Growth curve analysis with pavpop

Lars Lau Raket

2016-01-08

Berkeley Growth Study data

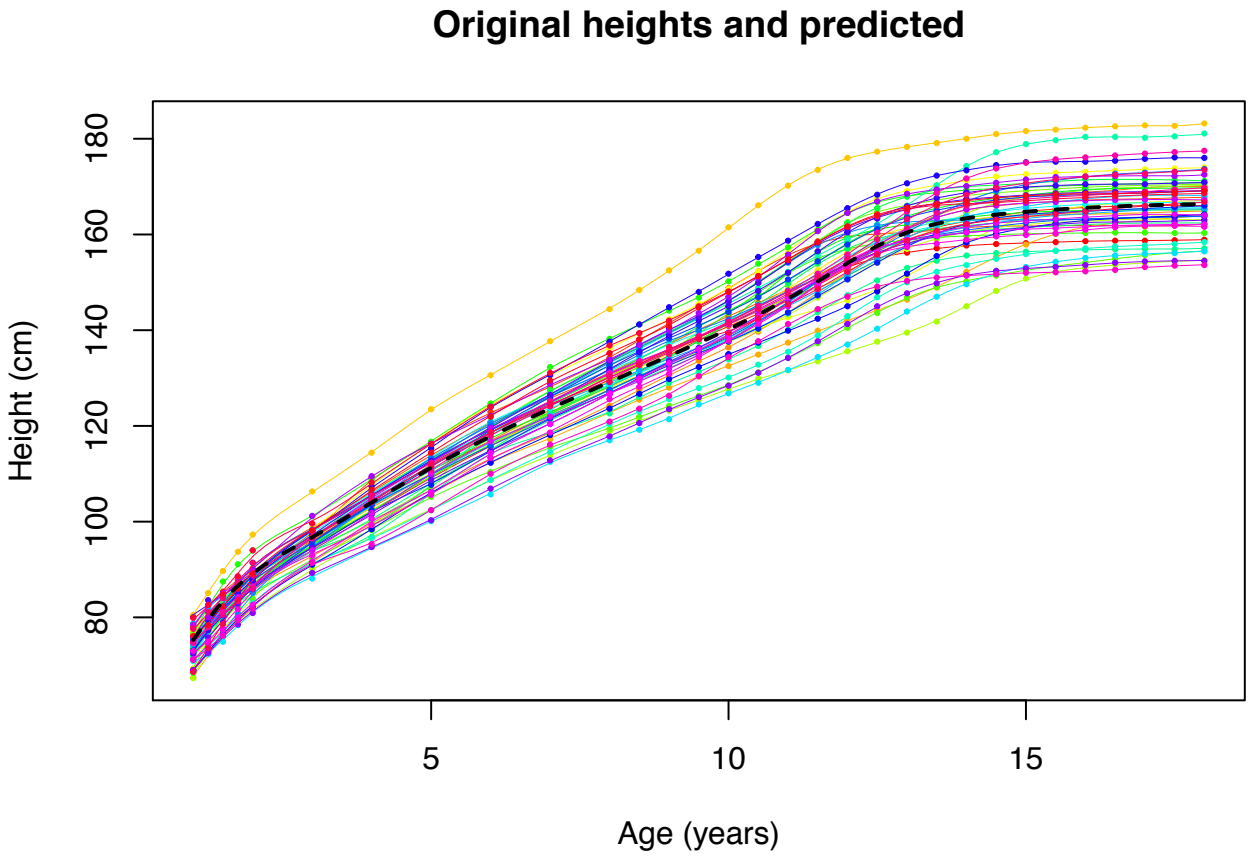
```
# Load female growth data from the Berkeley growth study
t <- fda::growth$age
y <- fda::growth$hgtf
m <- nrow(y)
n <- ncol(y)

# Specify age rage for controlling boundary points
t_range <- c(0, 20)
t <- replicate(n, t, simplify = FALSE)
y <- lapply(1:n, function (x) y[, x])
```

Matern covariance

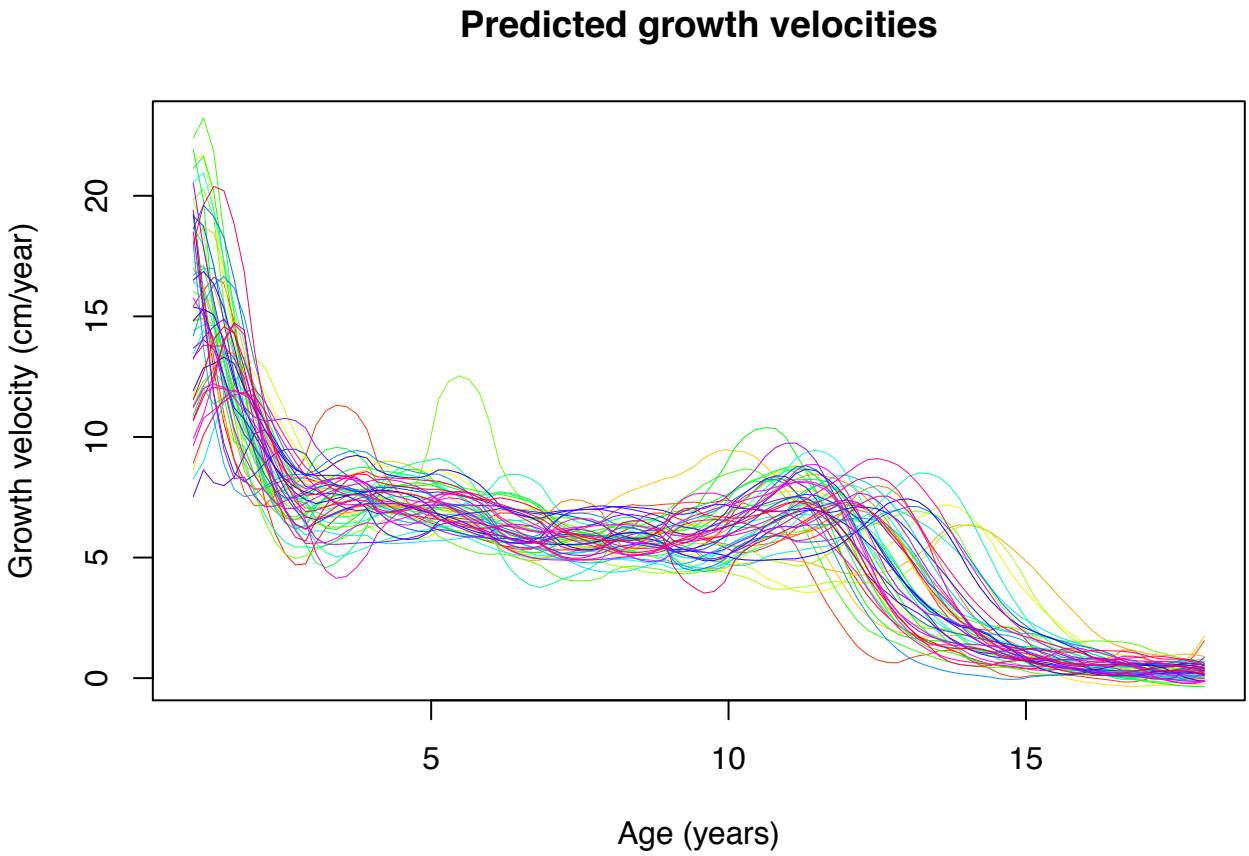
```
# Set up basis function
kts <- seq(t_range[1], t_range[2], length = 15)
basis_fct <- make_basis_fct(kts = kts, type = 'increasing', intercept = TRUE,
                     control = list(boundary = t_range))

# Set up warp function
tw <- seq(t_range[1], t_range[2], length = 6)
warp_fct <- make_warp_fct('smooth', tw, control = list(wright = 'extrapolate'))
mw <- attr(warp_fct, 'mw')

# Set up covariance functions
warp_cov_par <- c(tau = 10)
warp_cov <- make_cov_fct(Brownian, noise = FALSE, param = warp_cov_par, type = 'motion',
                         range = t_range)

amp_cov_par <- c(scale = 200, range = 10, smoothness = 3)
amp_cov <- make_cov_fct(Matern, noise = TRUE, param = amp_cov_par)


# Estimate in the model

# Bounds of parameters
# NOTE: Prediction of velocities is only meaningful
#       when the smoothness parameter is > 0.5
lower <- c(1e-2, 1e-2, 0.5001, 1e-2)
upper <- c(1000, Inf, Inf, Inf)

res <- pavpop(y, t, basis_fct, warp_fct, amp_cov, warp_cov, homeomorphisms = 'soft',
              like_optim_control = list(lower = lower, upper = upper))
```

```
#> Outer    :   Inner   :   Estimates
#> 1    :   1   2   3   4   5   :    200.3704 9.183805 0.9211515 0.7840058
#> Linearized likelihood:    -736.33
#> 2    :   1   2   3   4   5   :    200.7289 8.450124 0.9634986 0.8126704
#> Linearized likelihood:    -813.2733
#> 3    :   1   2   3   4   5   .
#> Likelihood not improved, returning best likelihood estimates.
```

