Spline question

Steve Simon

11/17/2018

Hi Dr. Simon,

I had a couple questions about the homework for module 4. In particular, question 1B, where we are calculating a cubic spline model for systolic blood pressure with four degrees of freedom.

(note - I'm using R for my work)

Please see the attachment.

1. How do we interpret the coeff and p values here – do we say that the linear effect of sysbp is significant (p=.03) but that the spline effect of sysbp is not (p=.05)?
2. However, when we plot the linear effect of sysbp against the spline effect, we see it deviates from being linear, which would suggest we want to make use of the spline effect… correct? Also, the loglik seems to indicate that the spline is the better fit…
3. Lastly, why does R report the AIC and BIC for the spline as "NA"?

Many thanks! Ethan

```
suppressMessages(suppressWarnings(library(broom)))
suppressMessages(suppressWarnings(library(dplyr)))
suppressMessages(suppressWarnings(library(ggplot2)))
suppressMessages(suppressWarnings(library(magrittr)))
suppressMessages(suppressWarnings(library(survival)))
suppressMessages(suppressWarnings(library(tidyr)))
fn <- "../../data/whas500.RData"
load(fn)
head(whas500)
```

```
##   id age gender hr sysbp diasbp      bmi cvd afb sho chf av3     miord
## 1  1  83   Male 89   152     78 25.54051  No Yes  No  No  No Recurrent
## 2  2  49   Male 84   120     60 24.02398  No  No  No  No  No     First
## 3  3  70 Female 83   147     88 22.14290  No  No  No  No  No     First
## 4  4  70   Male 65   123     76 26.63187 Yes  No  No Yes  No     First
## 5  5  70   Male 63   135     85 24.41255  No  No  No  No  No     First
## 6  6  70   Male 76    83     54 23.24236 Yes  No  No  No Yes     First
##       mitype year  admitdate    disdate      fdate los dstat lenfol fstat
## 1 Non Q-wave 1997 01/13/1997 01/18/1997 12/31/2002   5 Alive   2178 Alive
## 2     Q-wave 1997 01/19/1997 01/24/1997 12/31/2002   5 Alive   2172 Alive
## 3     Q-wave 1997 01/01/1997 01/06/1997 12/31/2002   5 Alive   2190 Alive
## 4     Q-wave 1997 02/17/1997 02/27/1997 12/11/1997  10 Alive    297  Dead
## 5     Q-wave 1997 03/01/1997 03/07/1997 12/31/2002   6 Alive   2131 Alive
## 6 Non Q-wave 1997 03/11/1997 03/12/1997 03/12/1997   1  Dead      1  Dead
##      time_yrs
## 1 5.963039014
## 2 5.946611910
## 3 5.995893224
## 4 0.813141684
## 5 5.834360027
## 6 0.002737851
```

Fit a linear effect of sysbp and a penalized spline with four degrees of freedom.

```
cox_sysbp <- coxph(
  Surv(time_yrs, fstat=="Dead")~sysbp,
    data=whas500)
cox_pspline4 <- coxph(
  Surv(time_yrs, fstat=="Dead") ~
    pspline(sysbp, df=4),
      data=whas500)
cox_sysbp
```

```
## Call:
## coxph(formula = Surv(time_yrs, fstat == "Dead") ~ sysbp, data = whas500)
##
##           coef exp(coef) se(coef)     z     p
## sysbp -0.00452   0.99549  0.00223 -2.03 0.042
##
## Likelihood ratio test=4.19  on 1 df, p=0.0406
## n= 500, number of events= 215
```

```
cox_pspline4
```

```
## Call:
## coxph(formula = Surv(time_yrs, fstat == "Dead") ~ pspline(sysbp, 
##     df = 4), data = whas500)
## 
##                               coef se(coef)      se2    Chisq   DF     p
## pspline(sysbp, df = 4), l -0.00423  0.00200  0.00200  4.47698 1.00 0.034
## pspline(sysbp, df = 4), n                              7.88091 3.04 0.050
## 
## Iterations: 5 outer, 12 Newton-Raphson
##      Theta= 0.839 
## Degrees of freedom for terms= 4 
## Likelihood ratio test=12.8  on 4.04 df, p=0.0128  n= 500
```

