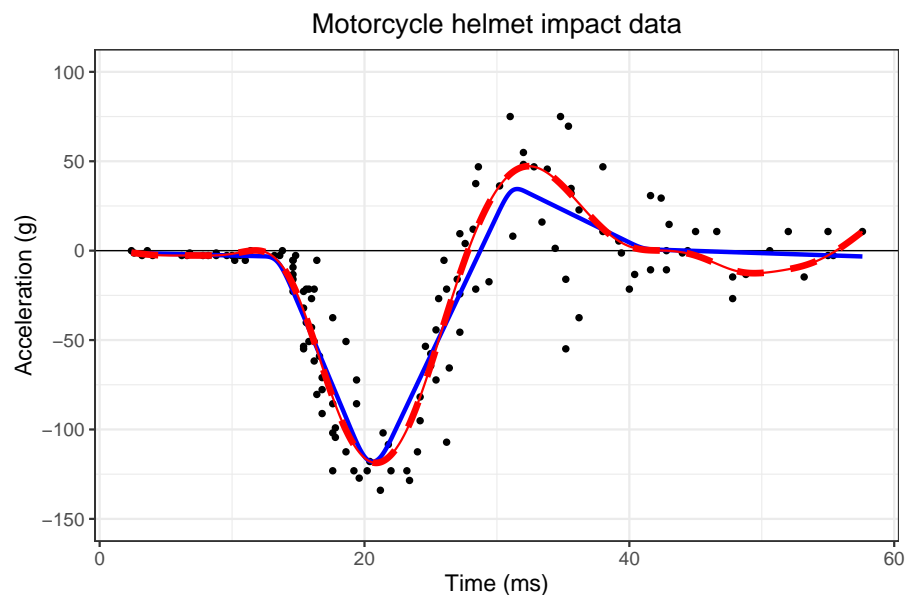


Median P-spline smoothing with L_1 and L_2 penalty (Motorcycle data)



Median smoothing of the motorcycle data using cubic P-splines, with 50 segments on the domain from 0 to 60 ms. The piecewise-linear curve (solid blue line) is obtained with absolute values in the penalty ($\lambda = 10$). The smoother curve (broken red line) is obtained with squares in the penalty ($\lambda = 1$). R code in `f-mot-median-iter.R`

```
# Median smoothing with L1 and L2 penalty (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 = 50
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)

d = 2
D = diff(diag(n), diff = d)
```

```

lambda = 10
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

# Iterative Comput penalized least squares fit
lambda2 = lambda * 0.1
beta = 1e-04 * (max(y) - min(y))
W = diag(length(y))
a2 = 0
for (it in 1:50) {
  anew = solve(t(B) %%% W %%% B + lambda2 * t(D) %%% D, t(B) %%% W %%%
    y)
  da = max(abs(a2 - anew))
  a2 = anew
  if (max(da) < 1e-06)
    break
  yhat = B %%% a2
  r = y - yhat
  w = 1/sqrt(r^2 + beta^2)
  W = diag(c(w))
  cat(it, da, max(abs(a2)), da/max(abs(a2)), "\n")
}
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 = 0.5, colour = I("red"), lty = 1) +
  geom_line(data = Zf2, size = 1.5, colour = I("red"), lty = 2) +
  xlab("Time (ms)") + ylab("Acceleration (g)") +
  ggtitle("Motorcycle helmet impact data") +
  ylim(c(-150, 100)) +
  JOPS_theme()

print(plt1)

```
