Week 12: Functional Data Analysis

MATH-516 Applied Statistics

Tomas Masak

May 8th 2023

Section 1

PCA

PCA on the Population Level

- random vector $X \in \mathbb{R}^p$ such that $\mathbb{E}\|X\|^2 < \infty$
  - $\mu = \mathbb{E}X$
  - $\mathbf{C} = \mathbb{E}(X - \mu)(X - \mu)^\top$
- eigendecomposition: $\mathbf{C} = \sum_{j=1}^{p} \lambda_j e_j e_j^\top$
- define $Z_j = e_j^\top (X - m)$ leading to the expansion

$$X = \mu + \sum_{j=1}^{p} Z_j e_j$$

  - X can be represented as a weighted sum of the eigenvectors of $\mathbf{C}$ with the weights uncorrelated variables with variances that are eigenvalues of $\mathbf{C}$
- typically, we retain $r < p$ PCs and approximate $X \approx \mu + \sum_{j=1}^{q} Z_j e_j$, retaining $\sum_{j=1}^{q} \lambda_j / \sum_{j=1}^{p} \lambda_j$ proportion of variance explained
- optimality properties come from the ordering of eigenvalues $\lambda_1 \geq \lambda_2 \geq \ldots$, thus the approximation above is the optimal q-dimensional one

PCA on the Sample Level

- random sample $X_1, \ldots, X_N \in \mathbb{R}^p$ such that $\mathbb{E}\|X_1\|^2 < \infty$
  - $\mu = \mathbb{E}X_1$ estimated empirically as $\widehat{\mu} = \frac{1}{N}\sum_n X_n$
  - $\mathbf{C} = \mathbb{E}(X_1 - \mu)(X_1 - \mu)^\top$ estimated as $\widehat{\mathbf{C}} = \frac{1}{N}\sum_n (X_n - \widehat{\mu})(X_n - \widehat{\mu})^\top$
- eigendecomposition: $\widehat{\mathbf{C}} = \sum_{j=1}^{p\wedge N} \widehat{\lambda}_j \widehat{e}_j \widehat{e}_j^\top$
- for $n = 1, \ldots, N$ define $z_{nj} = \widehat{e}_j^\top (X_n - \widehat{\mu})$ leading to the expansions

$$X_n = \widehat{\mu} + \sum_{j=1}^{p\wedge N} z_{nj}\widehat{e}_j$$

- again, we can approximate by retaining $r < p \wedge N$ components only, leading to the (least squares) optimal q-dimensional approximation
- in practice, PCA is performed via SVD:
  - let $\mathbf{X} \in \mathbb{R}^{N\times p}$ be the data matrix (X_1 is the 1st row, etc.)
  - let $\mathbf{X} = \mathbf{UDV}^\top$ be its SVD
  - then $\widehat{e}_j$ is the j-th column of V and $(z_{nj}) = \mathbf{UD}$ are the scores

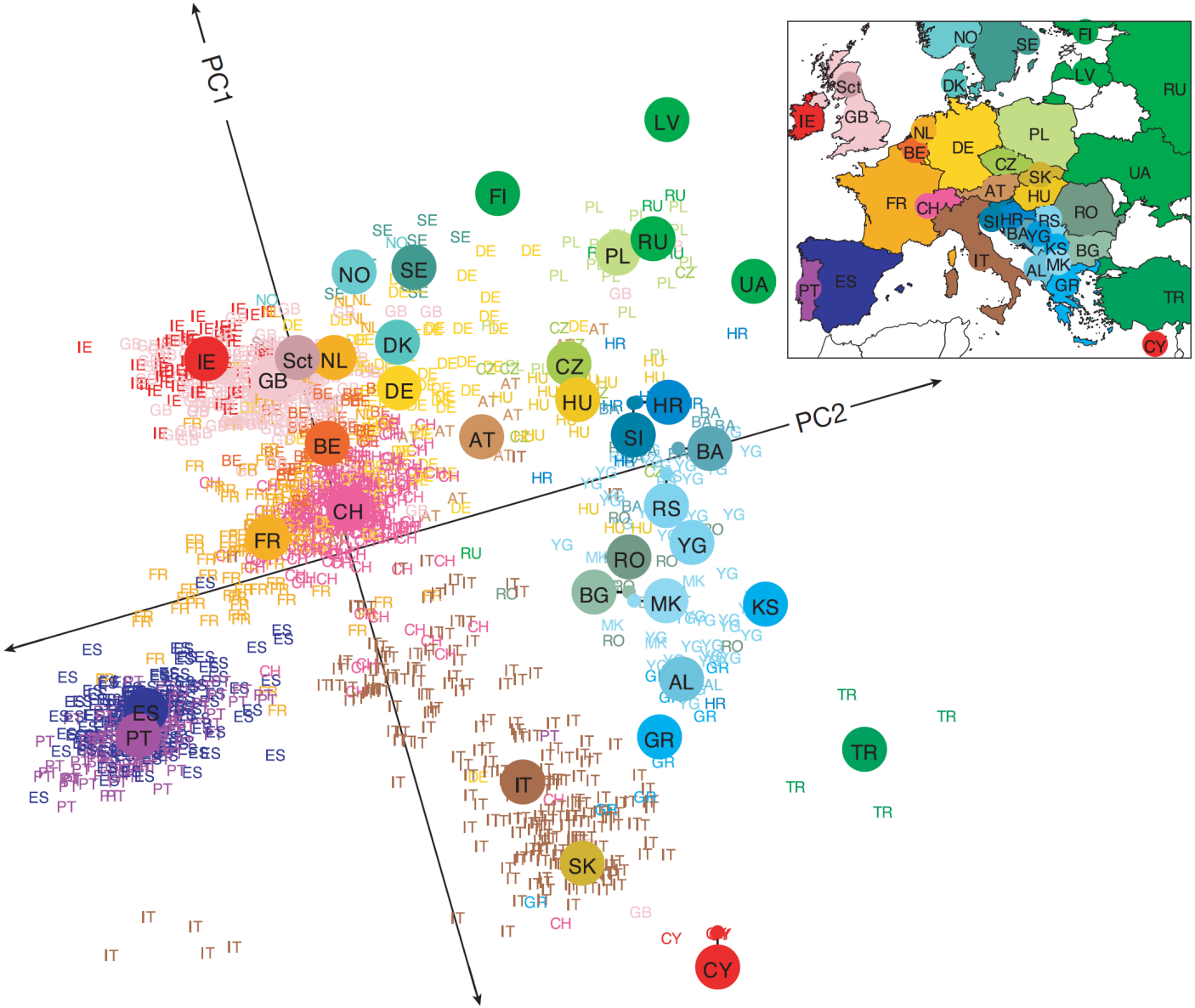
Example: Genes

Anyone has underwent a genetic ancestry testing?

- $N = 1387$ European individuals genotyped at $p = 500568$ DNA loci (using a SNP chip)
  - only individuals with European ancestry in the data set
- 2 PCs kept for *visualization* purposes (percentage of variability retained is not reported)

Source: Novembre at al. (2008) Genes mirror geography within Europe. *Nature*.

