Advanced registration examples

Lars Lau Raket

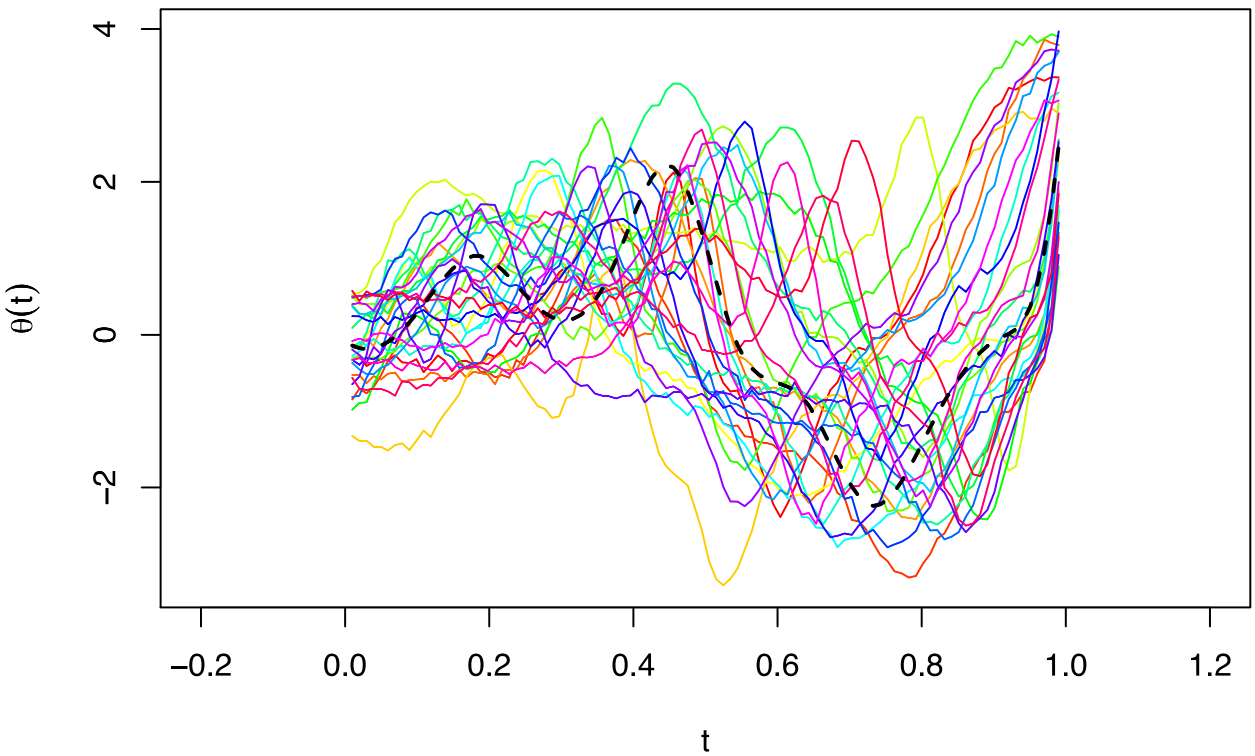
2016-01-08

In this vignette we will consider some more advanced examples of data that needs registration.