First things first. The coefficients in any spline model are impossible to interpret. For the linear fit, the negative coefficient tells you that as sysbp increases, the hazard decreases. But the wide range of coefficients in a penalized spline are just plain confusing.

```
coef(cox_sysbp)
```

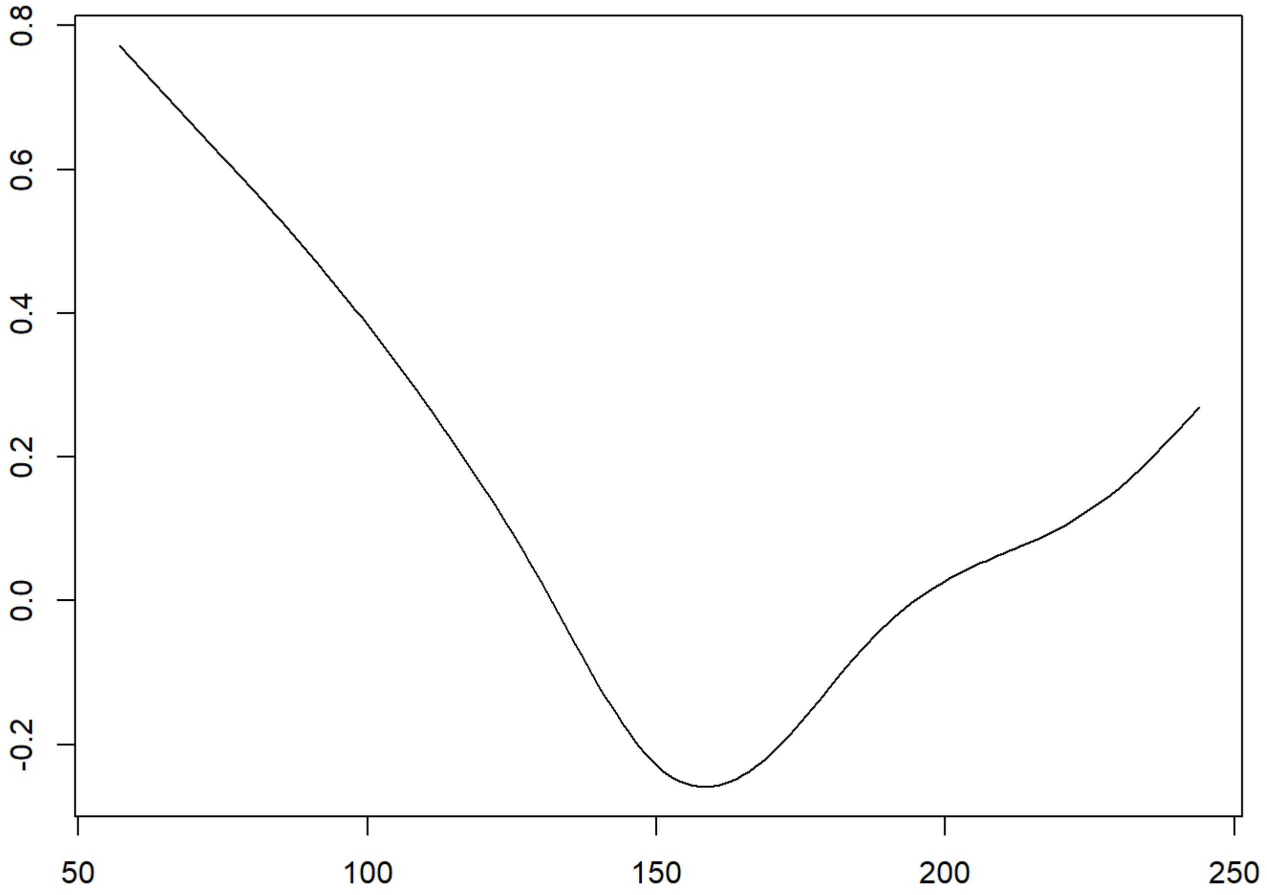
```
##        sysbp 
## -0.004520436
```

```
coef(cox_pspline4)
```

```
##  ps(sysbp)3  ps(sysbp)4  ps(sysbp)5  ps(sysbp)6  ps(sysbp)7  ps(sysbp)8 
##  -0.1700822  -0.3285876  -0.4964212  -0.6944745  -0.9288487  -1.2313653 
##  ps(sysbp)9 ps(sysbp)10 ps(sysbp)11 ps(sysbp)12 ps(sysbp)13 ps(sysbp)14 
##  -1.1886794  -0.9658867  -0.8808761  -0.8327127  -0.6752012  -0.4942463
```

