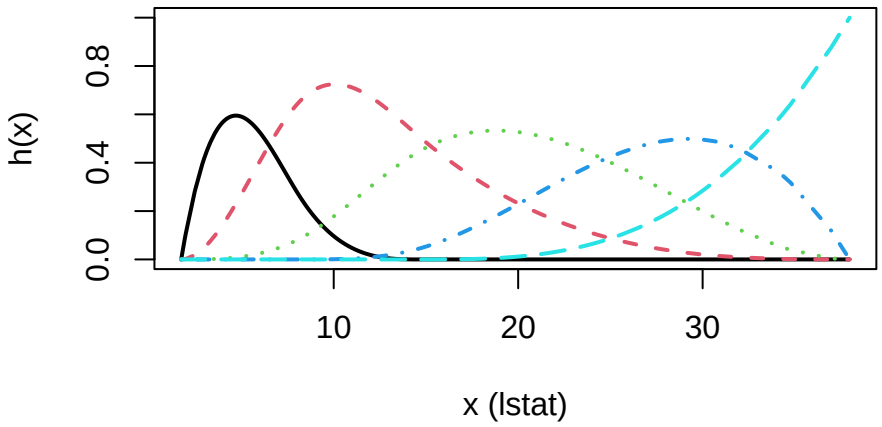
Answers to exercsies Session 5

Marjolein Fokkema

Exercise 1: Cubic and natural splines

a, b, c)

```
library("MASS")
data(Boston)
library("splines")
Boston <- Boston[order(Boston$lstat), ]
bs_x <- bs(Boston$lstat, df = 5)
matplot(Boston$lstat[order(Boston$lstat)], bs_x[order(Boston$lstat), ],
        type = "l", lwd = 2, xlab = "x (lstat)", ylab = "h(x)")
```




```
head(bs_x)
```

```
##              1           2            3 4 5
## [1,] 0.0000000 0.000000000 0.000000e+00 0 0
## [2,] 0.0828165 0.001247442 2.216045e-06 0 0
## [3,] 0.1074457 0.002148435 5.048215e-06 0 0
## [4,] 0.2824476 0.018017765 1.309221e-04 0 0
## [5,] 0.3928704 0.041199495 4.786657e-04 0 0
## [6,] 0.3952802 0.041885740 4.913730e-04 0 0
```

```
attr(bs_x, "knots")
```

```
## 33.33333% 66.66667%
##  8.316667 14.696667
```

```
attr(bs_x, "degree")
```

```
## [1] 3
```

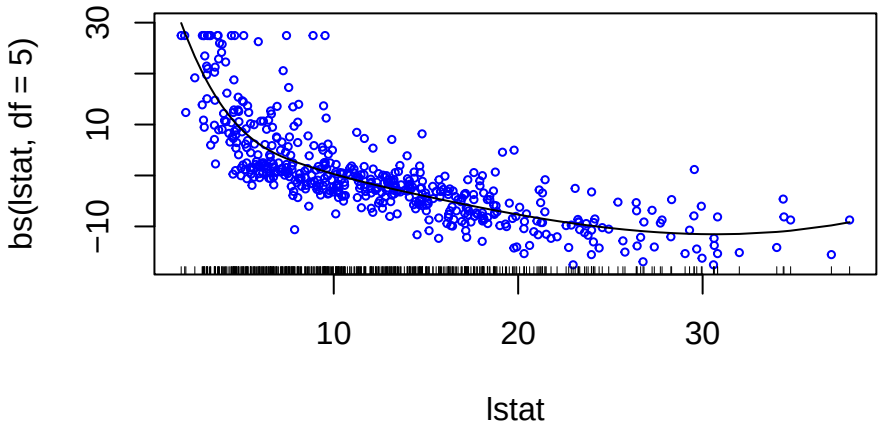
With `df = 5`, we obtain a design matrix of 5 columns. With a cubic spline, we use up 3 df for the first three expansions. Thus, with 5 df, we have 2 df 'left' to spend on the knots. Each knot introduces one additional basis function. Thus, with 5 df for a cubic spline, we can use 2 knots. Note that the knots are placed based on the univariate distribution of the predicton.

