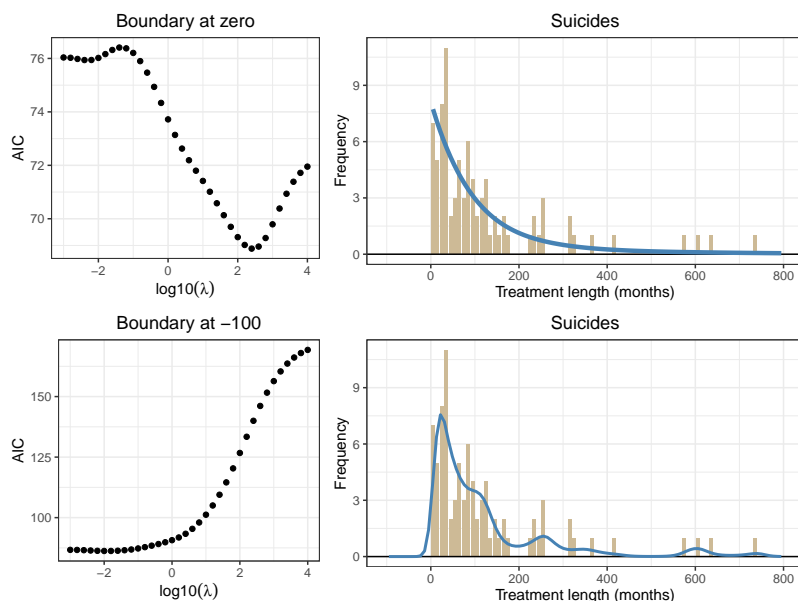


Density estimation with a sharp boundary (Suicide data)



Optimal P-spline density based on AIC for suicide data using different lower domain limits (20 cubic segments, $d = 2$). R code in `f-suicide-opt.R`

```
# Density estimation with a sharp boundary (Suicide data)
# A graph in the book 'Practical Smoothing. The Joys of P-splines'
# Paul Eilers and Brian Marx, 2019

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

# Get the data
# Note: Data is also in library(bde), but is scaled to unit interval
v = JOPS::Suicide

# Process the data in histograms
bw = 10
hst = hist(v, breaks = seq(0, 800, by = bw), plot = F)
x = hst$mids
y = hst$counts
Data = data.frame(x = x, y = y)

# Optimize AIC, for P-spline density estimation using Poisson counts
lambdas = 10^seq(-3, 4, by = 0.2)
aics = NULL
for (lambda in lambdas) {
  fit = psPoisson(x, y, nseg = 20, pord = 2, lambda = lambda, show = F)
  aics = c(aics, fit$aic)
}
ka = which.min(aics)
lambda = lambdas[ka]

# Final fit with optimal tuning
fit = psPoisson(x, y, nseg = 20, pord = 2, lambda = lambda, show = F)

# Data frames for gridded output
F1 = data.frame(x = log10(lambdas), y = aics)
Fit = data.frame(x = fit$xgrid, y = fit$mugrid)
```

```

# Plots using ggplot
plt1 = ggplot(aes(x = x, y = y), data = F1) +
  geom_point() + ggtitle(paste("Boundary at zero")) +
  xlab(expression(log10(lambda))) + ylab('AIC') +
  JOPS_theme()

plt2 = ggplot(aes(x = x, y = y, fill = I("wheat3")), data = Data) +
  geom_bar(stat = "identity") +
  geom_hline(yintercept = 0) +
  xlab("Treatment length (months)") + ylab("Frequency") + xlim(-100, 800) +
  ggtitle("Suicides") +
  geom_line(data = Fit, col = "steelblue", size = 1.5) +
  JOPS_theme()

bw = 10
hst = hist(v, breaks = seq(-100, 800, by = bw), plot = F)
x = hst$mids
y = hst$counts
Data = data.frame(x = x, y = y)

#lambdas = 10 ^ seq(-4, 2, by = 0.1)
aics = NULL
for (lambda in lambdas) {
  fit = psPoisson(x, y, nseg = 20, pord = 2, lambda = lambda, show = F)
  aics = c(aics, fit$aic)
}
ka = which.min(aics)
lambda = lambdas[ka]

fit = psPoisson(x, y, nseg = 20, pord = 2, lambda = lambda, show = F)
F1 = data.frame(x = log10(lambdas), y = aics)
Fit = data.frame(x = fit$xgrid, y = fit$mugrid)
plt3 = ggplot(aes(x = x, y = y), data = F1) +
  geom_point() + ggtitle(paste("Boundary at -100")) +
  xlab(expression(log10(lambda))) + ylab('AIC') +
  JOPS_theme()

plt4 = ggplot(aes(x = x, y = y, fill = I("wheat3")), data = Data) +
  geom_bar(stat = "identity") +
  geom_hline(yintercept = 0) +
  xlab("Treatment length (months)") + ylab("Frequency") + xlim(-100, 800) +
  ggtitle("Suicides") +
  geom_line(data = Fit, col = I("steelblue"), size = 1) +
  JOPS_theme()

# Make and save pdf
grid.arrange(plt1, plt2, plt3, plt4, ncol = 2, nrow = 2, widths = c(4, 6))

```