Example: Genes



Example: Running Race

- $N = 80$ runners competing in a 100 km race
- data consist of average velocity of runners on intervals of 10 kilometers, i.e. $p = 10$
- 3 PCs kept (again, percentage of variability retained not reported, but likely high)

Source: Jolliffe (2002) *Principal Component Analysis*. Springer.

Interpretation of the PCs on the next slide:

- 1st PC is all positive, hence it captures variability in overall speed
- 2nd PC contrasts the beginning against the end of the race
  - i.e. captures how much people slow down in this case
  - do not mistake with the overall slow down pattern captured by the mean!
- 3rd PC contrasts the very beginning and the very end of the race against the middle part
  - intepretation for an individual would depend on 2nd PC score, but captures e.g. the bounce-back/burn-out effect towards the very end of the race

Example: Running Race

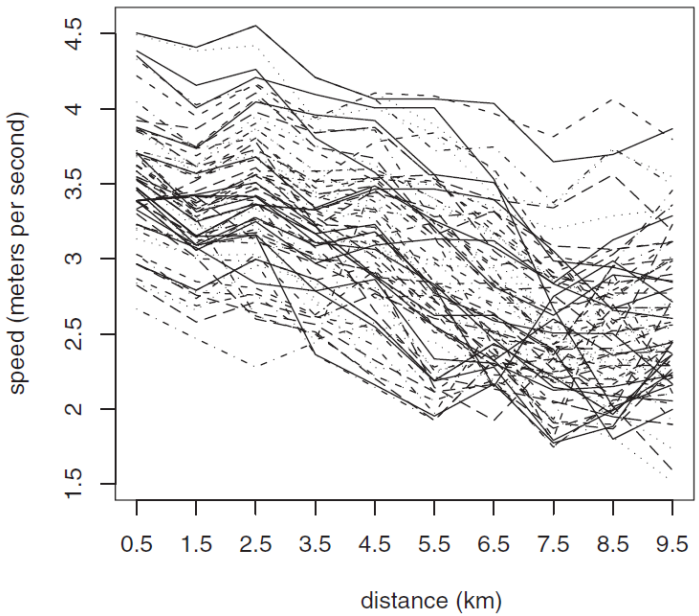


Figure 12.9. Plots of speed for 80 competitors in a 100 km race.

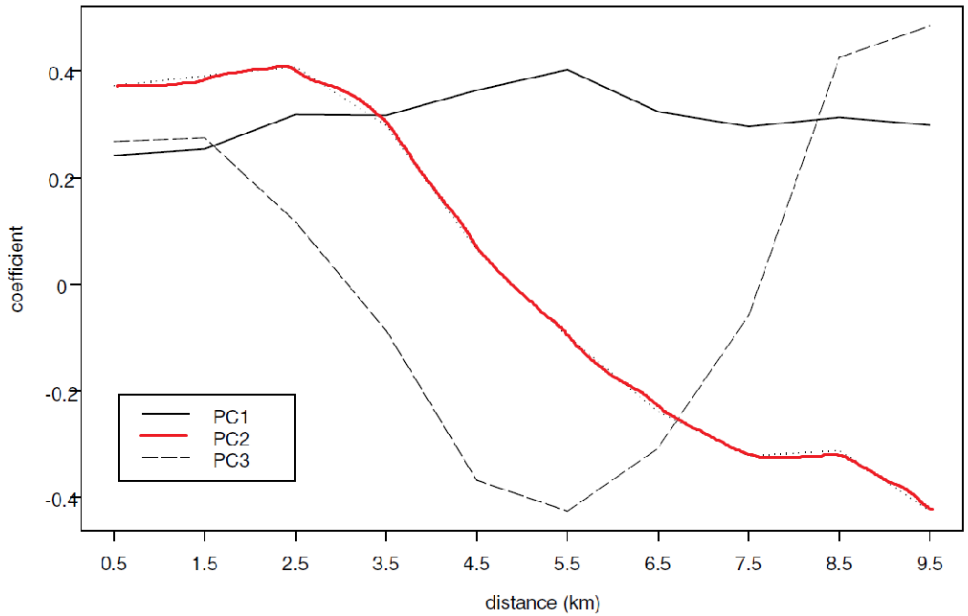


Figure 12.10. Coefficients for first three PCs from the 100 km speed data.

The Two Examples Compared

In the case of genes:

- results are very nice, but also arbitrary and lucky
- does not make sense to try to plot or interpret the data and/or the PCs
  - interpretation of the score plot comes from external data
- it is a **multivariate** example: the genome is an extremely long sequence and we subsampled it into a (still large) vector
  - we could increase p and every new locus would bring an entirely new piece of information (new degree of freedom)
  - covariance would increase in size, the new eigenvalues would not go down to zero (would not be *summable* in the limit), the leading PCs could potentially explain less and less variance

The Two Examples Compared

In the case of running race:

- it makes sense to plot the data and interpret the PCs
  - interpretation is intrinsic
  - we could also plot scores and then we could look at a specific runner and decide on his characteristics (how fast overall, how much he slowed down and how much he bounced back/burnt out towards the end) based on his position in the 3D plot
- it is a **functional** example: we have discrete measurements over a latent continuous process
  - time is continuous, runners have underlying velocity curves over time, and this curve is continuous by the laws of nature
  - if we increased p (took more measurements), every new measurement would not bring an entirely new piece of information
  - the covariance would increase in size, but the eigenvalues would go down to zero (would be *summable* in the limit), and the leading PCs would still explain approximately the same portion of variance
  - derivatives are meaningful (data points themselves derivatives of times)

Functional PCA

- random function $X \in \mathcal{L}^2[0,1]$ such that $\mathbb{E}\|X\|^2 < \infty$
  - $\mu = \mathbb{E}X$ ... now a function
  - $\mathcal{C} = \mathbb{E}(X-\mu) \otimes (X-\mu)$ ... now an operator
- $\mathcal{C}$ has a corresponding kernel c such that $c(t,s) = \mathrm{cov}(X(t), X(s))$
- eigendecomposition $\mathcal{C} = \sum_{j=1}^{\infty} \lambda_j e_j \otimes e_j$
- **Mercer's Theorem**: if c is continuous, then $c(t,s) = \sum_{j=1}^{\infty} \lambda_j e_j(t) e_j(s)$
- define $Z_j = \langle e_j, X - m \rangle$ leading to the expansion

$$X = \mu + \sum_{j=1}^{\infty} Z_j e_j \quad \text{and approximation} \quad X^{(r)} = \mu + \sum_{j=1}^{r} Z_j e_j$$

- equalities hold in mean-square sense, but under continuity also in the uniform sense

Karhunen-Loeve Theorem: If μ and c are continuous, then

$$\sup_{t \in [0,1]} \mathbb{E}\|X(t) - X^{(r)}(t)\|_2^2 \to 0 \quad \text{as } r \to \infty$$