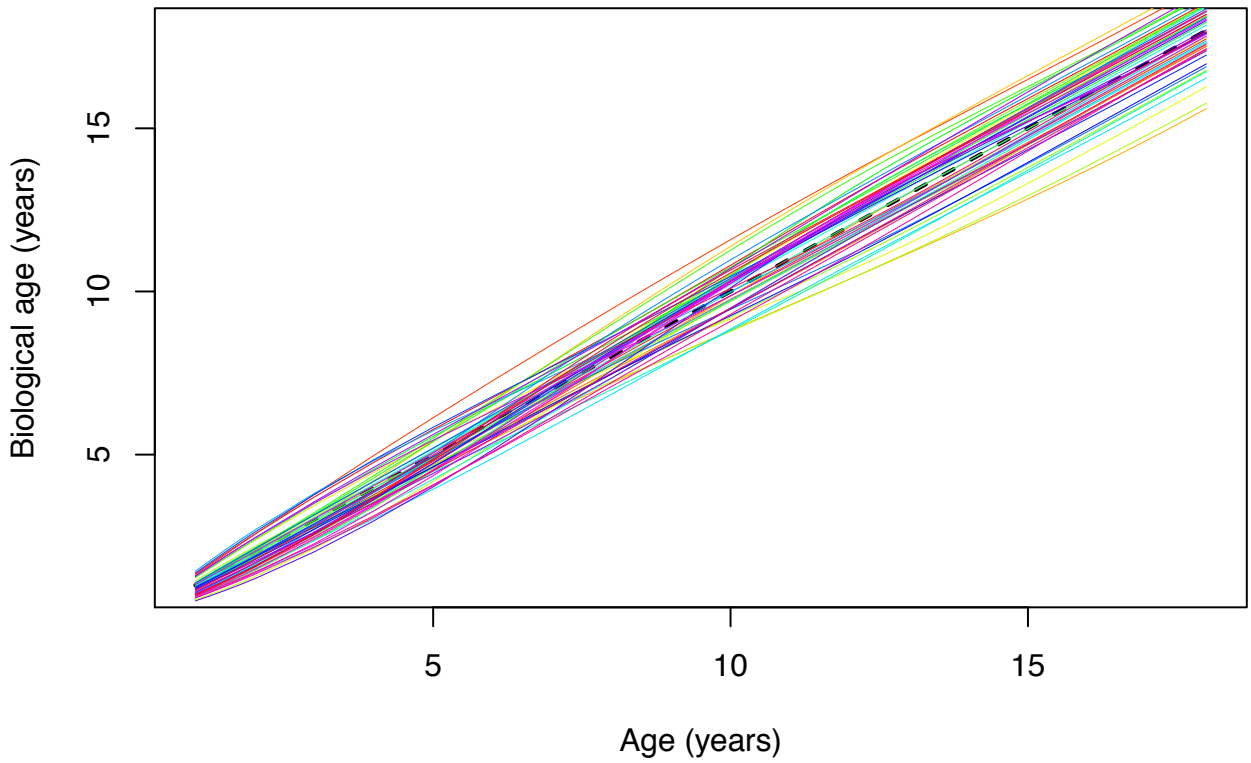
```r
#
# Plot results
#

t_p <- seq(range(t)[1], range(t)[2], length = 100)

# Functional fixed effect
theta <- basis_fct(t_p) %*% res$c

# Display data with predictions
plot(t_p, theta, ylim = range(y), type = 'n', main = 'Original heights and predicted',
     xlab = 'Age (years)', ylab = 'Height (cm)')
for (i in 1:n) {
  points(t[[i]], y[[i]], pch = 19, cex = 0.3, col = rainbow(n)[i])
  lines(t_p, predict_curve(t_p, t[[i]], y[[i]], basis_fct, res$c, warp_fct, res$w[, i],
                           amp_cov, res$amp_cov_par),
        lwd = 0.5, col = rainbow(n)[i])

}
lines(t_p, theta, ylim = range(y), lwd = 2, lty = 2)
```




```
# Compute and display growth velocities
plot(t_p, t_p, ylim = c(0, 23), type = 'n', main = 'Predicted growth velocities',
     xlab = 'Age (years)', ylab = 'Growth velocity (cm/year)')
for (i in 1:n) {
  lines(t_p, predict_curve(t_p, t[[i]], y[[i]], basis_fct, res$c, warp_fct, res$w[, i],
                           amp_cov, res$amp_cov_par, deriv = TRUE),
        lwd = 0.5, col = rainbow(n)[i])
}
```




```
# Display predicted warping functions
plot(t_p, t_p, type = 'l', lwd = 2, lty = 2, main = 'Warping functions',
     xlab = 'Age (years)', ylab = 'Biological age (years)')
for (i in 1:n) lines(t[[i]], warp_fct(res$w[,i], t[[i]]), lwd = 0.4, col = rainbow(n)[i])
```




```
# Display estimated warp covariance
res$sigma^2 * warp_cov(tw[2:(mw + 1)], param = res$warp_cov_par)
```

```
#>           [,1]      [,2]      [,3]      [,4]
#> [1,] 0.2477081 0.2477081 0.2477081 0.2477081
#> [2,] 0.2477081 0.4954162 0.4954162 0.4954162
#> [3,] 0.2477081 0.4954162 0.7431242 0.7431242
#> [4,] 0.2477081 0.4954162 0.7431242 0.9908323
```

