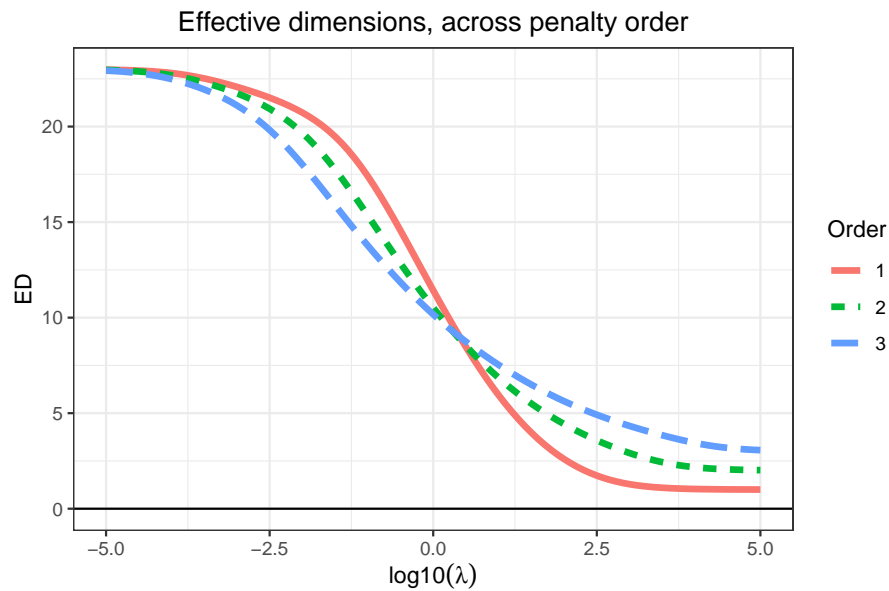


Effective dimension with increased penalization



ED vs. $\log_{10}(\lambda)$, by penalty order using 20 segments on the basis. R code in *f-Peffdim.R*

```
# Effective dimension with increased penalization
# A graph in the book 'Practical Smoothing. The Joys of P-splines'
# Paul Eilers and Brian Marx, 2019

library(MASS)
library(ggplot2)
library(JOPS)

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

mmin = min(x)
mmax = max(x)

# Double loop, for log(lambda) and order d
llambs = seq(from = -5, to = 5, by = 0.1)
lambin = 10^llambs
r = length(llambs)
cv1 = ed1 = rep(-999, r)
for (i in 1:r) {
  fitt = psNormal(x, y, mmin, mmax, 20, 3, 1, lambin[i])
  # cv1[i]=fitt$cv
  ed1[i] = fitt$eff
}

cv2 = ed2 = rep(-999, r)
for (i in 1:r) {
  fitt = psNormal(x, y, mmin, mmax, 20, 3, 2, lambin[i])
  ed2[i] = fitt$eff
}

cv3 = ed3 = rep(-999, r)
for (i in 1:r) {
  fitt = psNormal(x, y, mmin, mmax, 20, 3, 3, lambin[i])
  ed3[i] = fitt$eff
}
```

```
}

# Dataframes for ggplot
F3 = data.frame(llambs = llambs, ed3 = ed3)
F2 = data.frame(llambs = llambs, ed2 = ed2)
F1 = data.frame(llambs = llambs, ed1 = ed1)
df = data.frame(xcomb = c(F1$llambs, F2$llambs, F3$llambs), ycomb = c(F1$ed1,
  F2$ed2, F3$ed3), Order = factor(rep(c(1, 2, 3), each = length(F1$llambs))))

# Plot and save graph
qp = ggplot(df) +
  geom_line(aes(x = xcomb, y = ycomb, color=Order, linetype=Order), size=1.5) +
  labs(x=expression(paste("log10"(lambda))), y="ED")+
  ggtitle('Effective dimensions, across penalty order') +
  geom_hline(yintercept = 0) +
  JOPS_theme()

print(qp)
```