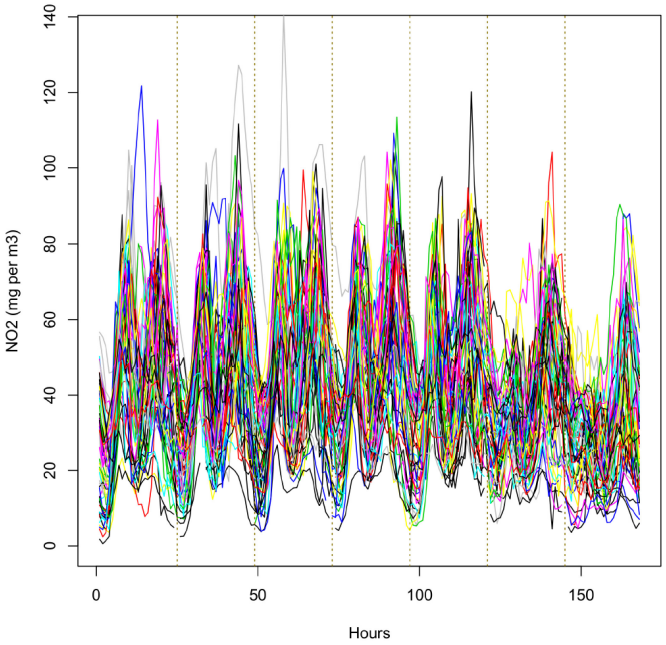
Functional PCA

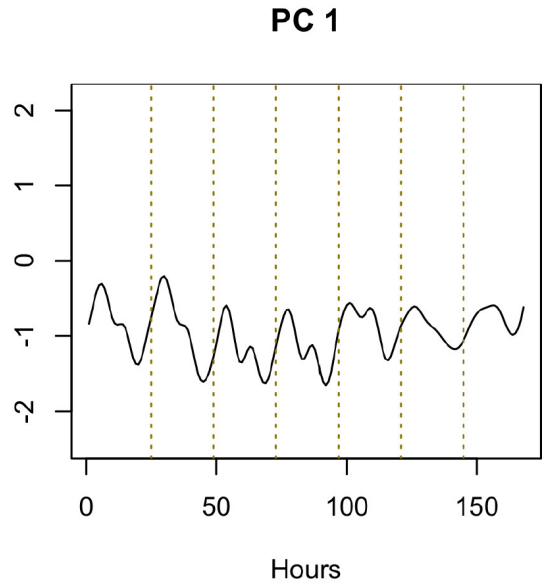
- and similarly for the sample version. . .
- data (even functional) are observed discretely, though not always on a grid
- the previous slide says that Functional PCA is the same once we decided how to treat the data
  - i.e. how measurements (data points) relate to the underlying function
- it is rather the idea of a continuous latent process that sets functional data apart

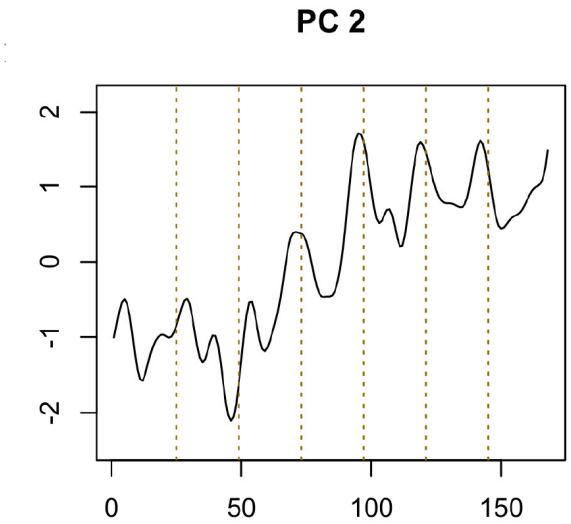
Air Quality around Lake Geneva

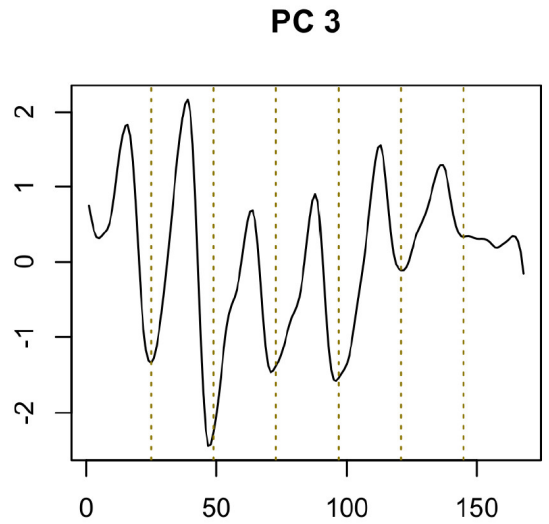
- NO2 quantity in the air recorded every hour between Sept and Nov of 2005 to 2011 and a single datum considered to be one week, i.e. $N = 62$ and $p = 7 \cdot 24 = 168$
  - how does this compare to time series?
- 3 PCs kept (again, percentage of variability retained not reported, but likely high)
- on the next slide:
  - vertical dashed lines show every midnight
  - eigenfunctions are slightly smoothed