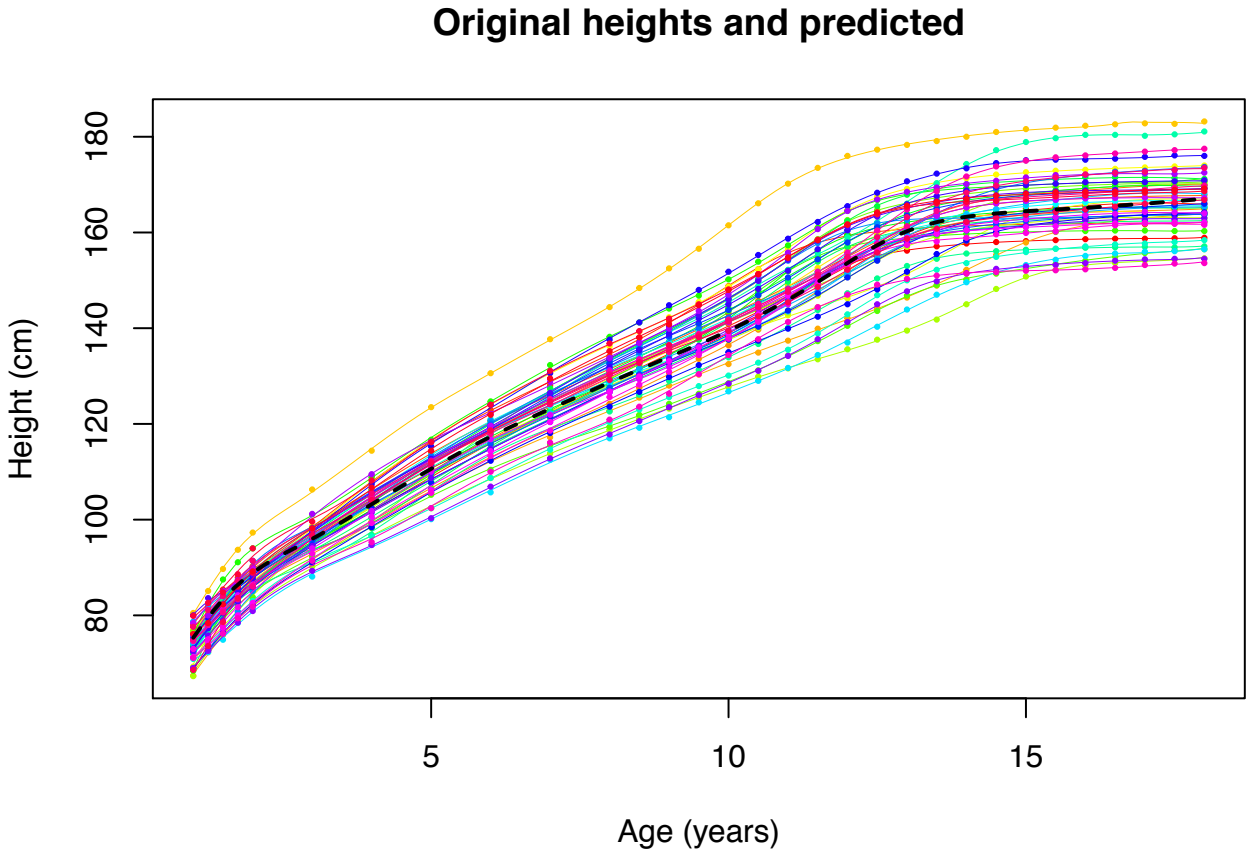
Random B-spline basis model

```
# Set up amplitude function and covariance
amp_fct <- make_basis_fct(df = 10, type = 'B-spline', intercept = TRUE, control = list(boundary = t_rang
# amp_cov <- make_cov_fct(id_cov, noise = FALSE, param = 300)
amp_cov <- make_cov_fct(diag_cov, noise = FALSE, param = rep(10, attr(amp_fct, 'df')))

# Estimate in the model

res <- pavpop(y, t, basis_fct, warp_fct, amp_cov, warp_cov, amp_fct, homeomorphisms = 'soft')
```

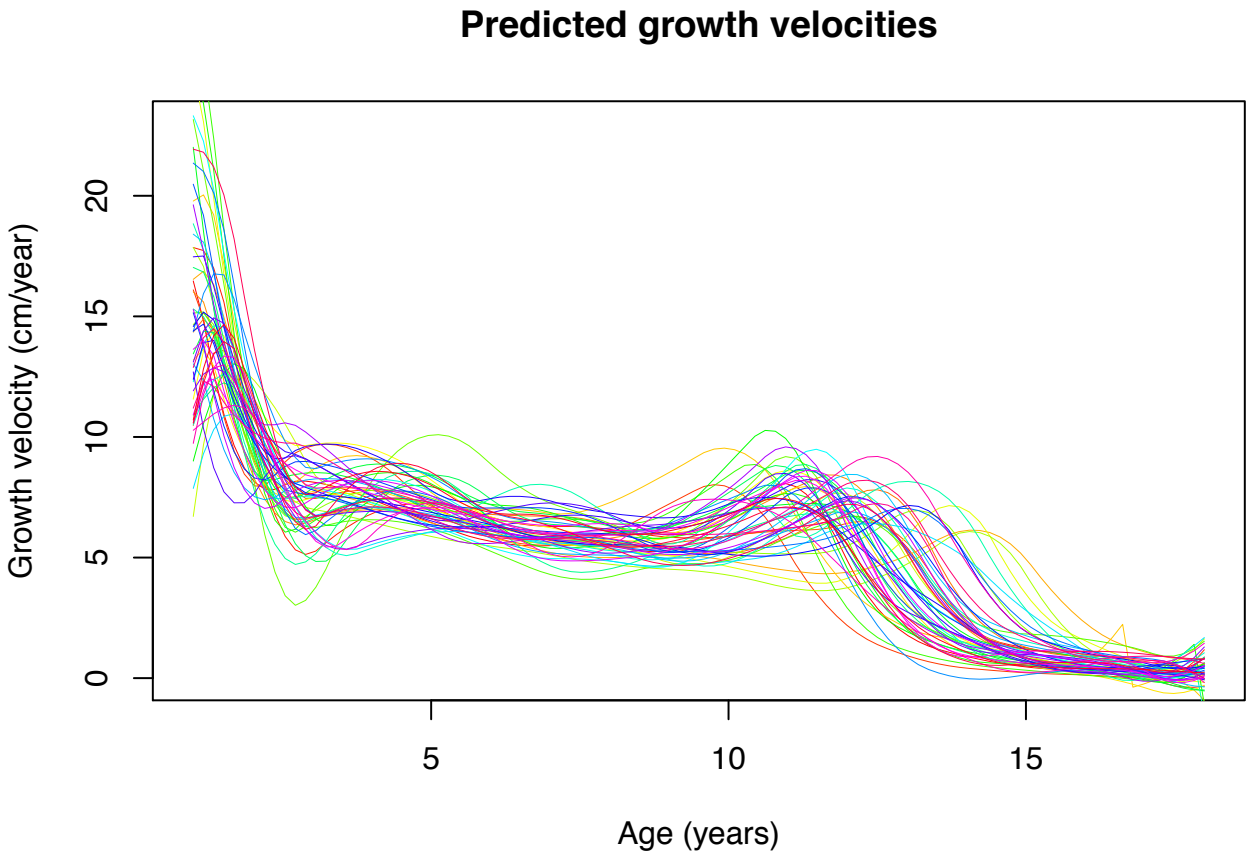
```
#> Outer    :   Inner   :   Estimates
#> 1    :   1   2   3   4   5   :    304.8309 213.0862 33.69864 41.13664 70.64906 212.2164 124.7177 183
#> Linearized likelihood:    -200.9149
```

```
#> 2    :   1   2   3   4   5   :    319.3977 217.6171 113.2815 44.03097 65.78571 212.2155 146.144 185.
#> Linearized likelihood:    -530.2299
#> 3    :   1   2   3   4   5   :    378.3275 236.996 83.57589 88.76444 59.25961 210.9888 229.4694 202.
#> Linearized likelihood:    -556.4917
#> 4    :   1   2   3   4   5   :    403.0438 248.1912 97.5146 54.70789 79.27733 211.8761 240.5799 210.
#> Linearized likelihood:    -572.3026
#> 5    :   1   2   3   4   5   :    458.4046 269.5867 92.60224 70.7864 57.69492 215.3344 261.8908 225.
#> Linearized likelihood:    -576.7103
```