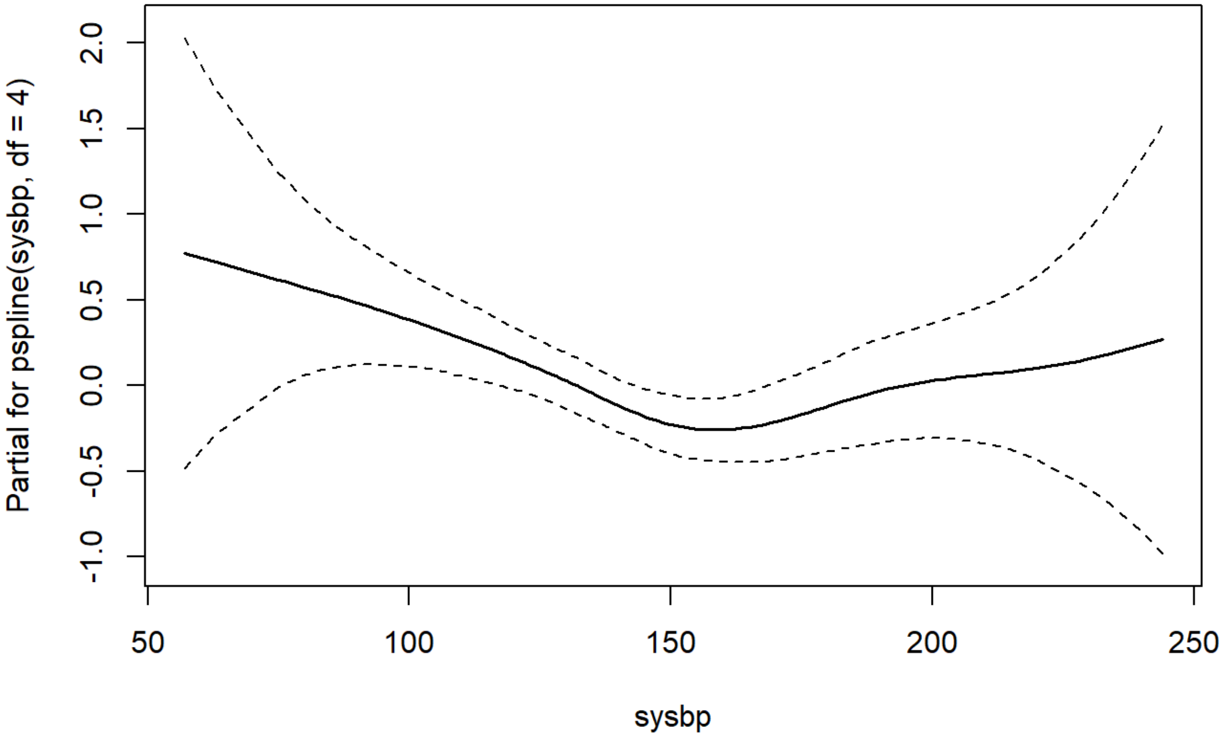
You should plot the spline function on the y-axis against the original variable on the x-axis.

```
terms_pspline4 <- predict(cox_pspline4, type="terms")
par(mar=c(2.6, 2.6, 0.6, 0.6))
o <- order(whas500$sysbp)
plot(whas500$sysbp[o], terms_pspline4[o , 1], type="l")
```



