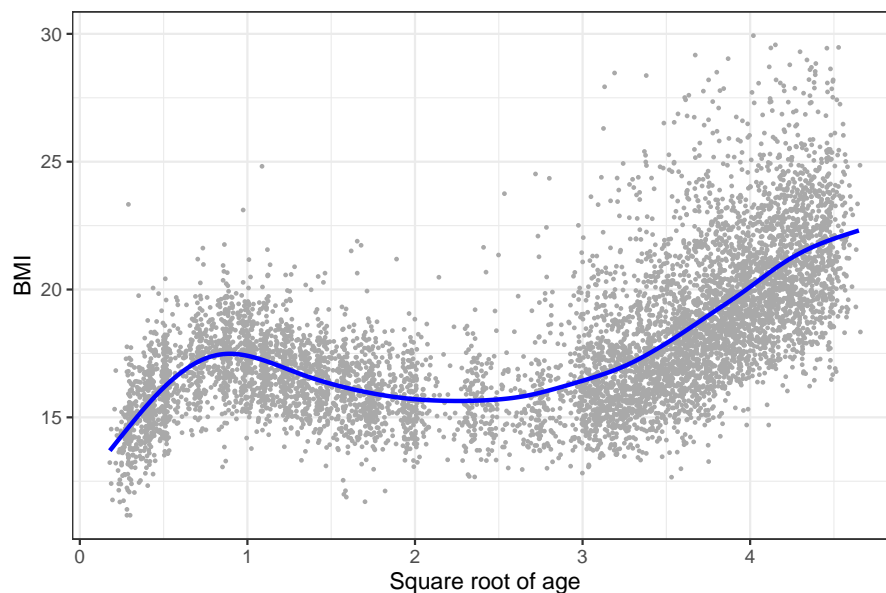


Smoothing of BMI against square root of age (Dutch boys data)



Body mass index of Dutch boys versus the square root of their age. Trend determined with the HFS algorithm. B-spline basis with 50 segments and second order penalty. R code in `f-BMI-sqrt.R`

```
# Smoothing of BMI against square root of age (Dutch boys data)
# A graph in the book 'Practical Smoothing. The Joys of P-splines'
# Paul Eilers and Brian Marx, 2019
```

```
library(ggplot2)
library(AGD)
library(JOPS)
```

```
# Get the data
data(boys7482)
Data = subset(boys7482, !is.na(age) & !is.na(bmi))
nd1 = nrow(Data)
Data = subset(Data, bmi > 10 & bmi < 30)
nd2 = nrow(Data)
cat(nd1 - nd2, "points with BMI < 10 or BMI > 30 left out\n")
x = Data$age
x = sqrt(x)
y = Data$bmi
m0 = length(y)
sel = seq(1, m0, by = 1)
x = x[sel]
y = y[sel]
```

```
# Set P-spline parameters
nseg = 50
xmin = 0
xmax = 5
B = bbase(x, xmin, xmax, nseg = nseg)
n = ncol(B)
D = diff(diag(n), diff = 2)
BtB = t(B) %*% B
Bty = t(B) %*% y
m = length(y)
```

```
# Smooth
```

```

lambda = 100
for (it in 1:10) {
  a = solve(BtB + lambda * t(D) %>% D, Bty)
  yhat = B %>% a
  G = solve(BtB + lambda * t(D) %>% D, BtB)
  ed = sum(diag(G))
  sig2 = sum((y - yhat)^2) / (m - ed - 2)
  tau2 = sum((D %>% a)^2) / ed
  lambda_new = sig2 / tau2
  dla = (lambda_new - lambda) / lambda
  lambda = lambda_new
  cat(it, ed, log10(lambda), lambda, dla, "\n")
  if (abs(dla) < 1e-05) break
}

# Compute trend on grid
xg = seq(min(x), max(x), by = 0.01)
Bg = bbase(xg, xmin, xmax, nseg = nseg)
zg = Bg %>% a

# Create data frames for ggplot
XY = data.frame(x = x, y = y)
Z = data.frame(x = xg, y = zg)

plt1 =
  ggplot(XY, aes(x = x, y = y)) +
  geom_point(color = "darkgrey", size = 0.4) +
  xlab("Square root of age") + ylab("BMI") +
  geom_line(data = Z, aes(x = x, y = y), size = 1, color = 'blue') +
  JOPS_theme()

# Make and save plot
plot(plt1)

```
