7.8.2 Splines

By Hanzheng Li, Henry Tazewell, Xander Giarracco and Xuanxiong Zhen

2/16/2021

Importing the necessary libraries 7.8.2

```
rm(list=ls())
library(ISLR)
attach(Wage)
library(splines)
```

```
agelims=range(age)
age.grid=seq(from=agelims[1], to=agelims[2])
```

To fit the regression splines in R, we use library(splines). Section 7.4 exhibited that regression splines can be fitted through an appropriate matrix of basis functions. The bs() function generates the entire matrix of basis functions for splines using the specified set of knots. The default for bs() is to produce cubic splines. Below, wage has been fitted to age using a regression spline.

```
#First Block
fit=lm(wage~bs(age,knots=c(25,40,60)),data=Wage)
pred=predict(fit,newdata=list(age=age.grid),se=T)
plot(age,wage,col="gray")
lines(age.grid,pred$fit,lwd=2)
lines(age.grid,pred$fit+2*pred$se,lty="dashed")
lines(age.grid,pred$fit-2*pred$se,lty="dashed")

#Second Block
dim(bs(age,knots=c(25,40,60)))
```

```
## [1] 3000    6
```

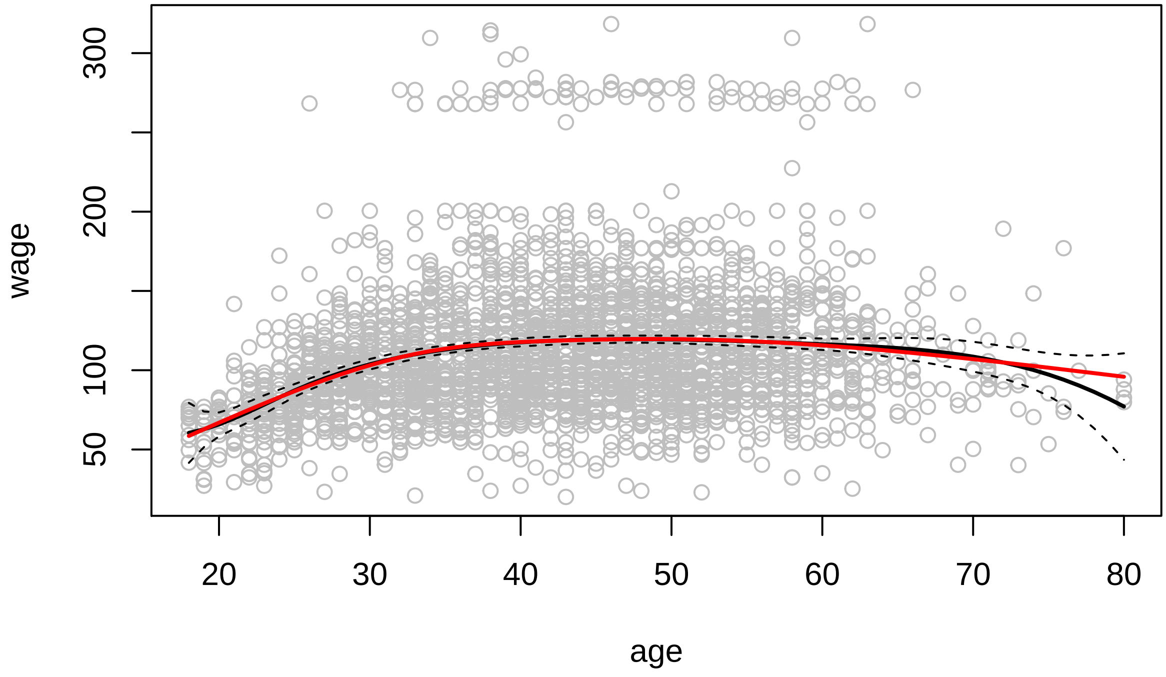
```
dim(bs(age,df=6))
```

```
## [1] 3000    6
```

```
attr(bs(age,df=6),"knots")
```

```
##   25%   50%   75%
## 33.75 42.00 51.00
```

```
#Third Block
fit2=lm(wage~ns(age,df=4),data=Wage)
pred2=predict(fit2,newdata=list(age=age.grid),se=T)
lines(age.grid, pred2$fit,col="red",lwd=2)
```



In the code above, we have placed knots at the ages 25, 40 and 60. This stands to reason because these are pivotal income years in a person's life. At from 25 to 40 years of age an individual will likely be experiencing more rapid career growth, from 40 to 60 years old there will be career stagnation and after age 60 people will begin to retire. Ultimately, these three knots create a spline with six basis functions. It is important to note that a cubic spline with three knots has seven degrees of freedom; these degrees of freedom are used up by an intercept, plus six basis functions.