Air Quality around Lake Geneva





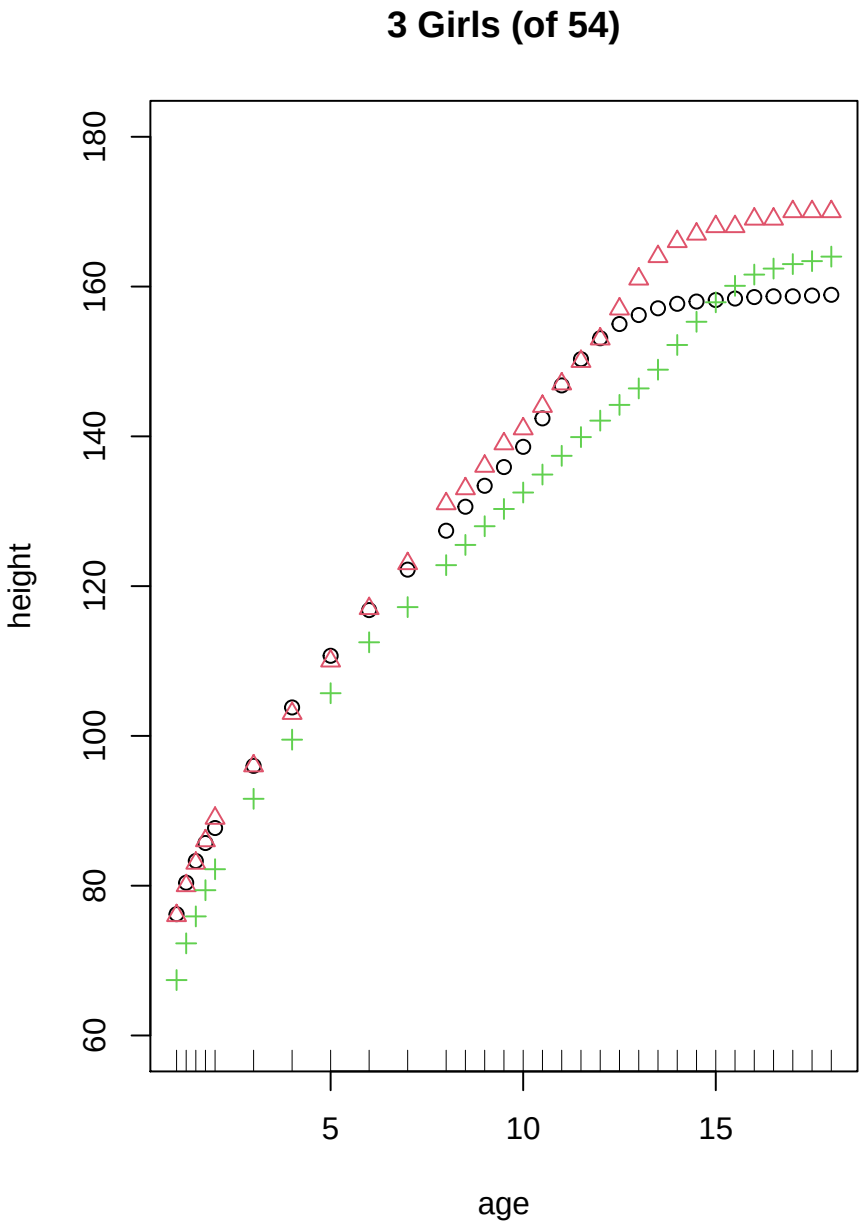


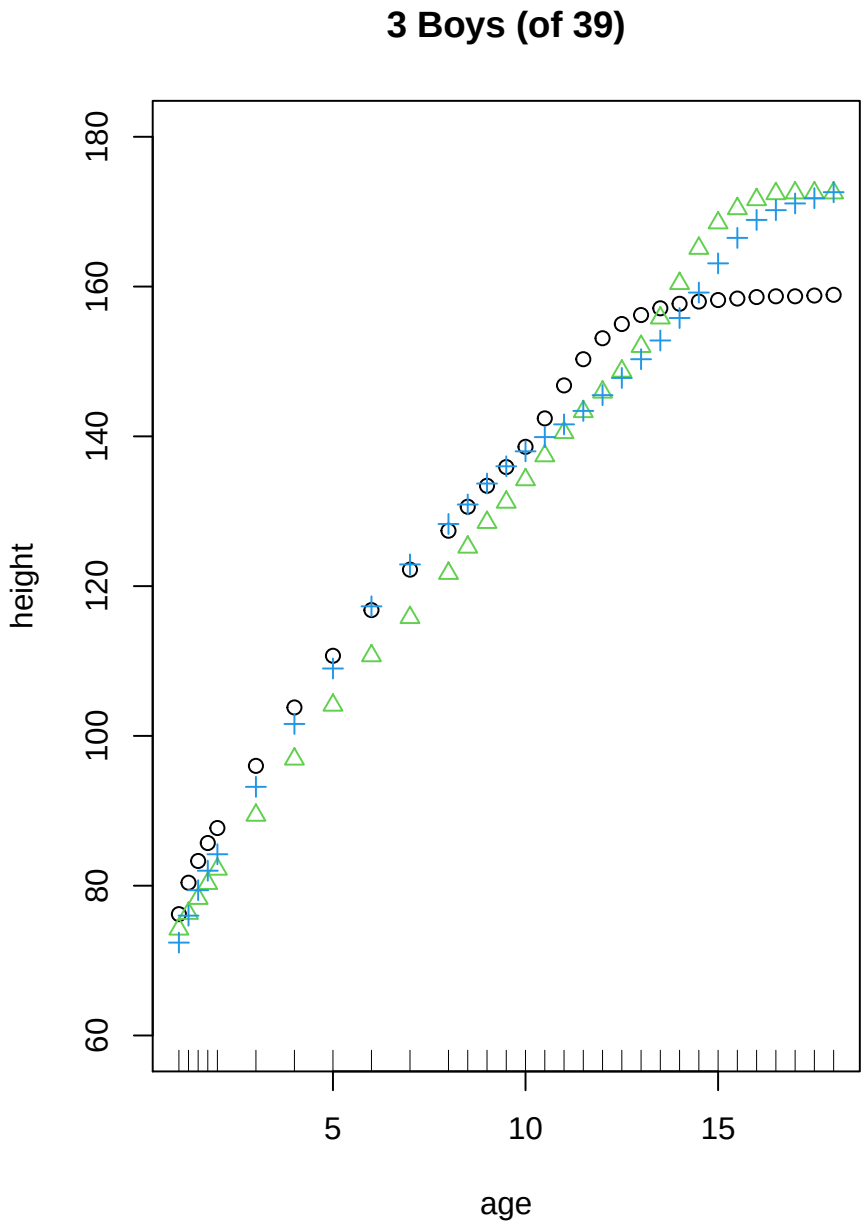


Section 2

B-splines

Berkeley Growth Data (Running Example)





