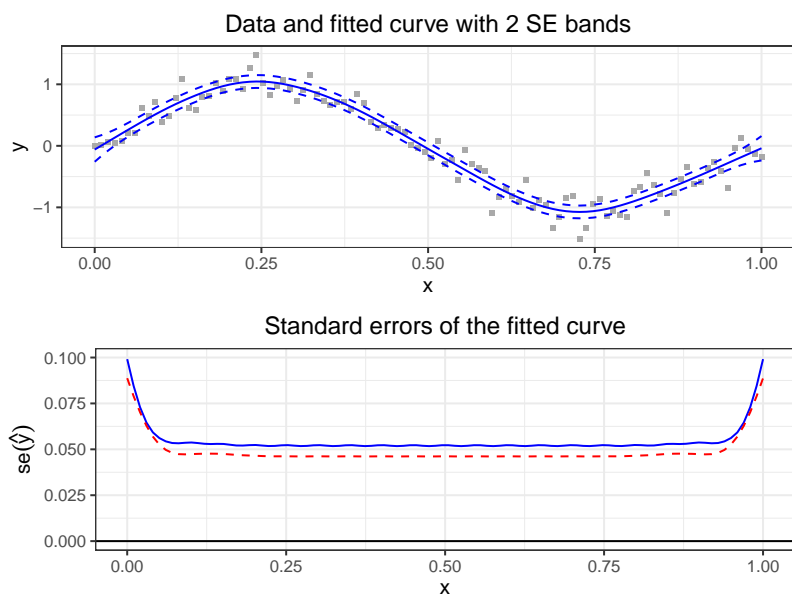


Bayesian vs. sandwich covariance estimators (Simulated data)



Standard error estimates for a P-spline fit. Top: simulated data, optimized fit (20 cubic segments, second order penalty), trend plus and minus two times the Bayesian estimate (broken lines). Bottom: Bayesian estimate (blue line) and sandwich estimate (red, broken line). R code in `f-covariance.R`

```
# Compare Bayesian and sandwich covariance estimators (Simulated data)
# A graph in the book 'Practical Smoothing. The Joys of P-splines'
# Paul Eilers and Brian Marx, 2019
```

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

```
# Simulate data
m = 100
set.seed(2019)
x = sort(runif(m))
x = seq(0, 1, length = m)
y = sin(2 * pi * x) + rnorm(m) * 0.2
B = bbase(x, 0, 1, nseg = 20)
n = ncol(B)

# Fit the model
d = 2
D = diff(diag(n), diff = d)
lambda = 10
Q = t(B) %%% B
# Find best lambda
for (it in 1:10) {
  P = lambda * t(D) %%% D
  a = solve(Q + P, t(B) %%% y)
  yhat = B %%% a
  G = solve(Q + P, Q)
  ed = sum(diag(G))
  sig2 = sum((y - yhat) ^ 2) / (m - ed - 2)
  tau2 = sum((D %%% a) ^ 2) / ed
  lanew = sig2 / tau2
  dla = lanew - lambda
  cat(it, lambda, dla, '\n')
```

```

    if (abs(dla / lambda) < 1e-4) break
    lambda = lanew
}

# Covariances of coefficients
U = solve(Q + P)
Csandw = sig2 * U %>% Q %>% U      # Sandwich
C = sig2 * U                       # Bayes

# Covariances of fitted values
V = B %>% C %>% t(B)
v = diag(V)
Vsandw = B %>% Csandw %>% t(B)
vsandw = diag(Vsandw)
c = diag(C)
csandw = diag(Csandw)

# Plot data and fits
DF1 = data.frame(x = x, y = y, yhat = yhat, se =sqrt(v), ses = sqrt(vsandw))
plt1 = ggplot(data = DF1) +
  geom_point(aes(x = x, y = y), pch = 15, col = 'darkgray', size = 1) +
  geom_line(aes(x = x, y = yhat), col = 'blue') +
  geom_line(aes(x = x, y = yhat + 2 * se), lty = 2, col = 'blue') +
  geom_line(aes(x = x, y = yhat - 2 * se), lty = 2, col = 'blue') +
  ggtitle('Data and fitted curve with 2 SE bands') +
  JOPS_theme()

# Plot SE
plt2 = ggplot(data = DF1) +
  geom_line(aes(x = x, y = se), col = 'blue') +
  geom_line(aes(x = x, y = ses), col = 'red', lty = 2) +
  ylim(0, 0.1) + ylab(expression(se(hat(y)))) +
  ggtitle('Standard errors of the fitted curve') +
  geom_hline(yintercept = 0) +
  JOPS_theme()

grid.arrange(plt1, plt2, ncol = 1, nrow = 2)

```