Referring to the second block of code, we can use df = 6 option to to produce a spline with knots at uniform quantiles of the data. R choose the position for the knots at ages 33.75 42.0 and 51.0.The function bs() also has a degree argument, so instead of using the default degree of 3 (cubic), we can fit a spline of any degree.

For the third block of code, we use the ns() function to fit a natural spline – in this particular case we use 4 degrees of freedom. Like the bs() function, we could instead specify the knots directly using the knots option.

To fit the smoothing spline we use the smooth.spline() function and produce Figure 7.8 with the following code:

```
plot(age,wage,xlim=agelims,cex=.5,col="darkgrey")
title("FIGURE 7.8: Smoothing Spline")
fit=smooth.spline(age,wage,df=16)
fit2=smooth.spline(age,wage,cv=TRUE)
```

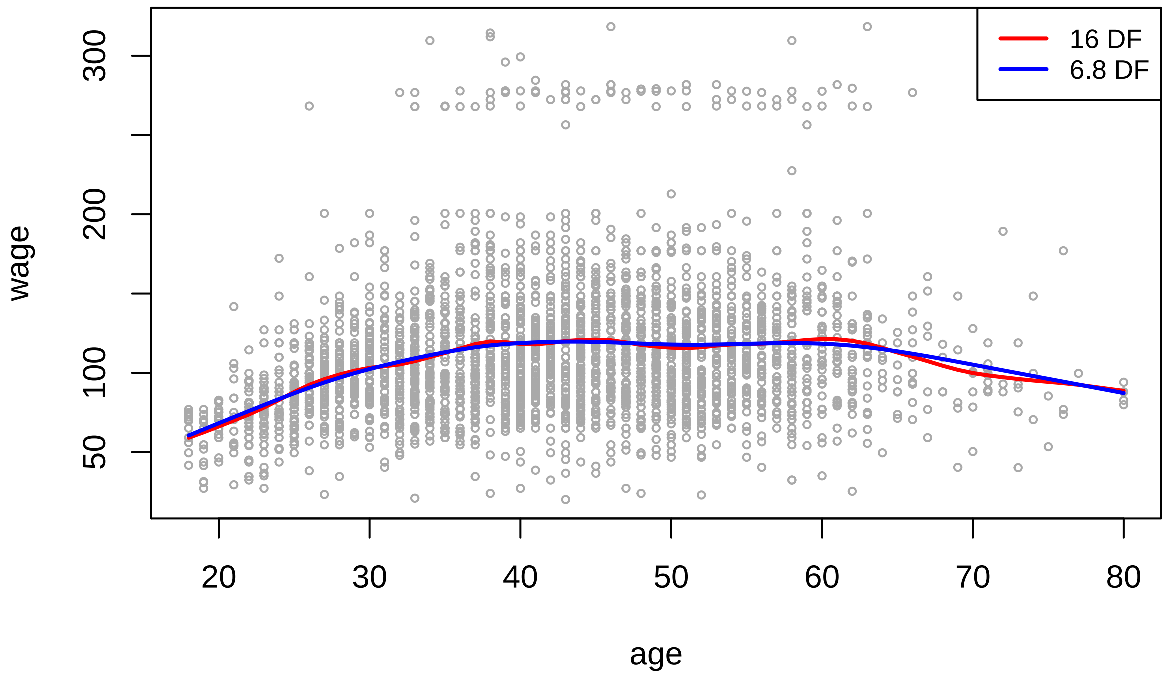
```
## Warning in smooth.spline(age, wage, cv = TRUE): cross-validation with non-unique
## 'x' values seems doubtful
```

```
fit2$df
```

```
## [1] 6.794596
```

```
lines(fit,col="red",lwd=2)
lines(fit2,col="blue",lwd=2)
legend("topright",legend=c("16 DF","6.8 DF"),col=c("red","blue"),lty=1,lwd=2,cex=.8)
```