Basis Representation

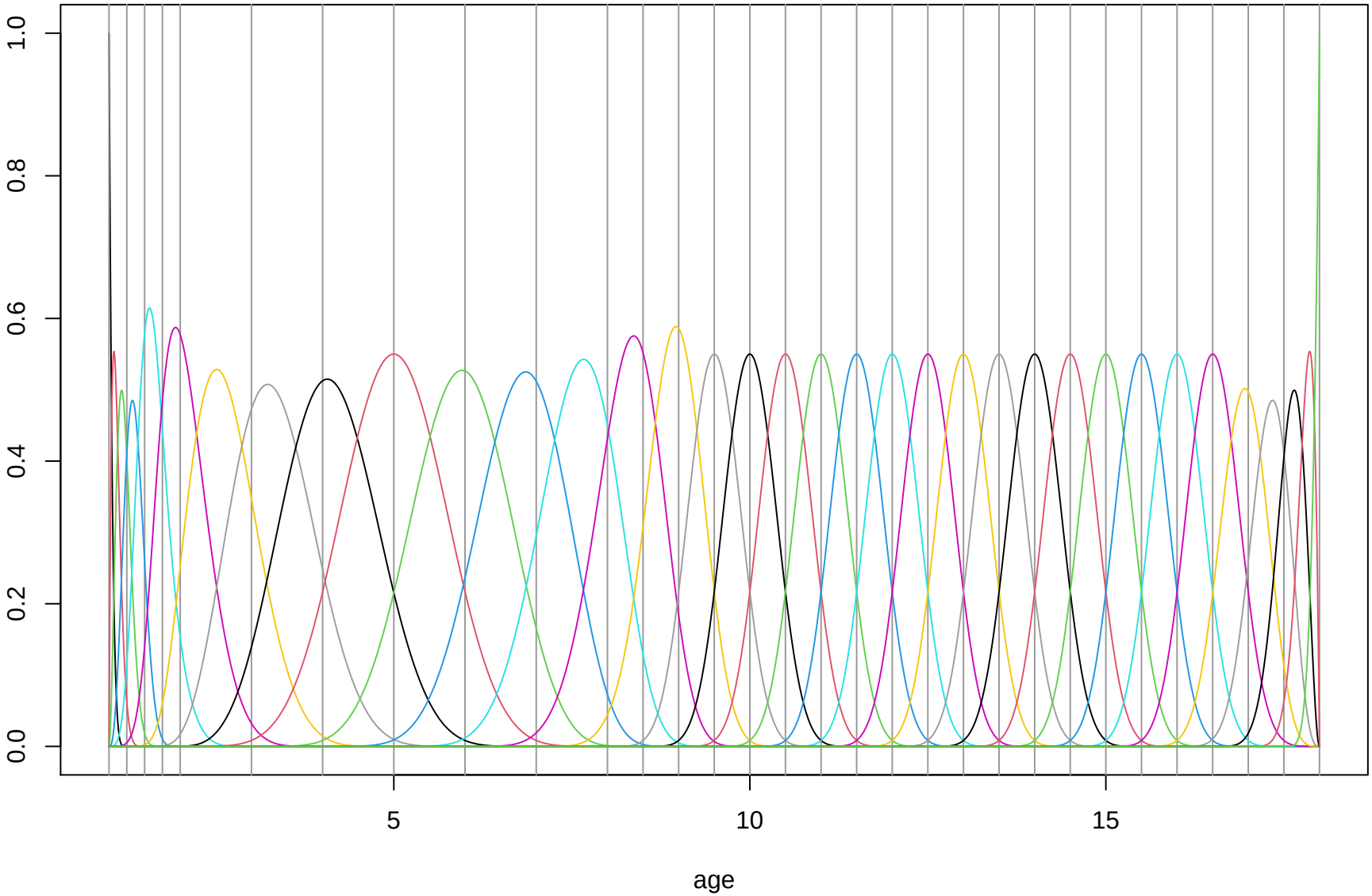
- let $y_1, \ldots, y_N$ (noisy) values of a function f at observation locations of $x_1, \ldots, x_N$ on an interval $[0, T]$
- let $\beta_1(x), \ldots, \beta_q(x)$ be basis functions chosen such that one can approximate $y(x) \doteq \sum_{j=1}^{q} \xi_j \beta_j(x)$
- let $b_{nj} = \beta_j(x_n)$ and $B = (b_{nj})$

$$\min_{\xi} \sum_{n=1}^{N} \left[y_n - \sum_{j=1}^{q} \xi_j \beta_j(x_n) \right]^2 = \min_{\xi} \|y - B\xi\|_2^2$$

- $\Rightarrow \quad \widehat{\xi} = \left(B^\top B \right)^\dagger B^\top y$
- let $z_1, \ldots, z_M \in [0, T]$ be evaluation locations
- we estimate $\left(f(z_1), \ldots, f(z_M) \right)^\top$ by $\widetilde{B}\widehat{\xi}$ where $\widetilde{B} = (\widetilde{b}_{ij})$ and $\widetilde{b}_{ij} = \beta_j(z_i)$

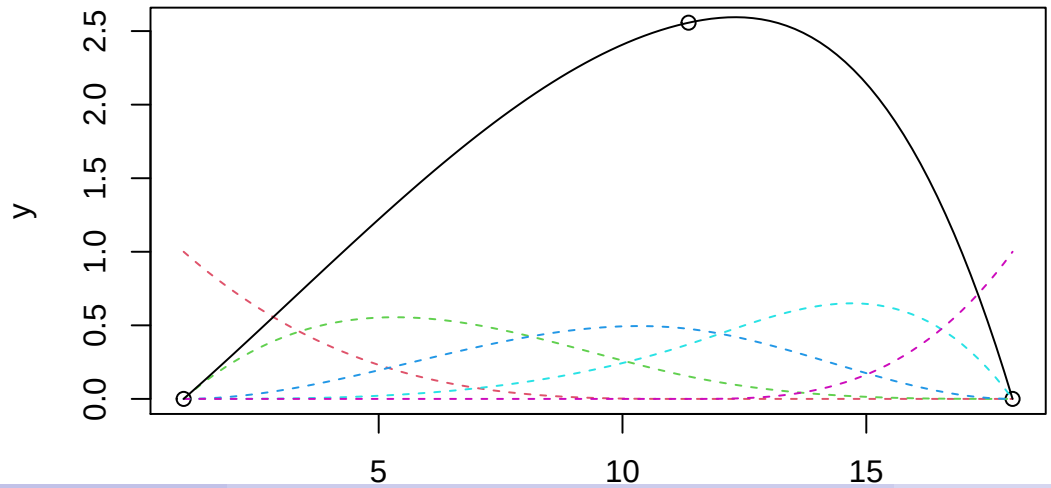
Note: This is just linear regression. If the no. of basis functions q is relatively high compared to N, use ridge regularization.

B-splines: Example



B-splines: Toy Example

```
x <- c(1.000, 11.355, 18.000)
y <- c(0.000, 2.557, 0.000)
z <- seq(range(x)[1], range(x)[2], length=100)
plot(x,y)
Bplot <- bsplineS(z, x)
for(n in 1:5) points(z,Bplot[,n],type="l",col=1+n, lty=2)
B <- bsplineS(x,x)
xi_hat <- ginv(t(B) %*% B) %*% t(B) %*% y
f_hat <- as.vector(Bplot %*% xi_hat)
points(z, f_hat, type="l")
```