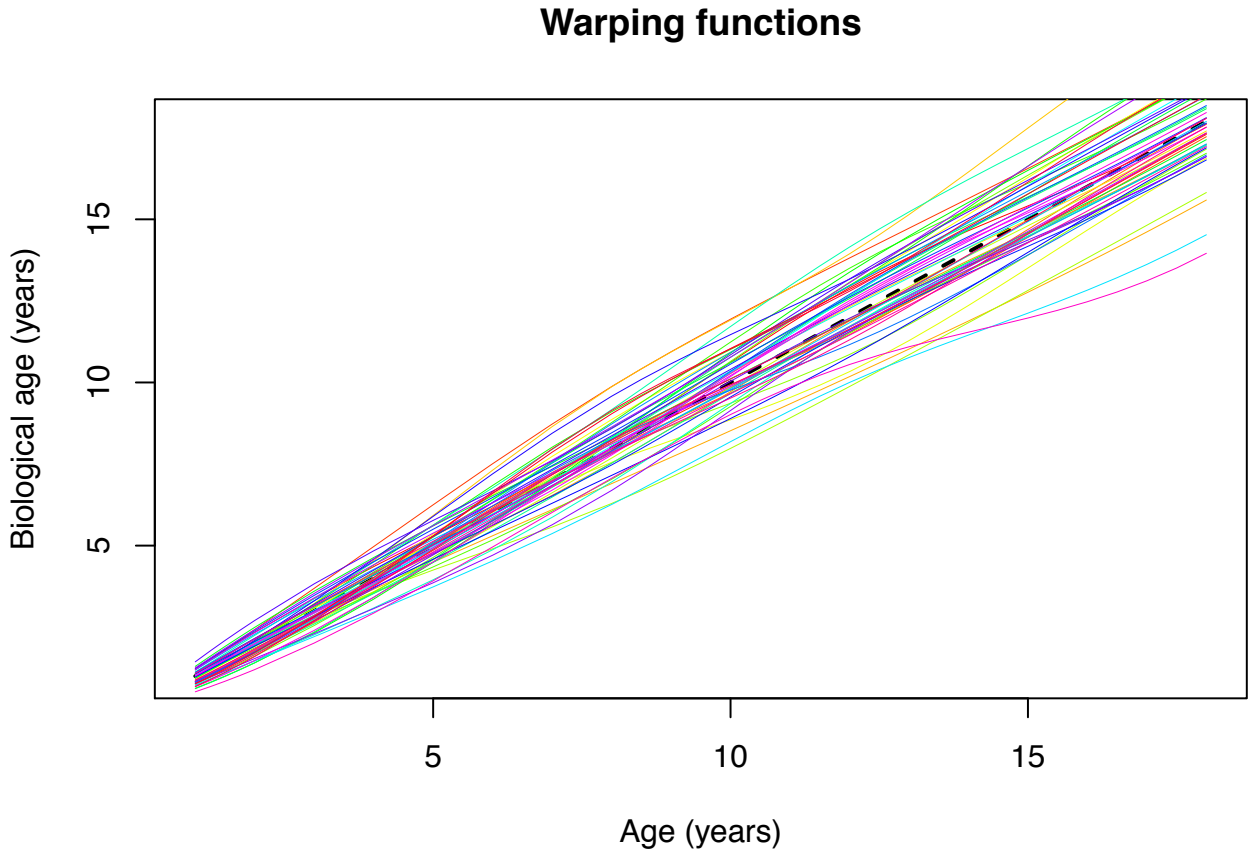
```r
#
# Plot results
#

# Functional fixed effect
theta <- basis_fct(t_p) %*% res$c

# Display data with predictions
plot(t_p, theta, ylim = range(y), type = 'n', main = 'Original heights and predicted',
     xlab = 'Age (years)', ylab = 'Height (cm)')
for (i in 1:n) {
  points(t[[i]], y[[i]], pch = 19, cex = 0.3, col = rainbow(n)[i])
  lines(t_p, predict_curve(t_p, t[[i]], y[[i]], basis_fct, res$c, warp_fct, res$w[, i],
                           amp_cov, res$amp_cov_par, amp_fct = amp_fct),
        lwd = 0.5, col = rainbow(n)[i])
}

lines(t_p, theta, ylim = range(y), lwd = 2, lty = 2)
```




```
# Compute and display growth velocities
plot(t_p, t_p, ylim = c(0, 23), type = 'n', main = 'Predicted growth velocities',
     xlab = 'Age (years)', ylab = 'Growth velocity (cm/year)')
for (i in 1:n) {
  lines(t_p, predict_curve(t_p, t[[i]], y[[i]], basis_fct, res$c, warp_fct, res$w[, i],
                           amp_cov, res$amp_cov_par, amp_fct = amp_fct, deriv = TRUE),
        lwd = 0.5, col = rainbow(n)[i])
}
```




```
# Display predicted warping functions
plot(t_p, t_p, type = 'l', lwd = 2, lty = 2, main = 'Warping functions',
     xlab = 'Age (years)', ylab = 'Biological age (years)')
for (i in 1:n) lines(t[[i]], warp_fct(res$w[,i], t[[i]]), lwd = 0.4, col = rainbow(n)[i])
```




```
# Display estimated warp covariance
res$sigma^2 * warp_cov(tw[2:(mw + 1)], param = res$warp_cov_par)
```

```
#>           [,1]      [,2]      [,3]      [,4]
#> [1,] 0.3559386 0.3559386 0.3559386 0.3559386
#> [2,] 0.3559386 0.7118773 0.7118773 0.7118773
#> [3,] 0.3559386 0.7118773 1.0678159 1.0678159
#> [4,] 0.3559386 0.7118773 1.0678159 1.4237545
```

