Functional Regression
Using the `fda` Package in R

Spencer Graves, Giles Hooker, James Ramsay

Ramsay, Hooker and Graves (2009) *Functional Data Analysis with R and Matlab* (Springer)

This Presentation

- What Is Functional Regression?
- Different types of Functional Regression
- fRegress.numeric: Scalar Response
- fRegress.fdPar: Functional response, x = scalar
- fRegress.fdPar: Concurrent Functional Model
- fRegress.formula: Simple fRegress Setup
- linmod: Full Integration Regression
- pda.fd: Estimating a Differential Equation
- Closing Remarks
- References

What Is Functional Regression?

Functional Data Analysis extends spline smoothing to:

- an arbitrary finite basis approximation to a function space
- smoothing with an arbitrary linear differential operator

Functional regression = fitting a model where

- the response or
- an explanatory variable

is a function.

Different types of Functional Regression

Functional regression = fitting a model where

- the response or
- an explanatory variable

is a function.

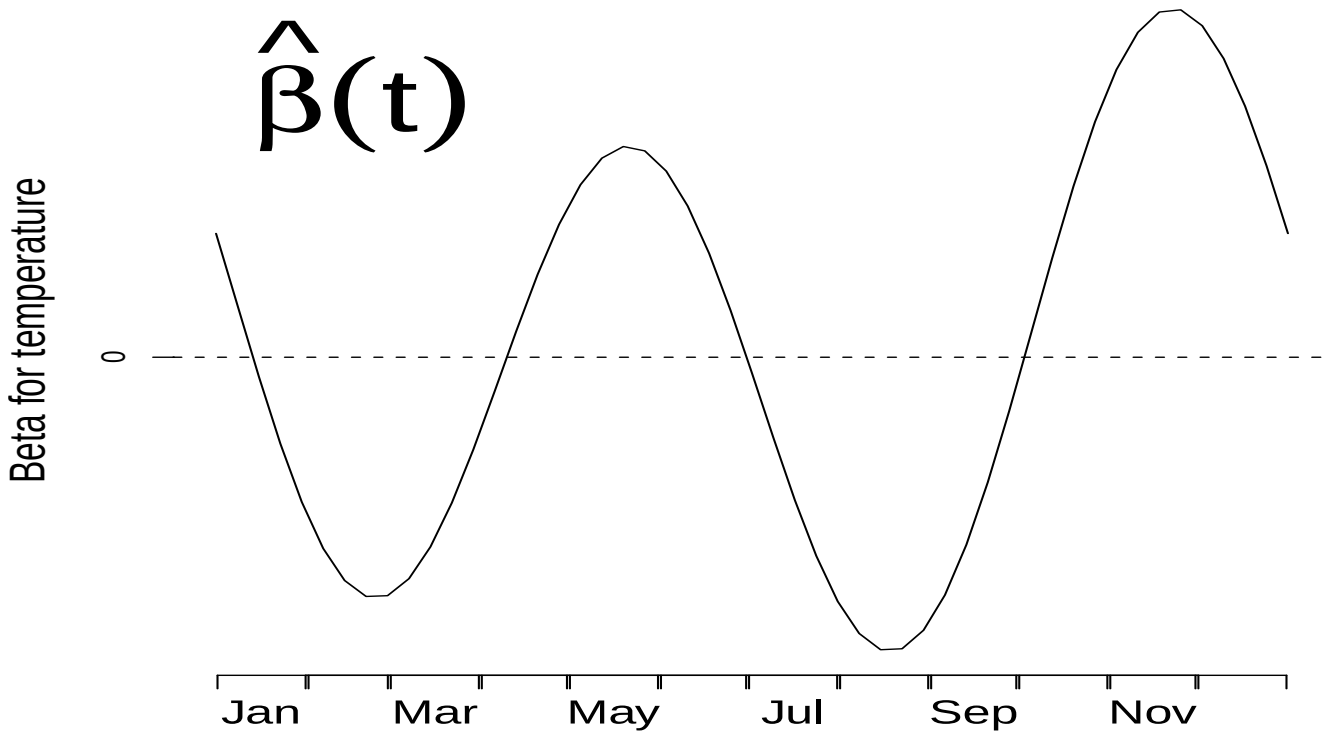
	Explanatory Variable	
response	*scalar*	*function*
scalar	lm	fRegress.numeric
function	fRegress.fdPar	fRegress.fdPar / linmod / pda.df

