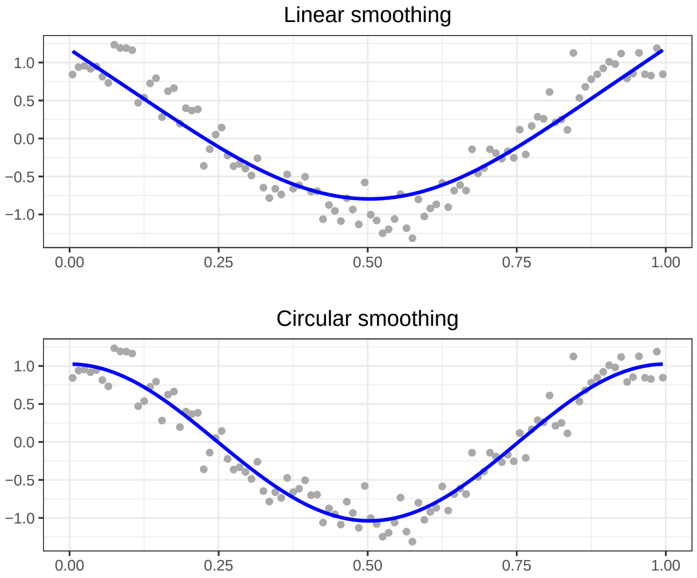
Smoothing with and without a circular B-spline basis



Smoothing with a standard (cubic) B-spline basis (top) and with a circular basis (bottom). R code in f-periodic.R

```
# Smoothing with and without a circular B-spline basis
# A graph in the book 'Practical Smoothing. The Joys of P-splines'
# Paul Eilers and Brian Marx, 2019

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

# Simulate data
m = 100
x = ((1:m) - 0.5)/m
y0 = cos(2 * pi * x)
set.seed(345)
y = y0 + rnorm(m) * 0.2

# Construct B-splines
n = 10
d = 3
B = bbase(x, 0, 1, n, 3)
nb = ncol(B)
C = cbase(x, 0, 1, n, 3)
nc = ncol(C)

# Construct penalty
D = diff(diag(nb), diff = 2)
Dc = cdiff(nc)

# P-spline solution (with and without circular penalty)
lambda = 30
ac = solve(t(C) %*% C + lambda * t(Dc) %*% Dc, t(C) %*% y)
zc = C %*% ac
a = solve(t(B) %*% B + lambda * t(D) %*% D, t(B) %*% y)
z = B %*% a

# Generate plots
Data = data.frame(x, y, z, zc)
```

```
plt1 = ggplot(Data, aes(x = x, y = y)) +
  geom_point(color = ’darkgrey’) +
  geom_line(aes(x = x, y = z), color = ’blue’, size = 1 )  +
  xlab("") + ylab("") +
  ggtitle("Linear smoothing") +
  JOPS_theme()

plt2 = ggplot(Data, aes(x = x, y = y)) +
  geom_point(color = ’darkgrey’) +
  geom_line(aes(x = x, y = zc), color = ’blue’, size = 1 )  +
  xlab("") + ylab("") +
  ggtitle("Circular smoothing") +
  JOPS_theme()

# Save graph
grid.arrange(plt1, plt2, nrow = 2, ncol = 1)
```