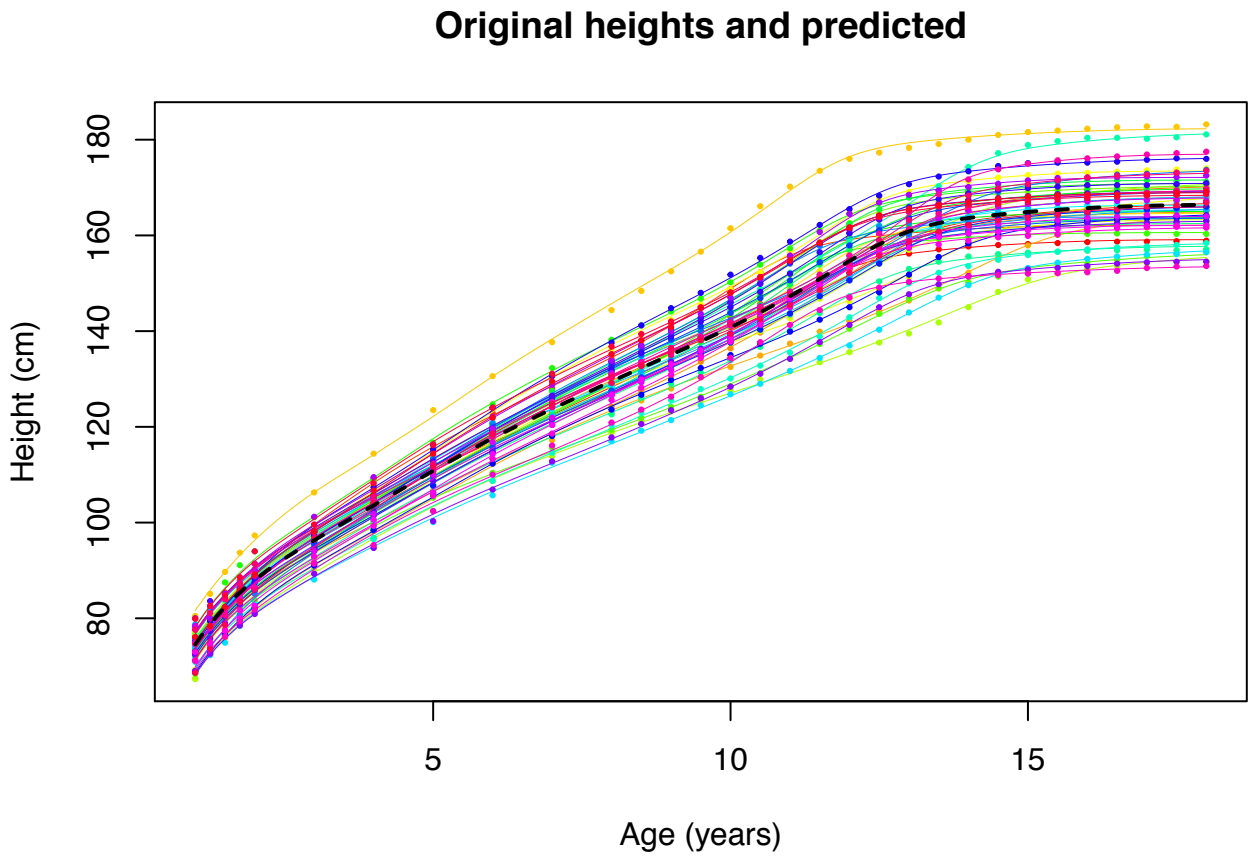
Random intercept model

```
# Set up amplitude function and covariance
amp_fct <- make_basis_fct(type = 'intercept')
amp_cov <- make_cov_fct(id_cov, noise = FALSE, param = 300)

# Estimate in the model

res <- pavpop(y, t, basis_fct, warp_fct, amp_cov, warp_cov, amp_fct, homeomorphisms = 'soft')
```

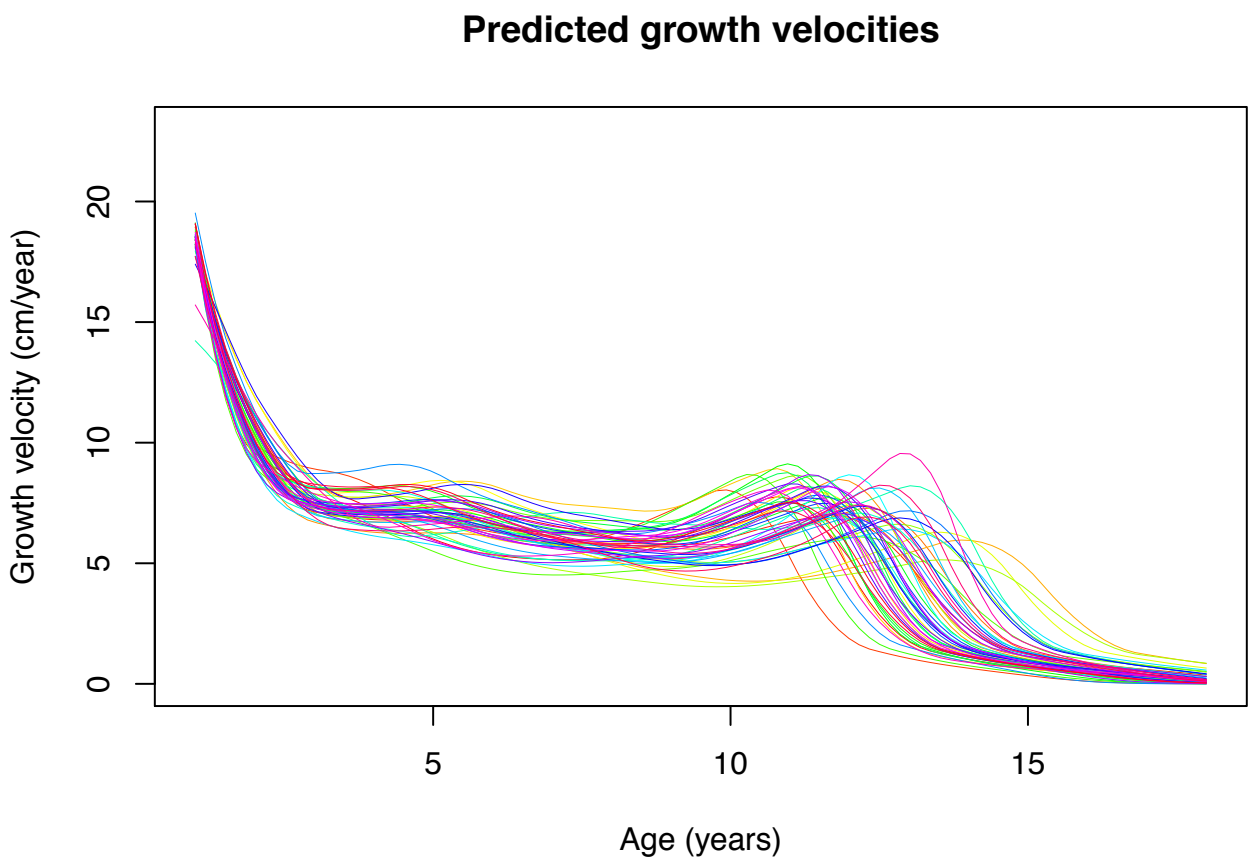
```
#> Outer    :   Inner   :   Estimates
#> 1    :   1   2   3   4   5   :    57.64654 0.3603482
#> Linearized likelihood:    118.0603
#> 2    :   1   2   3   4   5   :    73.59907 0.3565992
```

```
#> Linearized likelihood:    62.38817
#> 3    :   1   2   3   4   5   :    73.59908 0.3554691
#> Linearized likelihood:    59.02058
#> 4    :   1   2   3   4   5   :    73.59911 0.3550493
#> Linearized likelihood:    56.33105
#> 5    :   1   2   3   4   5   :    73.59912 0.3549287
#> Linearized likelihood:    54.27849
```