Example 1: Serial correlation, smooth warping functions and latent warp variables with unknown covariance

```
# Number of samples
n <- 30
# Number of observation points
m <- 100

# Observation points
t <- seq(0, 1, length = m + 2)[2:(m + 1)]

# Common basis function (both mean and amplitude variation)
kts <- seq(0, 1, length = 12)[2:11]
basis_fct <- make_basis_fct(kts = kts, intercept = TRUE)
df <- attr(basis_fct, 'df')

# Generate true mean weights
beta_t <- sample(-1:1, df, replace = TRUE) * rexp(df)

# Set noise standard deviation
sigma <- 0.05

# Make amplitude covariance
amp_par_t <- c(100, 0.3, 3)
amp_cov <- make_cov_fct(Matern, noise = TRUE)

# Generate warping function and random parameters
tw <- seq(0, 1, length = 4)
warp_fct <- make_warp_fct(type = 'smooth', tw = tw)

# Covariance for the latent warp variables
warp_cov <- make_cov_fct(unstr_cov, noise = FALSE, param = c(1, 1, 0))
warp_cov_true <- matrix(c(10, 8, 8, 15), 2, 2)
w_t <- replicate(n, (t(chol(warp_cov_true)) %*% rnorm(2, sd = sigma))[, 1])

# Generate data
y <- lapply(1:n, function(i) {(basis_fct(warp_fct(w_t[, i], t)) %*% beta_t
                              + sigma * t(chol(amp_cov(t, amp_par_t))) %*% rnorm(m))[, 1]})

t <- lapply(1:n, function(x) t)

# Plot observed curves
plot(0, 0, xlim = c(-0.2, 1.2), ylim = range(y), type = 'n',
```

```
     xlab = 't', ylab = expression(theta(t)))
for (i in 1:n) lines(t[[i]], y[[i]], col = rainbow(n)[i])
lines(t[[1]], basis_fct(t[[1]]) %*% beta_t , lwd = 2, lty = 2)
```



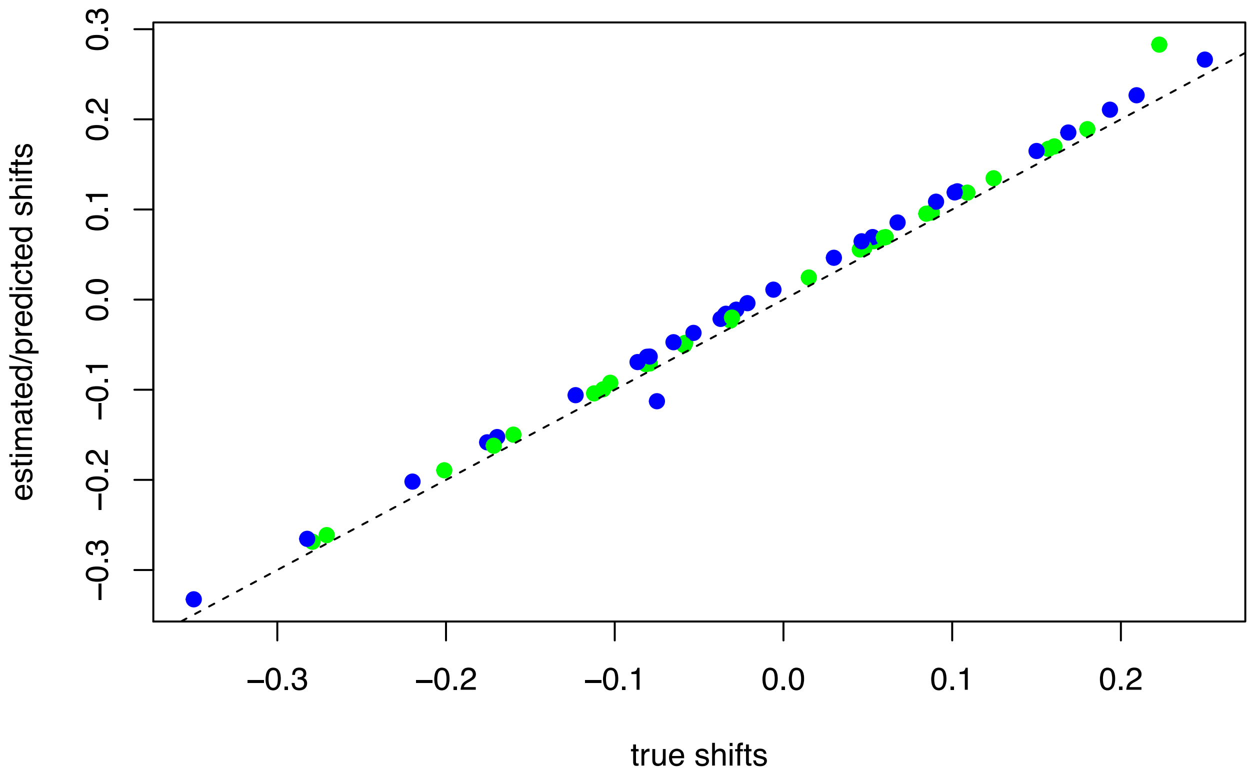
Estimate in the model

```
res <- pavpop(y, t, basis_fct, warp_fct, amp_cov = amp_cov, warp_cov = warp_cov,
              iter = c(10, 5))
```

