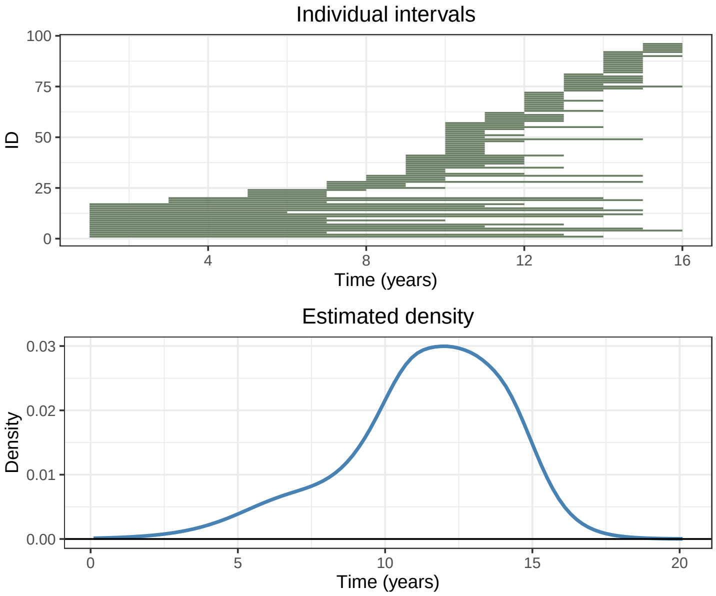
Density estimation from individual interval censored data (HIV data)



Top: individual intervals in which HIV infections occurred. Bottom: smooth estimate of the distribution, obtained with the penalized composite link model. Latent grid with 101 bins; identity matrix as basis. `R` code in `f-dens-HIV.R`

```
# Density estimation from individual interval censored data (HIV data)
# A graph in the book 'Practical Smoothing. The Joys of P-splines'
# Paul Eilers and Brian Marx, 2019

library(JOPS)
library(ggplot2)
library(ICE)
library(gridExtra)

# Get the data
data(ICHemophiliac)
la = ICHemophiliac[, 1]
fd = ICHemophiliac[, 2]
o = order(la)
la = la[o]
fd = fd[o]
m = length(la)
xmax = 20

# Create composition matreix
bw = 0.2
n = xmax / bw  + 1
C = matrix(0, m, n)
for (i in 1:m) {
  ri = ((la[i] / bw) : (fd[i] / bw)) + 1
  C[i, ri] = 1
}
y = rep(1, m)
G = matrix(0, m, n)
for (i in 1:m) G[i, ] = C[i, ] / sum(C[i, ])
x = ((1 : n) - 0.5) * bw

# Initialize
d = 2
```

```
D = diff(diag(n), diff = d)
lambda = 20000
eta = rep(1, n)
eta = eta - log(sum(exp(eta)) / n)

for (step in 1:20) {
  lambda2 = lambda / m
  P2 = lambda2 * t(D) %*% D
  for (it in 1:40) {
        gam = exp(eta)
        mu = C %*% gam
        v = t(1 / mu) %*% C / m
        r = c(v) * gam - gam + gam * eta
        U = diag(c(gam))
        enew = solve(U + P2, r)
        de = max(abs(enew - eta))
        if (de < 1e-4) break
        eta = c(enew)
  }

  # Compute new lambda
  H = solve(U + P2, U)
  ed = sum(diag(H))
  cat(lambda, ed, ’\n’)
  deta = D %*% eta
  lanew = (ed - d) /  sum(deta ˆ 2)
  lambda = lanew
}


DF1 = data.frame(x = x, y = gam)
cline = ’steelblue’
plt1 = ggplot(DF1, aes(x = x, y = y)) +
       geom_line(color = cline, size = 1) +
       geom_hline(yintercept = 0) +
       xlab(’Time (years)’) + ylab(’Density’) +
       ggtitle(’Estimated density’) +
       JOPS_theme()

A = cbind(la, fd)
DF2 = data.frame(x = as.vector(t(A)), y = rep(1:m, each = 2), grp = rep(1:m, each = 2))
plt2 = ggplot(DF2, aes(x = x, y = y, group = grp)) +
       geom_line(lwd = 0.5, color = hcl(120, 20, 50)) +
       xlab(’Time (years)’) + ylab(’ID’) +
       ggtitle(’Individual intervals’)+
       JOPS_theme()

# Show and save
grid.arrange(plt2, plt1, nrow = 2, ncol = 1)
```