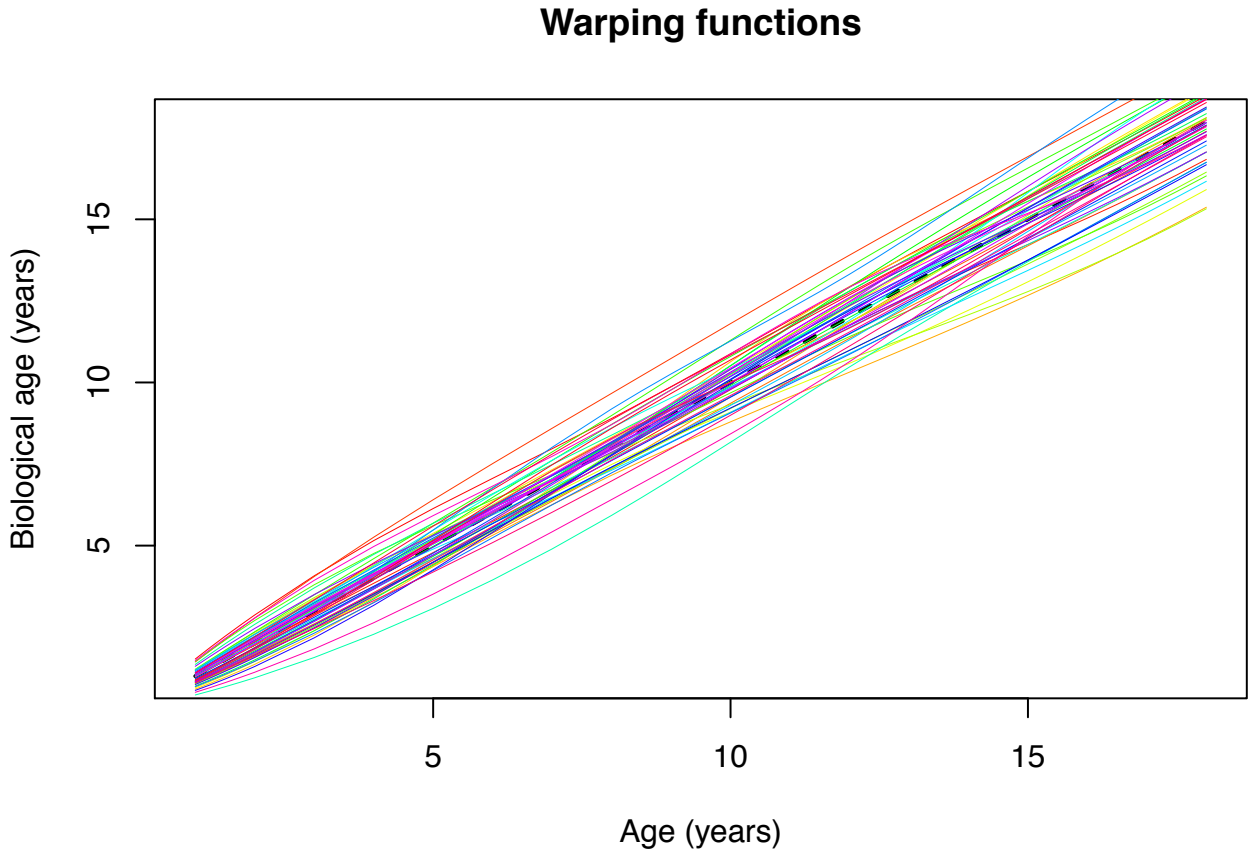
```r
#
# Plot results
#

# Functional fixed effect
theta <- basis_fct(t_p) %*% res$c

# Display data with predictions
plot(t_p, theta, ylim = range(y), type = 'n', main = 'Original heights and predicted',
     xlab = 'Age (years)', ylab = 'Height (cm)')
for (i in 1:n) {
  points(t[[i]], y[[i]], pch = 19, cex = 0.3, col = rainbow(n)[i])
  lines(t_p, predict_curve(t_p, t[[i]], y[[i]], basis_fct, res$c, warp_fct, res$w[, i],
                           amp_cov, res$amp_cov_par, amp_fct = amp_fct),
        lwd = 0.5, col = rainbow(n)[i])
}

lines(t_p, theta, ylim = range(y), lwd = 2, lty = 2)
```



```r
# Compute and display growth velocities
plot(t_p, t_p, ylim = c(0, 23), type = 'n', main = 'Predicted growth velocities',
     xlab = 'Age (years)', ylab = 'Growth velocity (cm/year)')
for (i in 1:n) {
  lines(t_p, predict_curve(t_p, t[[i]], y[[i]], basis_fct, res$c, warp_fct, res$w[, i],
                           amp_cov, res$amp_cov_par, amp_fct = amp_fct, deriv = TRUE),
        lwd = 0.5, col = rainbow(n)[i])
}
```




```r
# Display predicted warping functions
plot(t_p, t_p, type = 'l', lwd = 2, lty = 2, main = 'Warping functions',
     xlab = 'Age (years)', ylab = 'Biological age (years)')
for (i in 1:n) lines(t[[i]], warp_fct(res$w[,i], t[[i]]), lwd = 0.4, col = rainbow(n)[i])
```




```
# Display estimated warp covariance
res$sigma^2 * warp_cov(tw[2:(mw + 1)], param = res$warp_cov_par)
```

```
#>           [,1]      [,2]      [,3]      [,4]
#> [1,] 0.2488725 0.2488725 0.2488725 0.2488725
#> [2,] 0.2488725 0.4977450 0.4977450 0.4977450
#> [3,] 0.2488725 0.4977450 0.7466174 0.7466174
#> [4,] 0.2488725 0.4977450 0.7466174 0.9954899
```