FIGURE 7.8: Smoothing Spline

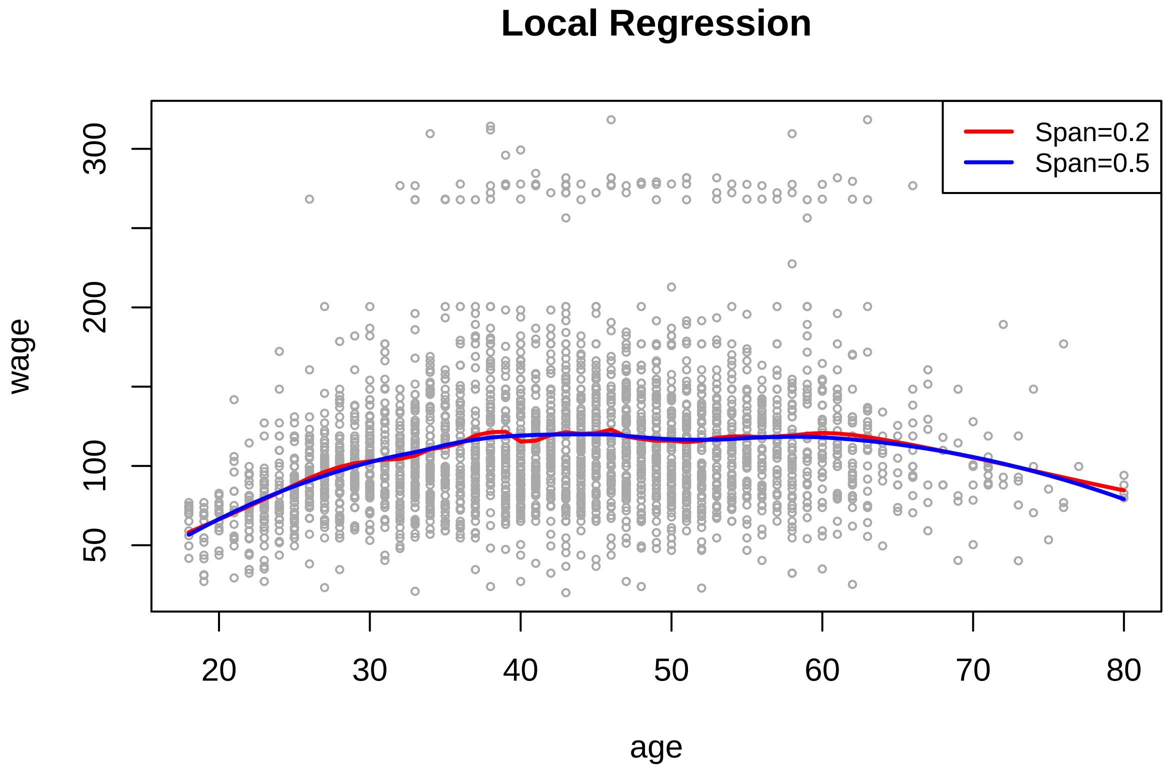


In our first call to smooth.spline(), we specified that df=16. With this input information, the function then determines which value of lambda leads to 16 degrees of freedom. The second call to smooth.spline(), we select the smoothness level through crossvalidation (shown as cv = TRUE); the result of this is a lambda that yields 6.8 degrees of freedom. In order to perform local regression, we use the loess() function. Because both models are similar in fit, we would select the smoothing spline model with the lower degrees of freedom (6.8). In other words, we chose the simpler model.

Lastly we performed the local linear regression spanning 20% to 50% of the observations. A takeaway from this is that the larger the span the smoother the fit will be. The locfit library can be used for fitting local regression models in R

```
plot(age,wage,xlim=agelims,cex=.5,col="darkgrey")
title("Local Regression")
fit=loess(wage~age,span=.2,data=Wage)
```

```
fit2=loess(wage~age,span =.5,data=Wage)
lines(age.grid,predict(fit,data.frame(age=age.grid)),col="red",lwd=2)
lines(age.grid,predict(fit2,data.frame(age=age.grid)),col="blue",lwd=2)
legend ("topright",legend=c("Span=0.2","Span=0.5"),col=c("red","blue"),lty=1,lwd=2,cex=.8)
```


Local Regression
Span=0.2
Span=0.5
wage
age