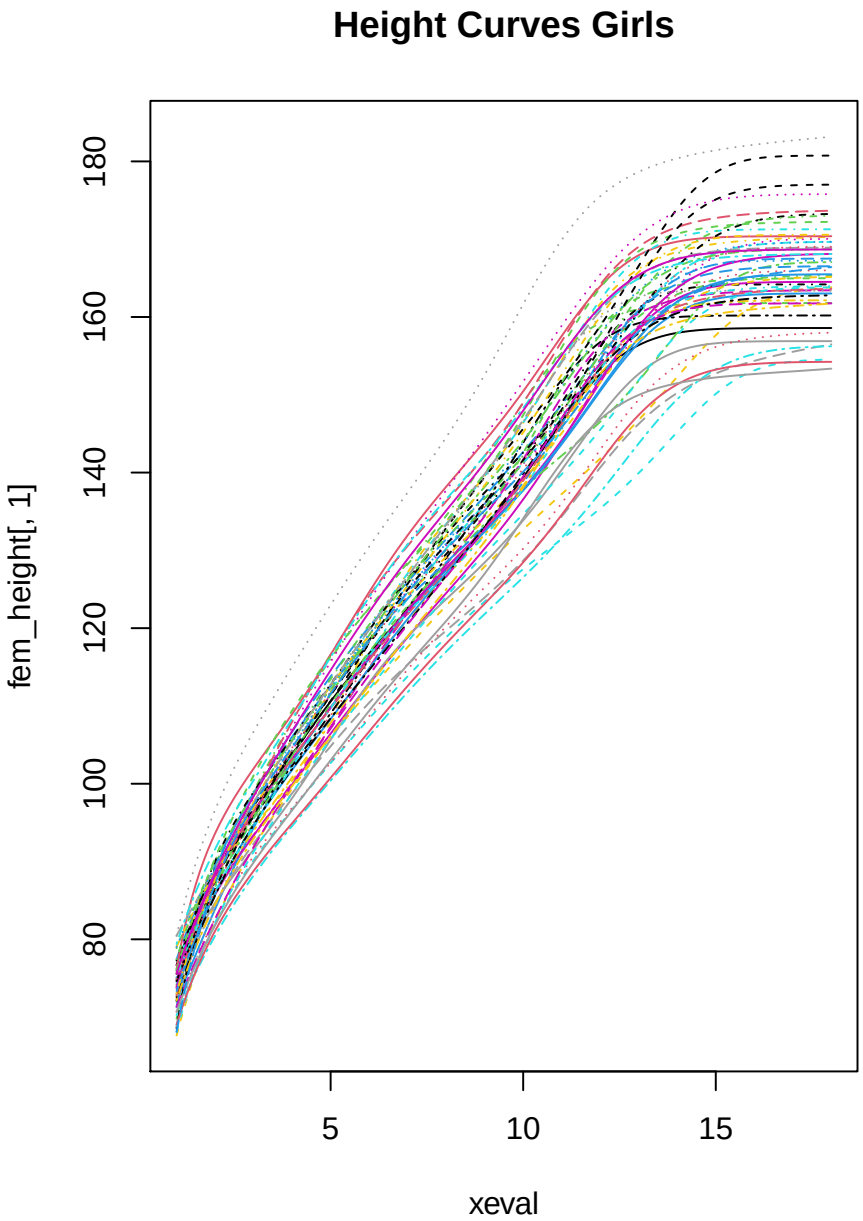


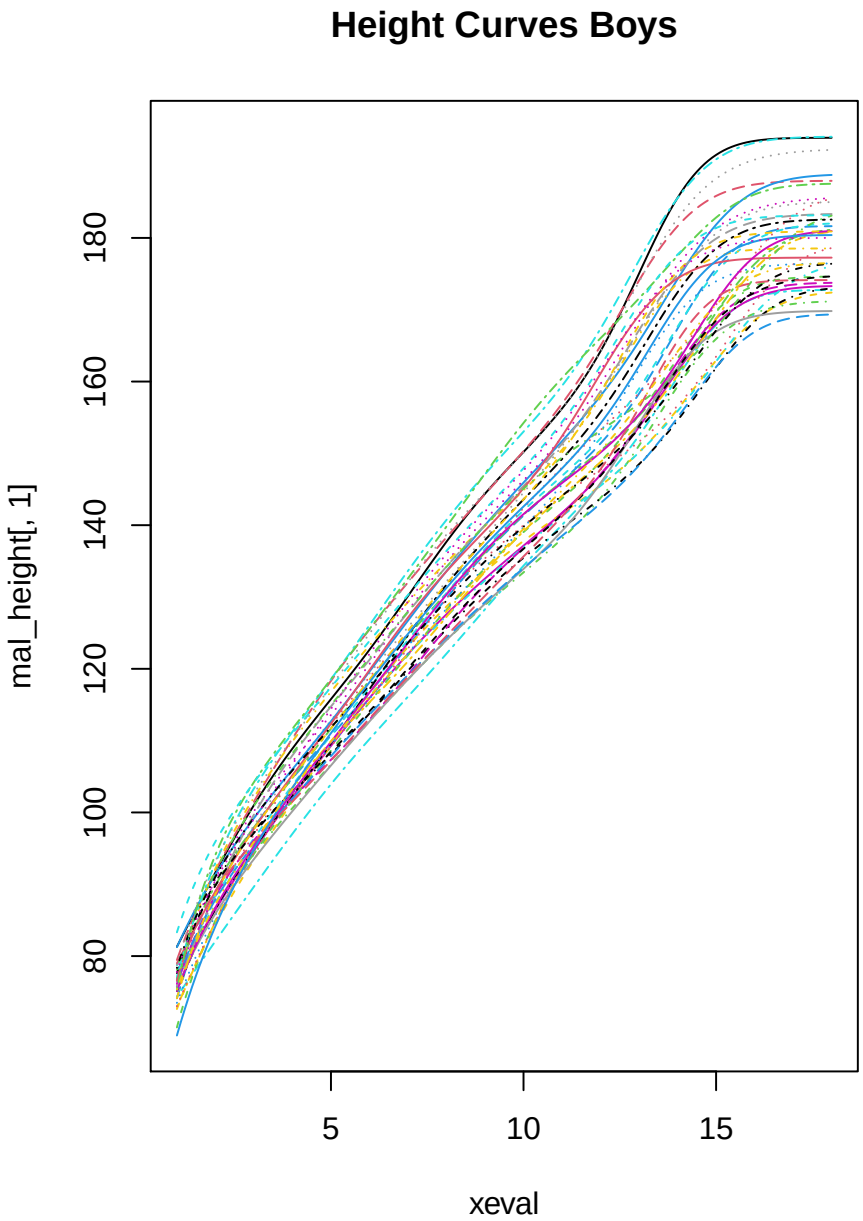
After Pre-processing (namely B-spline Smoothing)

- the `.RData` file below stores coefficients of the height curves (and velocity and acceleration, i.e. the 1st and 2nd derivatives of height) for the Berkeley growth data w.r.t the B-spline basis `B` below, which we evaluate on a grid `xeval` for plotting purposes
- here we use 6-order B-splines since we will also work with 2nd derivatives and we want to have cubic spline fit for those (order is one higher than the polynomial degree)

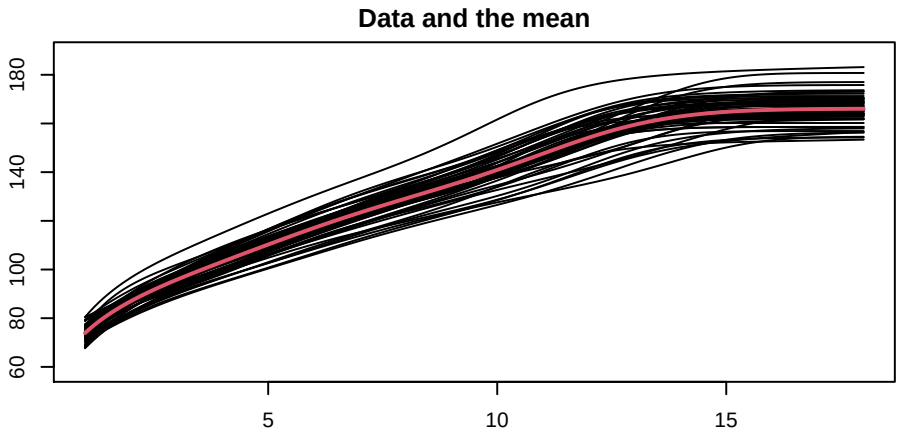
```
load("../Data/Berkeley_growth_preprocessed.RData") # Data
xeval <- seq(1,18,by=0.1)
B <- bsplineS(xeval, norder=6, breaks=growth$ag)
fem_height <- B %*% Data$fem_height
mal_height <- B %*% Data$mal_height
par(mfrow=c(1,2))
plot(xeval, fem_height[,1],type="l", ylim=range(fem_height), main="Height Curves Gi
for(n in 2:dim(Data$fem_height)[2]) points(xeval, fem_height[,n],
                                           type="l",col=n,lty=n%%6+1)
plot(xeval, mal_height[,1],type="l", ylim=range(mal_height), main="Height Curves Bo
for(n in 2:dim(Data$mal_height)[2]) points(xeval, mal_height[,n],
                                           type="l",col=n,lty=n%%6+1)
```