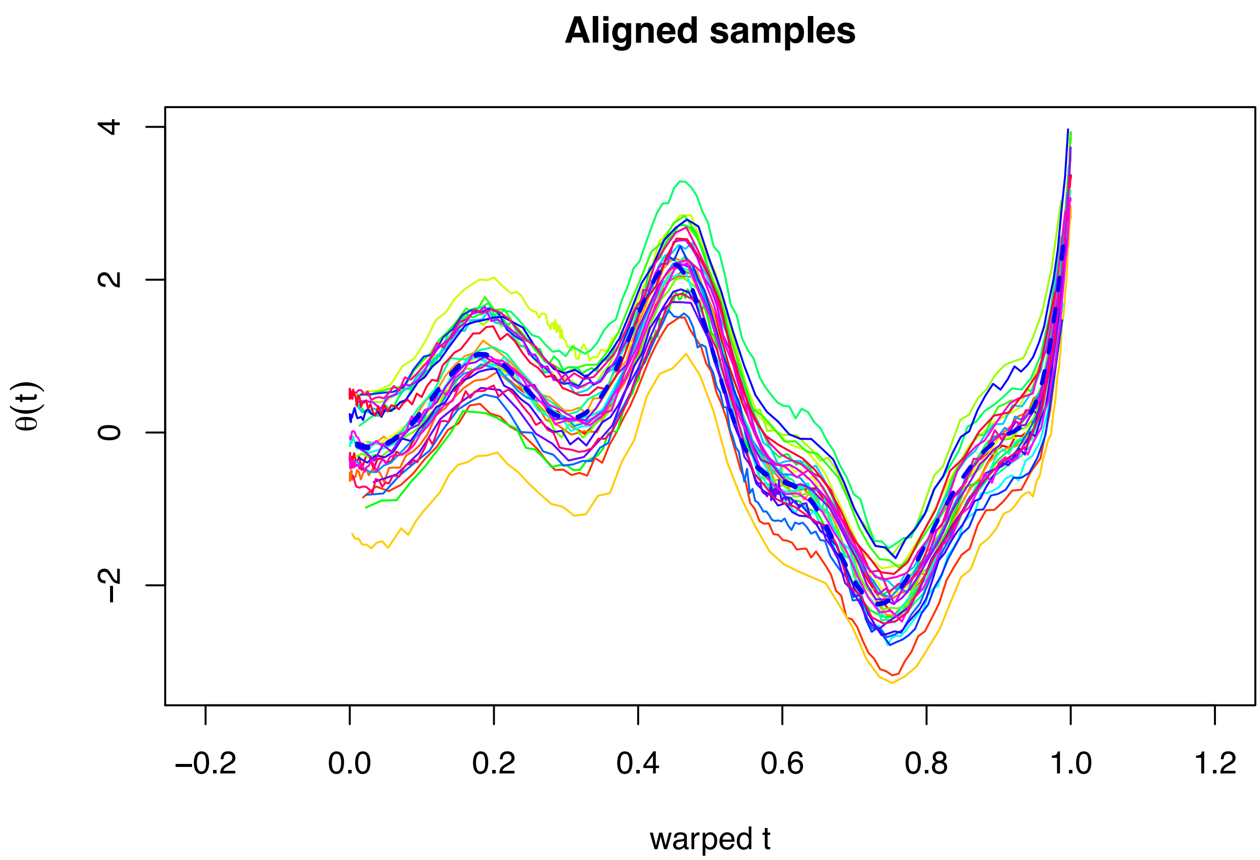
```
#> Outer    :   Inner   :   Estimates
#> 1    :   1   2   3   4   5   :    102.3591 0.3728278 0.6492638 11.94809 13.10092 1.928108
#> Linearized likelihood:    -14547.48
#> 2    :   1   2   3   4   5   :    102.4089 2.359256 0.491095 10.41457 12.03465 4.676706
#> Linearized likelihood:    -15401.09
#> 3    :   1   2   3   4   5   :    102.4547 0.4607253 2.172129 10.09659 11.77051 4.830682
#> Linearized likelihood:    -16290.78
#> 4    :   1   2   3   4   5   :    101.8509 0.3387756 2.950623 9.44868 9.951066 0.001
#> Linearized likelihood:    -16470.09
#> 5    :   1   2   3   4   5   :    101.6807 0.3247422 3.070692 8.012871 8.881977 2.466108
#> Linearized likelihood:    -16525.25
#> 6    :   1   2   3   4   5   :    101.313 0.3198467 3.108319 6.812938 8.102344 2.249765
#> Linearized likelihood:    -16536.83
#> 7    :   1   :    101.144 0.3258002 3.051452 6.868345 7.903965 2.264255
#> Linearized likelihood:    -16539.9
#> 8    :   1   2   3   4   5   .
#> Likelihood not improved, returning best likelihood estimates.
```