d)

```
library("gam")
mod_df5 <- gam(medv ~ bs(lstat, df = 5), data = Boston)
summary(mod_df5)
```

```
##
## Call: gam(formula = medv ~ bs(lstat, df = 5), data = Boston)
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -15.1774  -3.1790  -0.7981   2.0964  26.6755
##
## (Dispersion Parameter for gaussian family taken to be 27.109)
##
##     Null Deviance: 42716.3 on 505 degrees of freedom
## Residual Deviance: 13554.52 on 500 degrees of freedom
## AIC: 3113.663
##
## Number of Local Scoring Iterations: 2
##
## Anova for Parametric Effects
##                    Df Sum Sq Mean Sq F value    Pr(>F)
## bs(lstat, df = 5)   5  29162  5832.4  215.14 < 2.2e-16 ***
## Residuals         500  13554    27.1
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

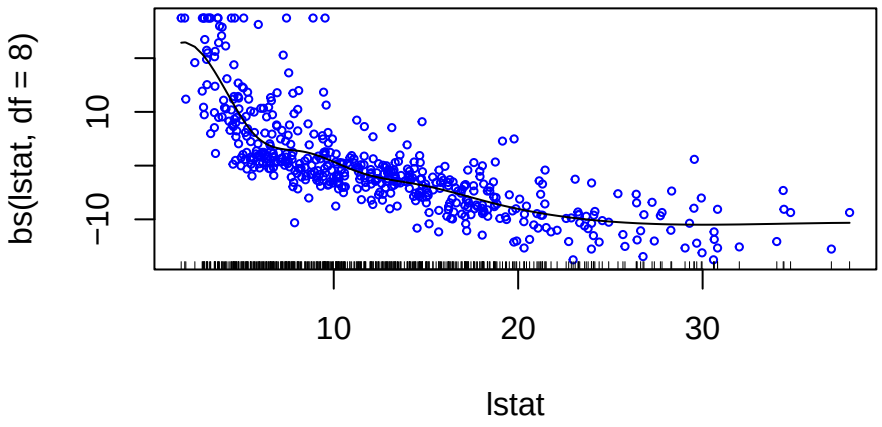
```
plot(mod_df5, residuals = TRUE, col = "blue", cex = .5)
```




```
mod_df8 <- gam(medv ~ bs(lstat, df = 8), data = Boston)
summary(mod_df8)
```

```
##
## Call: gam(formula = medv ~ bs(lstat, df = 8), data = Boston)
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -14.9627  -3.1253  -0.6612   2.0831  26.0972
##
## (Dispersion Parameter for gaussian family taken to be 26.7118)
##
##     Null Deviance: 42716.3 on 505 degrees of freedom
## Residual Deviance: 13275.77 on 497 degrees of freedom
## AIC: 3109.148
##
## Number of Local Scoring Iterations: 2
##
## Anova for Parametric Effects
##                    Df Sum Sq Mean Sq F value    Pr(>F)
## bs(lstat, df = 8)   8  29441  3680.1  137.77 < 2.2e-16 ***
## Residuals         497  13276    26.7
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
plot(mod_df8, residuals = TRUE, col = "blue", cex = .5)
```




```
BIC(mod_df5)
```

```
## [1] 3143.249
```

```
BIC(mod_df8)
```

```
## [1] 3151.414
```

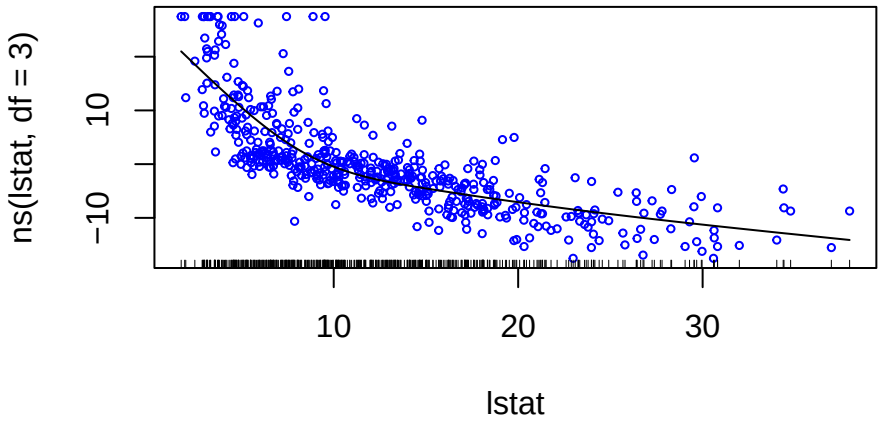
The 5 df cubic spline fits best according to BIC, the plots suggest similar: 8 df yields a slightly too wiggly function. Note that models with different (number of location of) knots are not nested, so we cannot use stattistical testing to compare the model fit.

e)