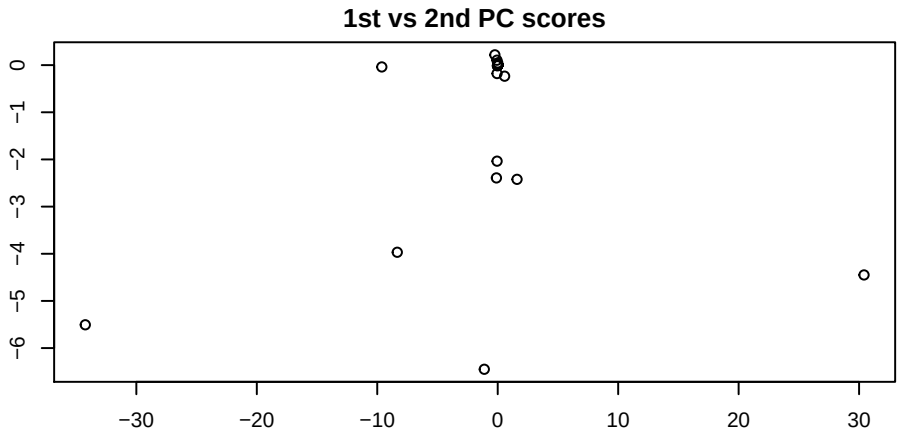
After Pre-processing (namely B-spline Smoothing)

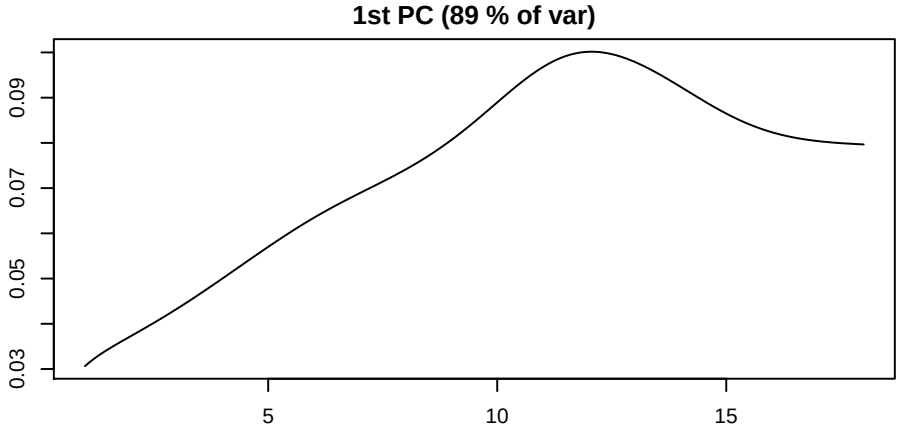


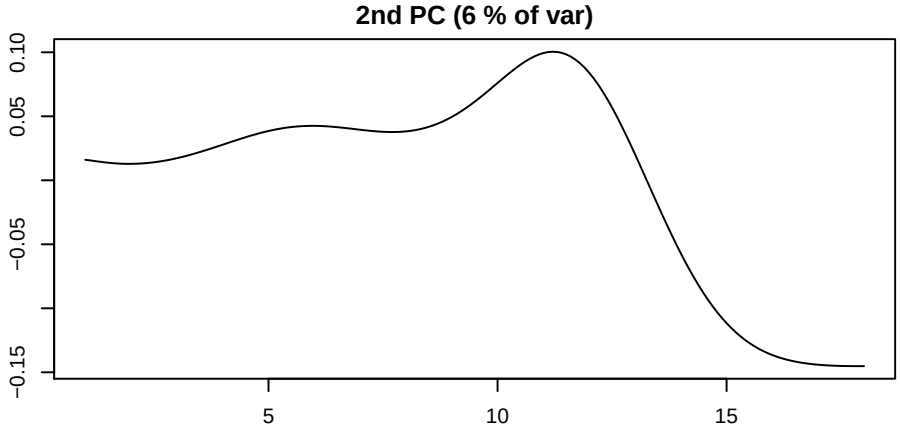


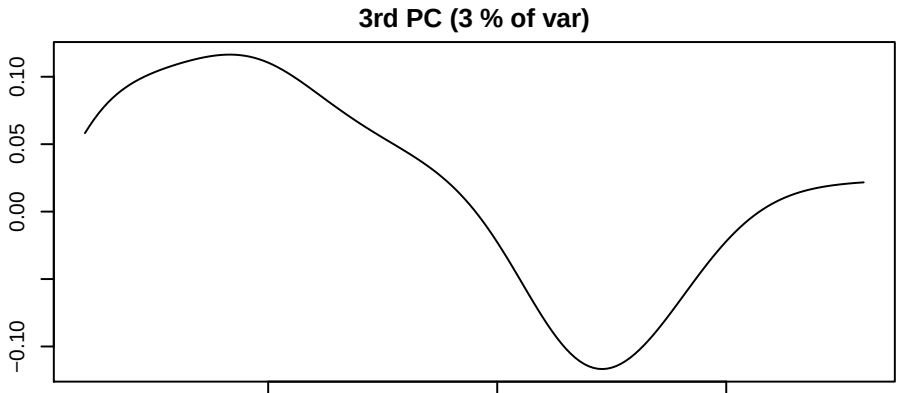
PCA of Height (Girls)

