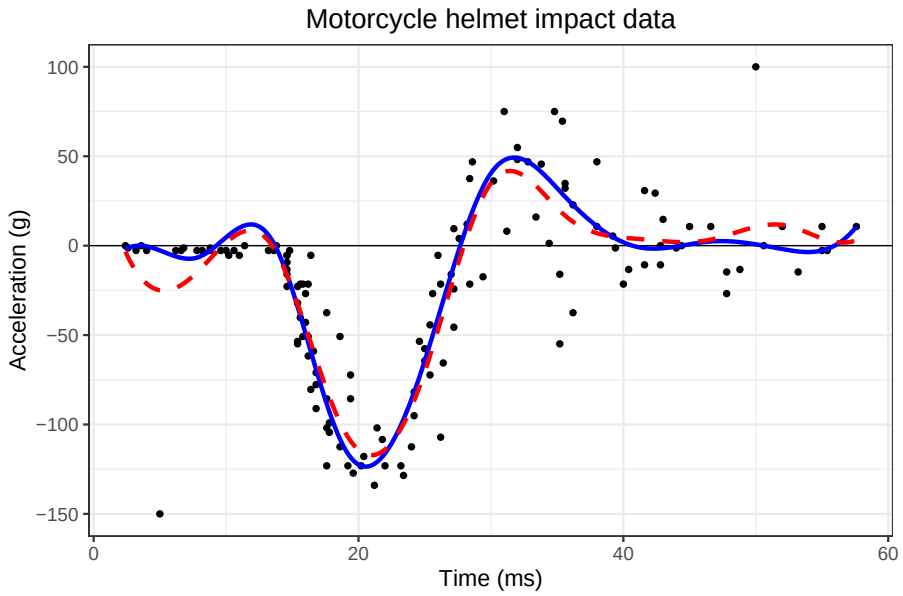
Median smoothing with P-splines (Motorcycle data)



Median smoothing of the motorcycle data with two artificial extreme outliers. Cubic P-splines, with 10 segments on the domain from 0 to 60 ms, and $\lambda = 0.01$ (solid blue line). The broken red line shows the fit of standard P-splines with the same parameter settings. R code in `f-mot-median.R`

```
# Median smoothing with P-splines (Motorcycle data)
# A graph in the book 'Practical Smoothing. The Joys of P-splines'
# Paul Eilers and Brian Marx, 2019

library(ggplot2)
library(gridExtra)
library(colorspace)
library(MASS)
library(quantreg)
library(JOPS)

# Get the data
data(mcycle)
x = mcycle$times
y = mcycle$accel

# Add outliers
x = c(x, 5, 50)
y = c(y, -150, 100)

# Compute the B-spline basis
deg = 3
xlo = min(x)
xhi = max(x)
ndx = 10
B = bbase(x, xlo, xhi, nseg = ndx, bdeg = deg)

# Basis for fit on grid
ng = 1000
xg = seq(min(x), max(x), length = ng)
Bg = bbase(xg, xlo, xhi, nseg = ndx, bdeg = deg)
n = ncol(B)

# Difference matrix
```

```
d = 2
D = diff(diag(n), diff = d)
lambda = 0.01
Bplus = rbind(B, lambda * D)
yplus = c(y, rep(0, n - d))

# Estimate the coefficients and compute fit on the grid
qfit = rq(yplus ~ Bplus - 1)
a = coefficients(qfit)
z1 = Bg %*% a

# Comput penalized least squares fit
lambda2 = lambda
a2 = solve(t(B) %*% B + lambda2 * t(D) %*% D, t(B) %*% y)
z2 = Bg %*% a2

# Create data frames for ggplot
Zf1 = data.frame(x = xg, y = z1, id = as.factor(1))
Zf2 = data.frame(x = xg, y = z2, id = as.factor(1))

# Build the graph
Data = data.frame(x, y)
plt1 = ggplot(Data, aes(x = x, y = y)) +
  geom_point(size = 1) +
  geom_hline(yintercept = 0, size = 0.3) +
  geom_line(data = Zf1, size = 1, colour = I("blue"), lty = 1) +
  geom_line(data = Zf2, size = 1, colour = I("red"), lty = 2) +
  xlab("Time (ms)") + ylab("Acceleration (g)") +
  ggtitle("Motorcycle helmet impact data") +
  ylim(c(-150, 100)) +
  JOPS_theme()

print(plt1)
```