R code for all of these appears in script files in the `fda` package

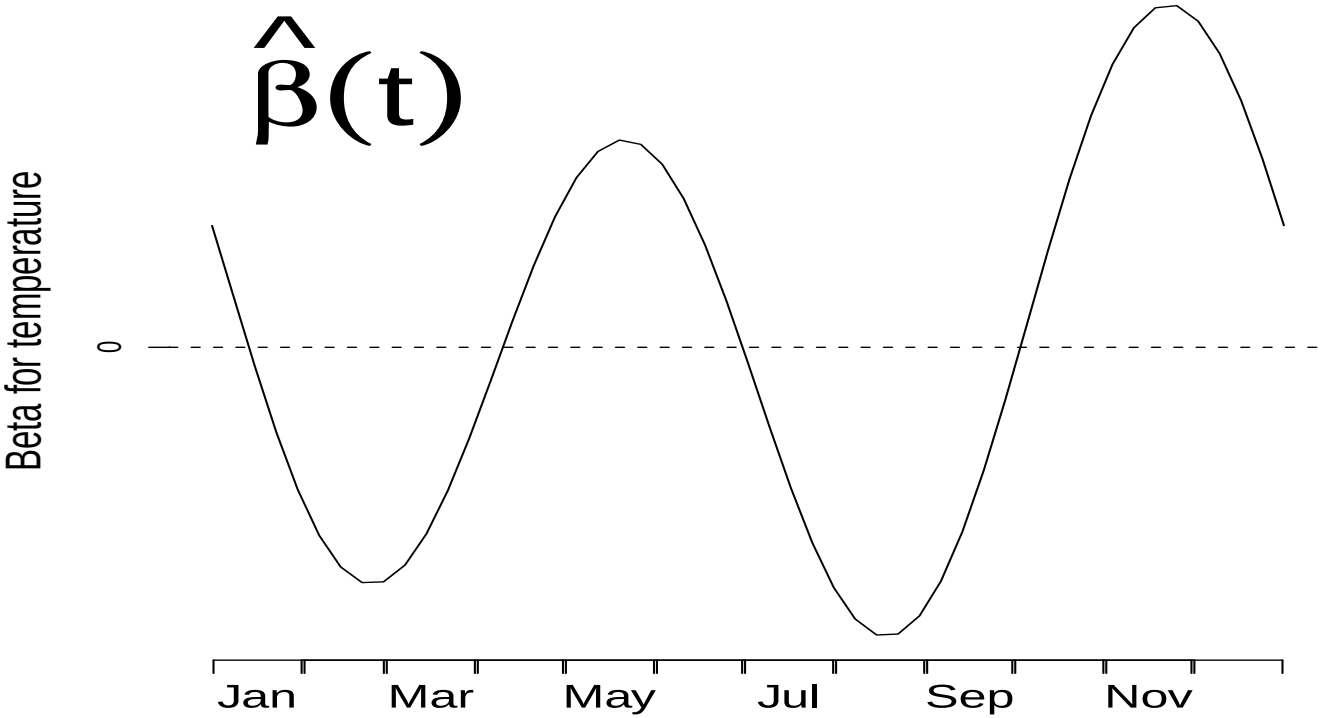
fRegress.numeric: Scalar Response

$$y_i = \alpha_0 + \int x_i(t)\beta(t)dt + \epsilon_i.$$

`log(annual precipitation) ~ (temperature profile)`



log(annual precipitation) ~ temperature(t)