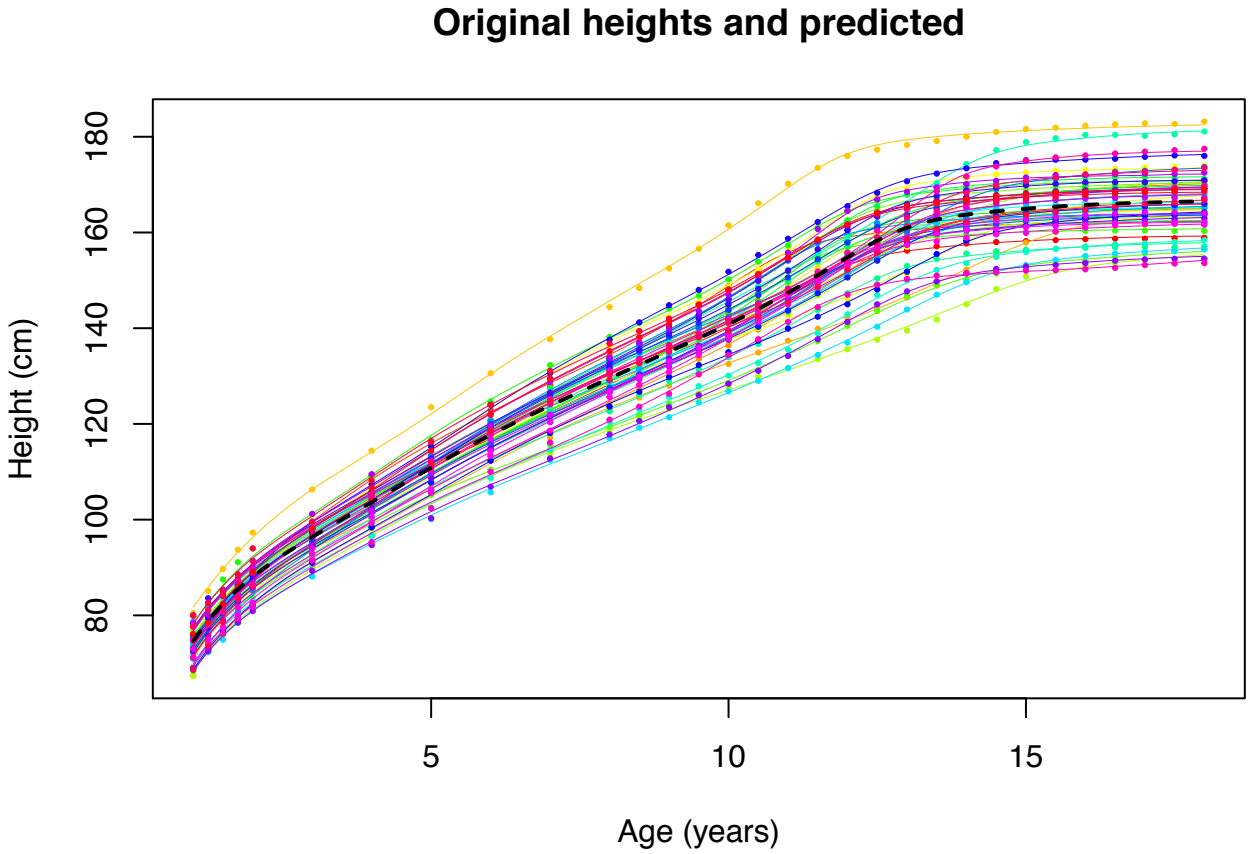
Random intercept model with free warp covariance

```
# Set up amplitude function and covariance
amp_fct <- make_basis_fct(type = 'intercept')
amp_cov <- make_cov_fct(id_cov, noise = FALSE, param = 300)

# Set warp covariance to be free
warp_cov <- make_cov_fct(unstr_cov, noise = FALSE, param = c(rep(10, mw), rep(0, mw * (mw - 1) / 2)))

# Estimate in the model

res <- pavpop(y, t, basis_fct, warp_fct, amp_cov, warp_cov, amp_fct, homeomorphisms = 'soft')
```

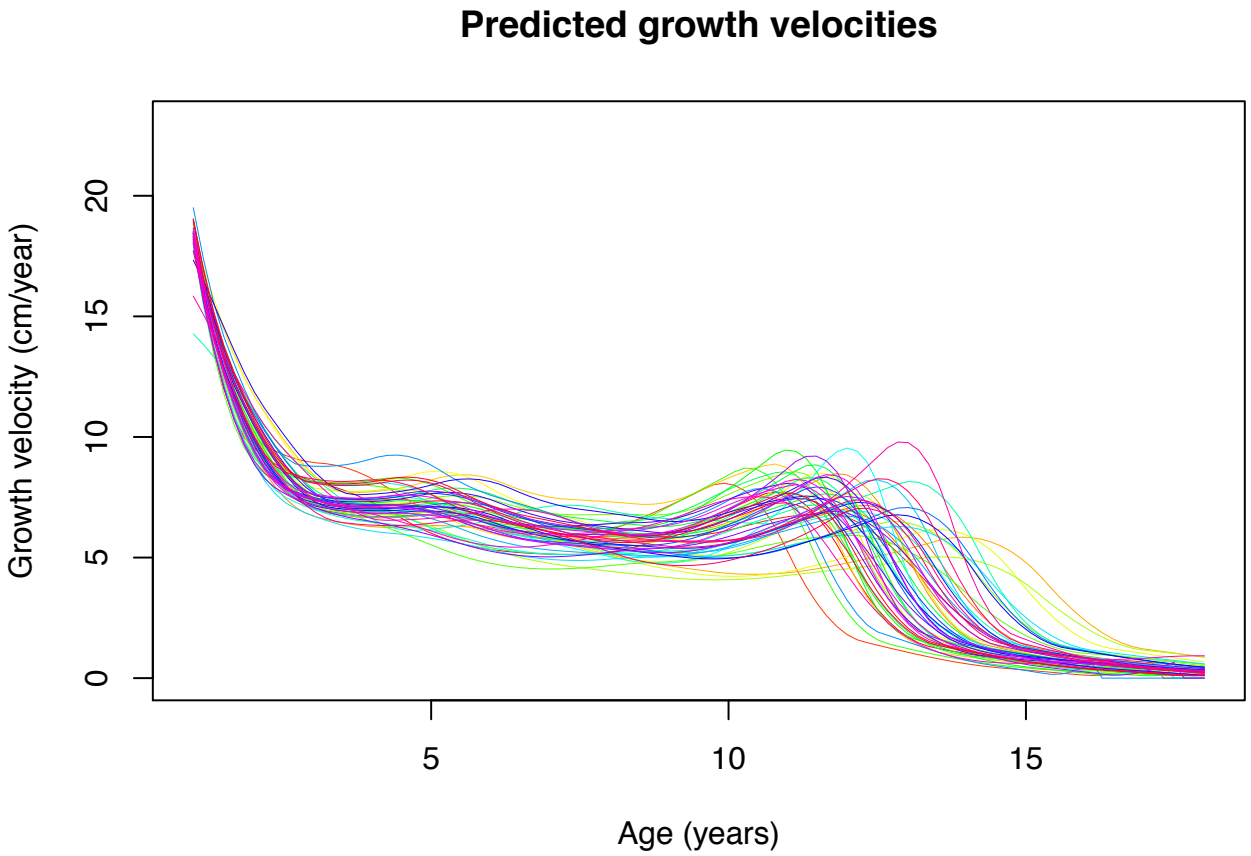
```
#> Outer    :   Inner    :   Estimates
```

```
#> 1    :   1   2   3   4   5   :    299.8103 2.649945 2.246822 2.760359 6.789382 2.114091 1.407377 2.1
#> Linearized likelihood:    117.8467
#> 2    :   1   2   3   4   5   :    299.8004 1.719261 2.23652 2.564312 6.612237 1.83218 1.663584 2.199
#> Linearized likelihood:    79.7593
#> 3    :   1   2   3   4   5   :    299.7869 1.255108 1.782373 2.511204 6.640173 1.367811 1.386651 1.9
#> Linearized likelihood:    66.77258
#> 4    :   1   2   3   4   5   :    299.7476 0.9700968 1.421578 1.907082 7.116022 1.036604 0.9439651 1
#> Linearized likelihood:    55.79284
#> 5    :   1   2   3   4   5   :    299.738 0.9042405 1.28328 1.973515 7.01776 0.9413618 0.9175333 1.4
#> Linearized likelihood:    52.39324
```