```
mod_ns3 <- gam(medv ~ ns(lstat, df = 3), data = Boston)
summary(mod_ns3)
```

```
##
## Call: gam(formula = medv ~ ns(lstat, df = 3), data = Boston)
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -13.7595  -3.3628  -0.6468   2.3062  27.2857
##
## (Dispersion Parameter for gaussian family taken to be 28.4261)
##
##     Null Deviance: 42716.3 on 505 degrees of freedom
## Residual Deviance: 14269.9 on 502 degrees of freedom
## AIC: 3135.688
##
## Number of Local Scoring Iterations: 2
##
## Anova for Parametric Effects
##                    Df Sum Sq Mean Sq F value    Pr(>F)
## ns(lstat, df = 3)   3  28446  9482.1  333.57 < 2.2e-16 ***
## Residuals         502  14270    28.4
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

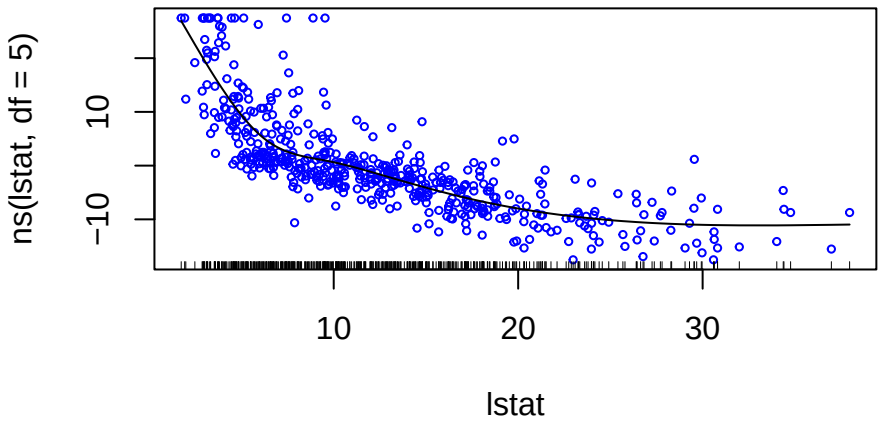
```
plot(mod_ns3, residuals = TRUE, col = "blue", cex = .5)
```




```
mod_ns5 <- gam(medv ~ ns(lstat, df = 5), data = Boston)
summary(mod_ns5)
```

```
##
## Call: gam(formula = medv ~ ns(lstat, df = 5), data = Boston)
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -13.9811  -3.0266  -0.7252   2.1416  26.5111
##
## (Dispersion Parameter for gaussian family taken to be 26.9021)
##
##     Null Deviance: 42716.3 on 505 degrees of freedom
## Residual Deviance: 13451.03 on 500 degrees of freedom
## AIC: 3109.785
##
## Number of Local Scoring Iterations: 2
##
## Anova for Parametric Effects
##                    Df Sum Sq Mean Sq F value    Pr(>F)
## ns(lstat, df = 5)   5  29265  5853.1  217.57 < 2.2e-16 ***
## Residuals         500  13451    26.9
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
plot(mod_ns5, residuals = TRUE, col = "blue", cex = .5)
```




```
BIC(mod_ns3)
```

```
## [1] 3156.82
```

```
BIC(mod_ns5)
```

```
## [1] 3139.37
```

The lowest BIC value was obtained for the natural spline with 5 df. Visually, both the 3 and 5 df natural splines seem to provide a good fit to the data.