We can not plot the results

```r
plot(as.numeric(w_t), as.numeric(res$w), xlab = 'true shifts', ylab = 'estimated/predicted shifts',
     pch = 19, col = c('green', 'blue'), ylim = range(res$w))
abline(0, 1, lty = 2)
```




```r
plot(0, 0, xlim = c(-0.2, 1.2), ylim = range(y), type = 'n',
     xlab = 'warped t', ylab = expression(theta(t)), main = 'Aligned samples')
for (i in 1:n) lines(warp_fct(res$w[, i], t[[i]]), y[[i]], col = rainbow(n)[i])
lines(t[[1]], basis_fct(t[[1]]) %*% res$c, lwd = 2, lty = 2, col = 'blue')
lines(t[[1]], basis_fct(t[[1]]) %*% beta_t, lwd = 2, lty = 2, col = 'blue')
```




```r
# Compare noise variance

sigma^2
```

```
#> [1] 0.0025
```

```r
res$sigma^2
```

```
#> [1] 0.002522391
```

```r
# Compare amplitude variance parameters

# True parameters
c(sigma^2 * amp_par_t[1], amp_par_t[-1])
```

```
#> [1] 0.25 0.30 3.00
```

```r
# Estimated
c(res$sigma^2 * res$amp_cov_par[1], res$amp_cov_par[-1])
```

```
#>      scale      range smoothness
#>  0.2551246  0.3258002  3.0514517
```

```r
# Compare estimated variance parameters of warps

# True covariance matrix
sigma^2 * warp_cov_true
```

```
#>       [,1]   [,2]
#> [1,] 0.025 0.0200
#> [2,] 0.020 0.0375
```

```r
# Covariance matrix of true shifts
var(t(w_t))
```

```
#>             [,1]        [,2]
#> [1,] 0.017077087 0.006399135
#> [2,] 0.006399135 0.020351488
```

```r
# Estimated covariance matrix
res$sigma^2 * warp_cov(1:2, res$warp_cov_par)
```

```
#>             [,1]        [,2]
#> [1,] 0.017324650 0.005711335
#> [2,] 0.005711335 0.019936889
```

```r
# Covariance matrix for predicted warps
var(t(res$w))
```

```
#>             [,1]        [,2]
#> [1,] 0.017968112 0.005770915
#> [2,] 0.005770915 0.020647848
```