Conclusion: Wetter locations tend to be

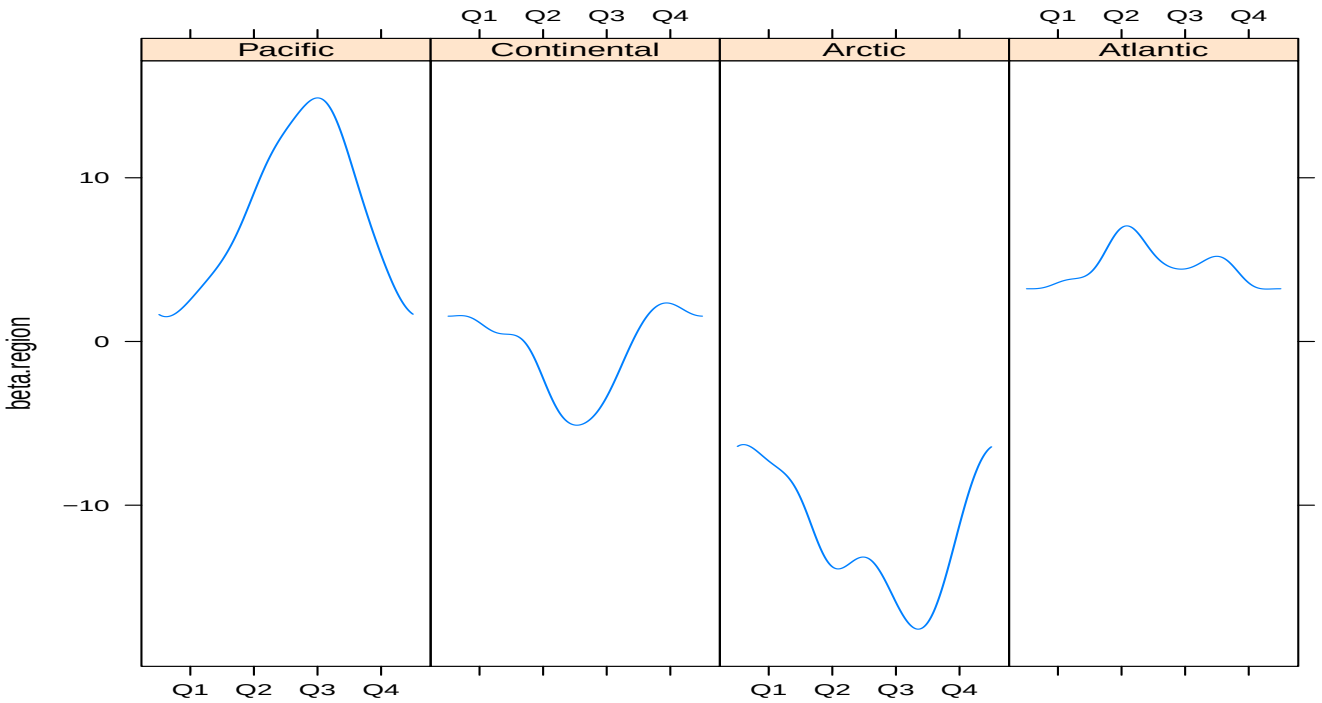
- cooler in February and August and
- warmer in May and November

Ramsay, Hooker, Graves (2009, Fig. 9.1)

fRegress.numeric: functional response, x = scalar

$$y_i(t) = \beta_0(t) + \sum x_{ij}\beta_j(t) + \epsilon_i(t)$$

`temperature ~ region`; Region Deviation:

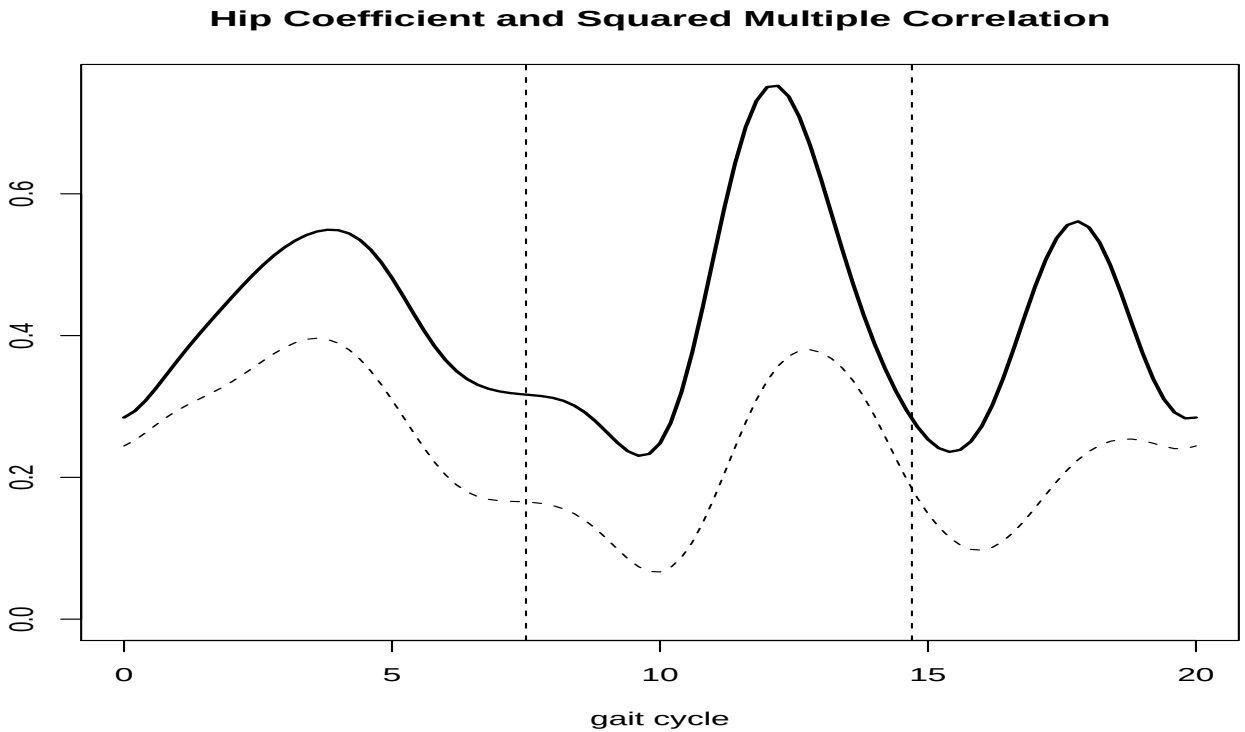


Ramsay, Hooker, Graves (2009, Fig. 10.1)

fRegress.fdPar: Concurrent Functional Model

$$y_i(t) = \beta_0(t) + \sum x_{ij}(t)\beta_j(t) + \epsilon_i(t)$$

`(knee angle) ~ (hip angle)`



Ramsay, Hooker and Graves (2009, Fig. 10.7)

fRegress.formula: Simple fRegress Setup

Traditional: fRegress(y, xlist, betalist)

Formula interface:

- model <- fRegress(y ~ x, method='model')
- model = list(y, xlist, betalist)

Manually adjust model to get what you want.

Easier than manual set up, esp. w. x factors?

linmod: Full Integration Regression

$$y_i(t) = \beta_0(t) + \int_{\Omega_t} \beta_1(t,s) x_i(s) ds + \epsilon_i(t)$$

$\beta_1(t,s)$ = bivariate regression coefficient function $\Omega_t = \{s < t\}$: historical linear model Ω_t = unconstrained: full integration regression

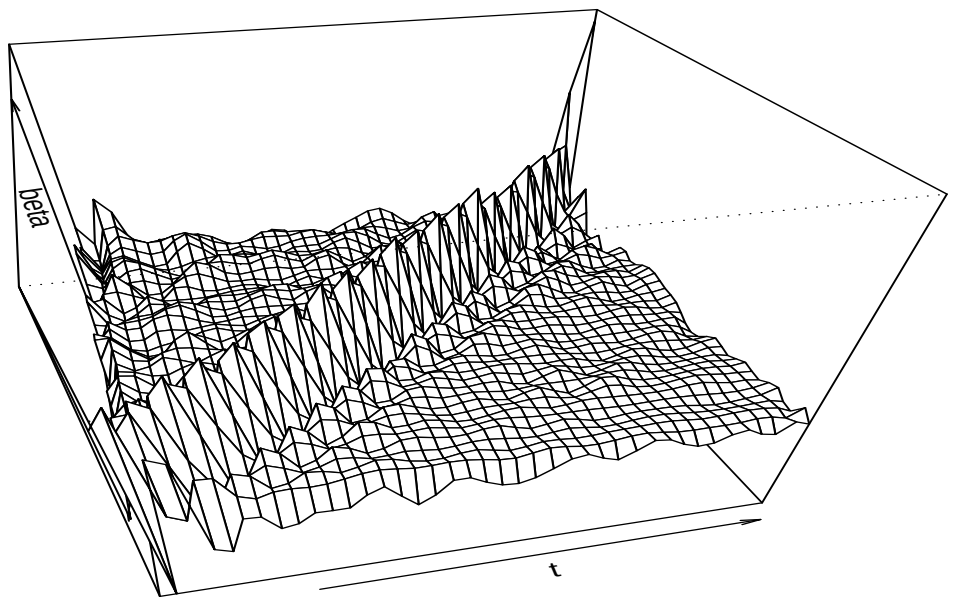
Example:
Swedish Female Mortality 1751 to 1914 Cohorts

$$x_{i+1}(t) = \beta_0(t) + \int \beta_1(s,t) x_i(t) ds + \epsilon_i(t)$$

$x_i(t)$ = log(hazard) at age t for cohort i

linmod: Full Integration Regression

$$x_{i+1}(t) = \beta_0(t) + \int \beta_1(s,t) x_i(t) ds + \epsilon_i(t)$$



Ramsay, Hooker and Graves (2009, Fig. 10.11)

pda.fd: Estimating a Differential Equation

Ramsay, Hooker and Graves (2009) *Functional Data Analysis with R and Matlab* (Springer, ch. 11)

Closing Remarks

References