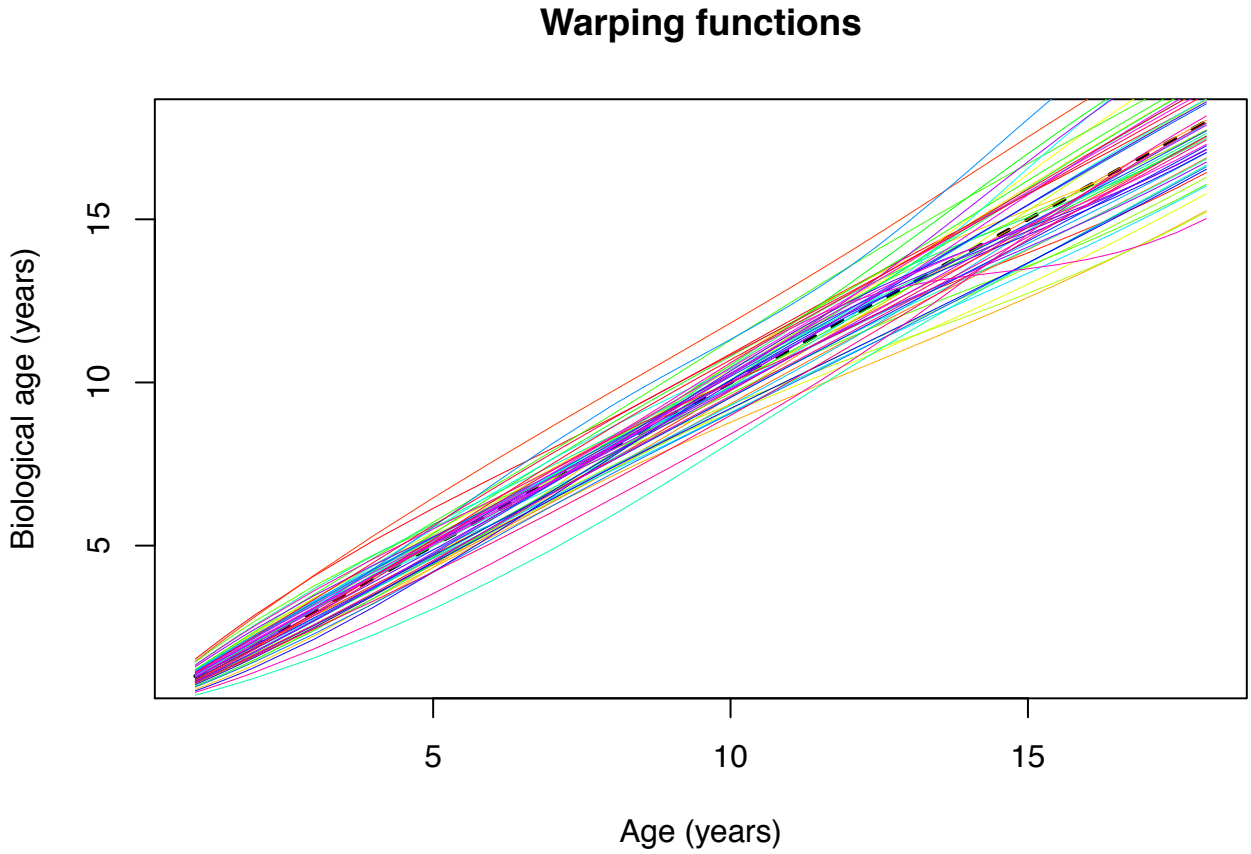
```
#
# Plot results
#

# Functional fixed effect
theta <- basis_fct(t_p) %*% res$c

# Display data with predictions
plot(t_p, theta, ylim = range(y), type = 'n', main = 'Original heights and predicted',
     xlab = 'Age (years)', ylab = 'Height (cm)')
for (i in 1:n) {
  points(t[[i]], y[[i]], pch = 19, cex = 0.3, col = rainbow(n)[i])
  lines(t_p, predict_curve(t_p, t[[i]], y[[i]], basis_fct, res$c, warp_fct, res$w[, i],
                           amp_cov, res$amp_cov_par, amp_fct = amp_fct),
        lwd = 0.5, col = rainbow(n)[i])
}

lines(t_p, theta, ylim = range(y), lwd = 2, lty = 2)
```




```
# Compute and display growth velocities
plot(t_p, t_p, ylim = c(0, 23), type = 'n', main = 'Predicted growth velocities',
     xlab = 'Age (years)', ylab = 'Growth velocity (cm/year)')
for (i in 1:n) {
  lines(t_p, predict_curve(t_p, t[[i]], y[[i]], basis_fct, res$c, warp_fct, res$w[, i],
                           amp_cov, res$amp_cov_par, amp_fct = amp_fct, deriv = TRUE),
        lwd = 0.5, col = rainbow(n)[i])
}
```




```
# Display predicted warping functions
plot(t_p, t_p, type = 'l', lwd = 2, lty = 2, main = 'Warping functions',
     xlab = 'Age (years)', ylab = 'Biological age (years)')
for (i in 1:n) lines(t[[i]], warp_fct(res$w[,i], t[[i]]), lwd = 0.4, col = rainbow(n)[i])
```




```
# Display estimated warp covariance
res$sigma^2 * warp_cov(1:mw, param = res$warp_cov_par)
```

```
#>           [,1]      [,2]      [,3]      [,4]
#> [1,] 0.4197233 0.4369540 0.4258935 0.6620671
#> [2,] 0.4369540 0.5956628 0.6620671 1.0805286
#> [3,] 0.4258935 0.6620671 0.9160511 1.4645002
#> [4,] 0.6620671 1.0805286 1.4645002 3.2574496
```