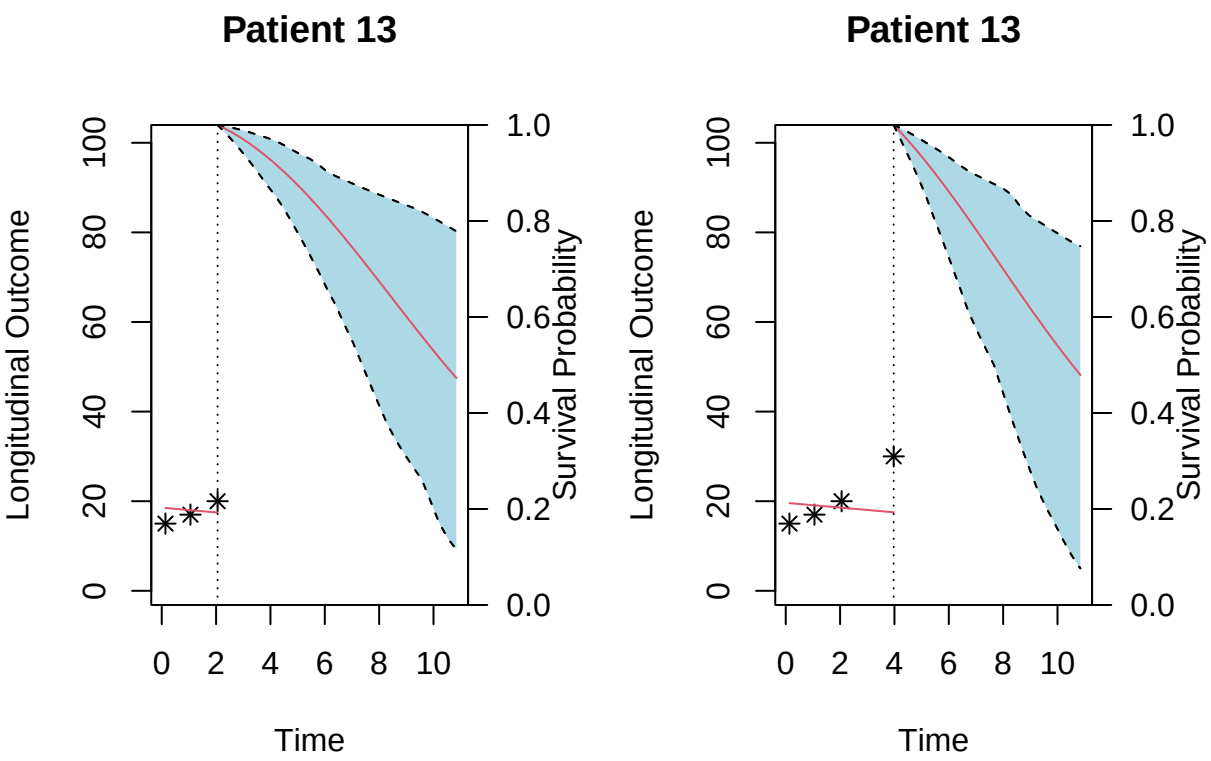
```
# Pacient 1 (no mort)
aids.id1 <- filter(heart.valve, num==1)
fit2 <- survfitJM(fitJM1, newdata = aids.id1[1:2, ], idVar = "num")
fit3 <- survfitJM(fitJM1, newdata = aids.id1[1:3, ], idVar = "num")
par(mfrow=c(1,2))
p1 <- plot(fit2, estimator="mean", include.y = TRUE, conf.int=0.95,
          fill.area=TRUE, col.area="lightblue", main="Patient 1")
p2 <- plot(fit3, estimator="mean", include.y = TRUE, conf.int=0.95,
           fill.area=TRUE, col.area="lightblue", main="Patient 1")
```




```
# Pacient 2 (no mort)
aids.id1 <- filter(heart.valve, num==2)
fit2 <- survfitJM(fitJM1, newdata = aids.id1[1:2, ], idVar = "num")
fit3 <- survfitJM(fitJM1, newdata = aids.id1[1:3, ], idVar = "num")
par(mfrow=c(1,2))
p1 <- plot(fit2, estimator="mean", include.y = TRUE, conf.int=0.95,
          fill.area=TRUE, col.area="lightblue", main="Patient 2")
p2 <- plot(fit3, estimator="mean", include.y = TRUE, conf.int=0.95,
           fill.area=TRUE, col.area="lightblue", main="Patient 2")
```




```
# Pacient 13 (mort)
aids.id1 <- filter(heart.valve, num==13)
fit2 <- survfitJM(fitJM1, newdata = aids.id1[1:3, ], idVar = "num")
fit3 <- survfitJM(fitJM1, newdata = aids.id1[1:4, ], idVar = "num")
par(mfrow=c(1,2))
p1 <- plot(fit2, estimator="mean", include.y = TRUE, conf.int=0.95,
          fill.area=TRUE, col.area="lightblue", main="Patient 13")
p2 <- plot(fit3, estimator="mean", include.y = TRUE, conf.int=0.95,
           fill.area=TRUE, col.area="lightblue", main="Patient 13")
```



Las representaciones anteriores, muestran las predicciones para los pacientes $1, 2, 13$ donde los dos primeros presentan `status` 0 y el otro `status` 1. En consecuencia, se puede ver que el longitudinal outcome en los dos primeros a lo0 largo del tiempo se mantiene constante (paciente 1) o disminuye (paciente 2) y, en cambio, se muestra un tendencia creciente para el paciente 13, un aumento de la variable longitudinal `grad` viene asociado a un mayor riesgo de mortalidad y en efecto el status que presenta es ese.