The problem with this plot is that you don’t have standard errors to judge the statistical significance of the spline. Use the termplot function to get a plot with error bounds.

```
termplot(cox_pspline4, term=1, se=TRUE, col.term=1, col.se=1)
```



The summary statistics produced by the glance function in broom are a bit confusing.

```
glance(cox_sysbp)                    %>%
  bind_rows(glance(cox_pspline4)) %>%
  mutate(lab=c(
    "linear (df=1)",
    "spline (df=4)"))                %>%
  select(lab, logLik, AIC, BIC)    -> compare_splines
```

```
## Warning: package 'bindrcpp' was built under R version 3.4.4
```

```
compare_splines
```

```
## # A tibble: 2 x 4
##   lab           logLik   AIC   BIC
##   <chr>          <dbl> <dbl> <dbl>
## 1 linear (df=1) -1225. 2452. 2456.
## 2 spline (df=4) -1221.   NA    NA
```

It would be better to use the anova function. First compare the spline fit to a null model.

```
anova(cox_pspline4)
```

```
## Analysis of Deviance Table
##  Cox model: response is Surv(time_yrs, fstat == "Dead")
## Terms added sequentially (first to last)
##
##                           loglik  Chisq     Df Pr(>|Chi|)
## NULL                     -1227.3
## pspline(sysbp, df = 4) -1220.9 12.779 4.0414    0.01282 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Next compare a spline model to a linear model.

```
anova(cox_sysbp, cox_pspline4)
```

```
## Analysis of Deviance Table
##  Cox model: response is  Surv(time_yrs, fstat == "Dead")
##  Model 1: ~ sysbp
##  Model 2: ~ pspline(sysbp, df = 4)
##    loglik  Chisq     Df P(>|Chi|)
## 1 -1225.2
## 2 -1220.9 8.5854 3.0414   0.03655 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

You can get the AIC for the linear model.

```
AIC(cox_sysbp)
```

```
## [1] 2452.448
```

but not for the penalized spline.

```
AIC(cox_pspline4)
```

```
## [1] NA
```

I’m not sure why this is, but I suspect it is related to the fact that a penalized spline only has an approximate degrees of freedom.

The restricted cubic splines in Frank Harrell’s rms package can provide an alternative to the penalized splines.

```
suppressMessages(suppressWarnings(library(rms)))
cox_rcs <- coxph(
  Surv(time_yrs, fstat=="Dead") ~
    rcs(sysbp, df=4),
      data=whas500)
cox_rcs
```

```
## Call:
## coxph(formula = Surv(time_yrs, fstat == "Dead") ~ rcs(sysbp,
##     df = 4), data = whas500)
##
##                                 coef exp(coef) se(coef)     z    p
## rcs(sysbp, df = 4)sysbp    -0.00641   0.99361  0.00987 -0.65 0.52
## rcs(sysbp, df = 4)sysbp'   -0.05138   0.94991  0.07116 -0.72 0.47
## rcs(sysbp, df = 4)sysbp''   0.29934   1.34897  0.38083  0.79 0.43
## rcs(sysbp, df = 4)sysbp''' -0.35639   0.70020  0.53886 -0.66 0.51
##
## Likelihood ratio test=11.9  on 4 df, p=0.0182
## n= 500, number of events= 215
```

