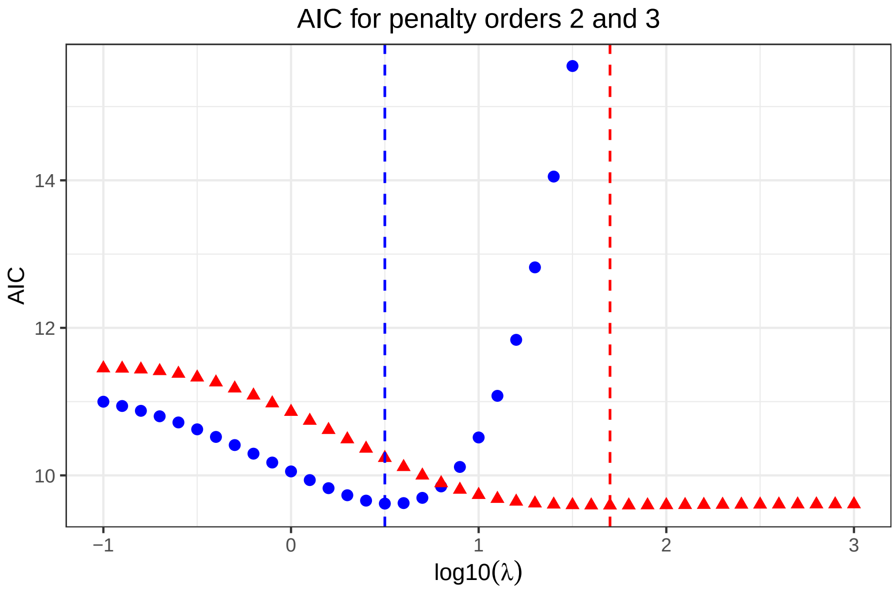
Tuning of the composite link function density estimation (Lead data)



AIC profiles for the lead in blood data for second (filled circles) and third order (triangle) penalties. The vertical lines indicate the positions of the minima. R code in f-lead-AIC.R

```
# Tuning of the composite link function density estimation (Lead data)
# A graph in the book 'Practical Smoothing. The Joys of P-splines'
# Paul Eilers and Brian Marx, 2019

library(ggplot2)
library(JOPS)

# Input data
cb <- c(1, 21, 31, 41, 51, 61)
ce <- c(20, 30, 40, 50, 60, 70)
y <- c(79, 54, 19, 1, 1, 0)
m <- length(y)
n <- ce[m]

C <- matrix(0, m, n)
for (i in 1:m) C[i, cb[i]:ce[i]] <- 1

mids = (cb + ce)/2 - 0.5
widths = ce - cb + 1
dens = y/widths/sum(y)

# Prepare B-spline matrix and penalty matrix
x <- 1:n
B <- bbase(x)
lambdas2 = 10^seq(-1, 1.5, by = 0.1)
aics2 = NULL
for (lambda in lambdas2) {
    fit2 = pclm(y, C, B, lambda = lambda, pord = 2, show = F)
    aics2 = c(aics2, fit2$aic)
}
A2 = data.frame(x = log10(lambdas2), y = aics2)
k2 = which.min(aics2)
lambdas3 = 10^seq(-1, 3, by = 0.1)
aics3 = NULL
for (lambda in lambdas3) {
    fit3 = pclm(y, C, B, lambda = lambda, pord = 3, show = F)
    aics3 = c(aics3, fit3$aic)
}
k3 = which.min(aics3)
A3 = data.frame(x = log10(lambdas3), y = aics3)

# Make plot
plt = ggplot(aes(x = x, y = y) , data = A2)  +
  geom_point(size = 2, col = 'blue') +
  geom_point(aes(x = x, y = y) , data = A3,size = 2, pch = 17, col = 'red') +
  geom_vline(xintercept = log10(lambdas2[k2]), col = 'blue', linetype = 2,  size = 0.6) +               geom_vl
  ggtitle('AIC for penalty orders 2 and 3') +
  xlab(expression(paste("log10"(lambda)))) +
  ylab('AIC') +
  JOPS_theme()

# Save plot
print(plt)
```