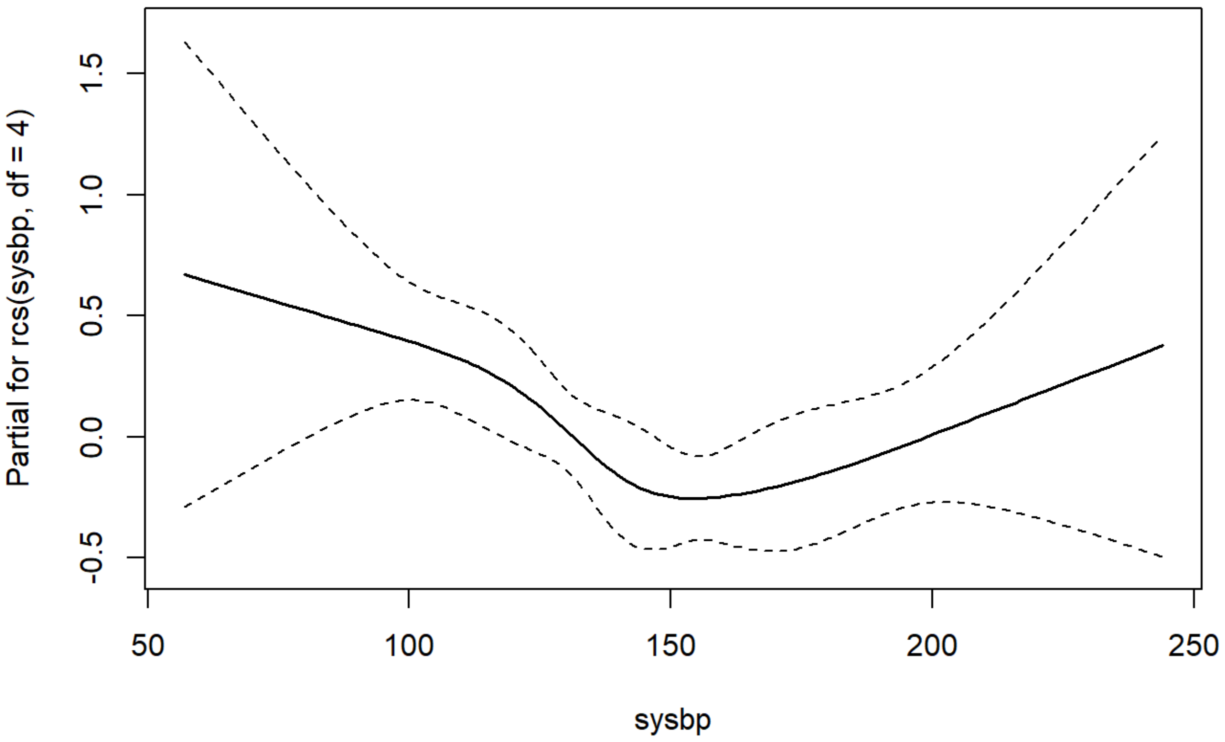
The coefficients are still rather cryptic.

```
coef(cox_rcs)
```

```
##    rcs(sysbp, df = 4)sysbp   rcs(sysbp, df = 4)sysbp'
##               -0.006407818               -0.051383876
##  rcs(sysbp, df = 4)sysbp'' rcs(sysbp, df = 4)sysbp'''
##                0.299343733               -0.356386244
```

The plot is somewhat similar to the penalized spline.

```
termplot(cox_rcs, se=TRUE, col.term=1, col.se=1)
```



You can get a formal test and the AIC for the restricted cubic spline. The formulation of the restricted cubic spline provides an exact degrees of freedom.

```
anova(cox_rcs)
```

```
## Analysis of Deviance Table
##  Cox model: response is Surv(time_yrs, fstat == "Dead")
## Terms added sequentially (first to last)
##
##                      loglik  Chisq Df Pr(>|Chi|)
## NULL               -1227.3
## rcs(sysbp, df = 4) -1221.4 11.894  4    0.01816 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
anova(cox_sysbp, cox_rcs)
```

```
## Analysis of Deviance Table
##  Cox model: response is  Surv(time_yrs, fstat == "Dead")
##  Model 1: ~ sysbp
##  Model 2: ~ rcs(sysbp, df = 4)
##    loglik  Chisq Df P(>|Chi|)
## 1 -1225.2
## 2 -1221.4 7.7004  3   0.05263 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
AIC(cox_sysbp)
```

```
## [1] 2452.448
```

```
AIC(cox_rcs)
```

```
## [1] 2450.747
```