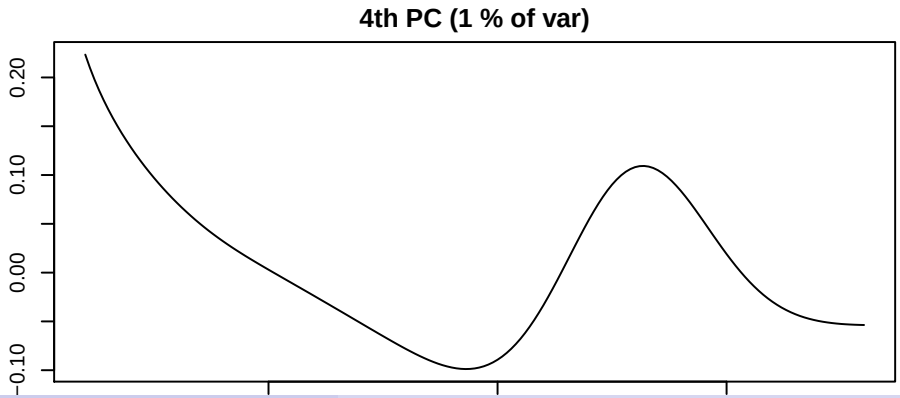




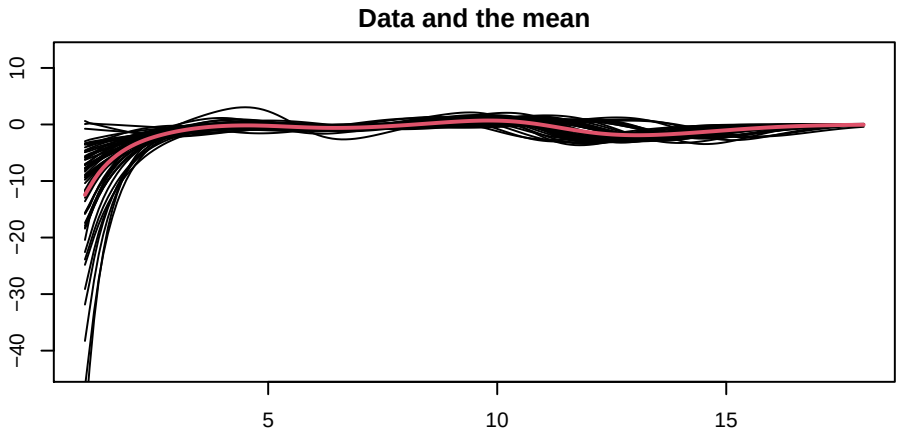


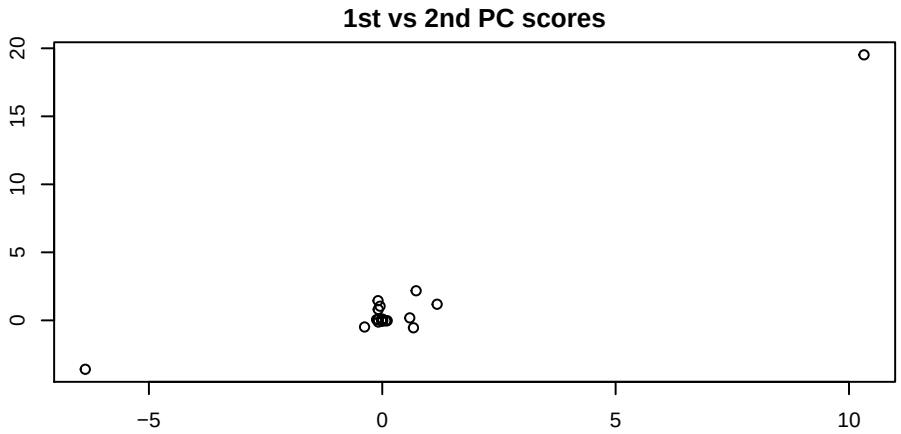


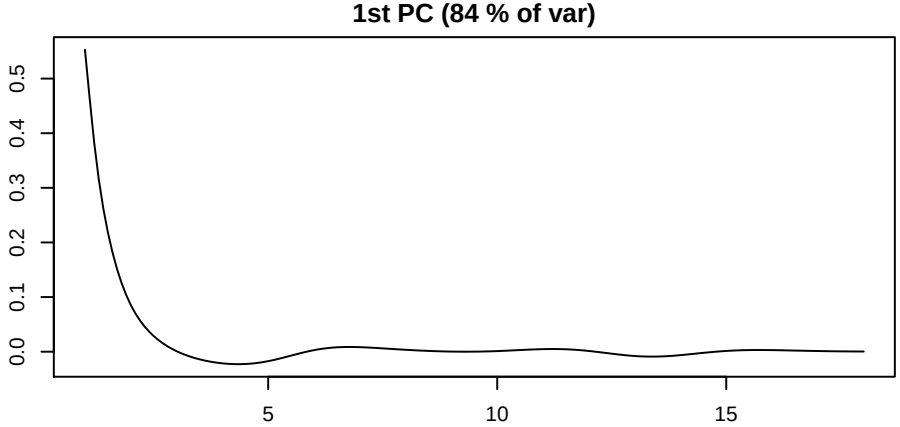


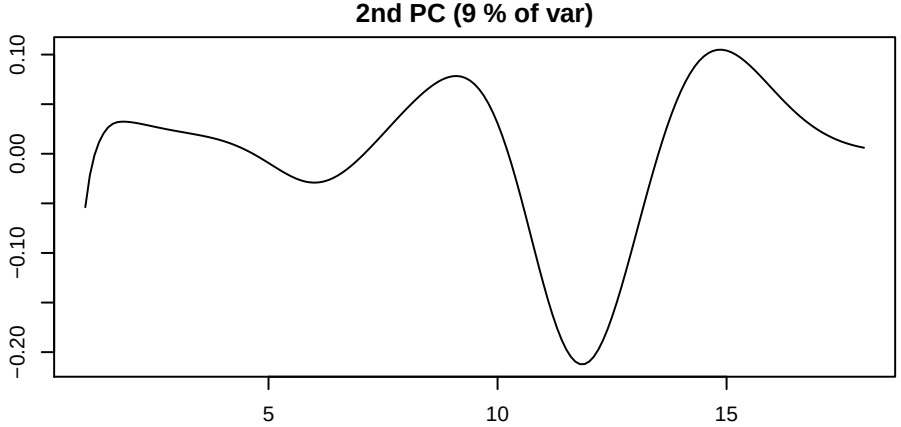


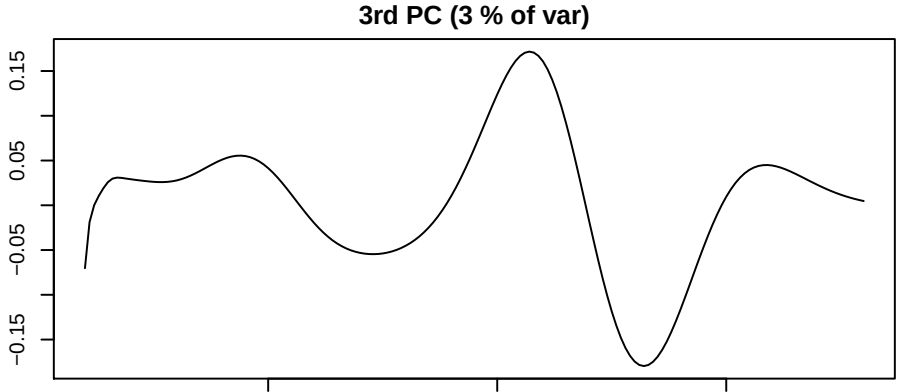
PCA of Acceleration (Girls, 2nd Derivative of Height)

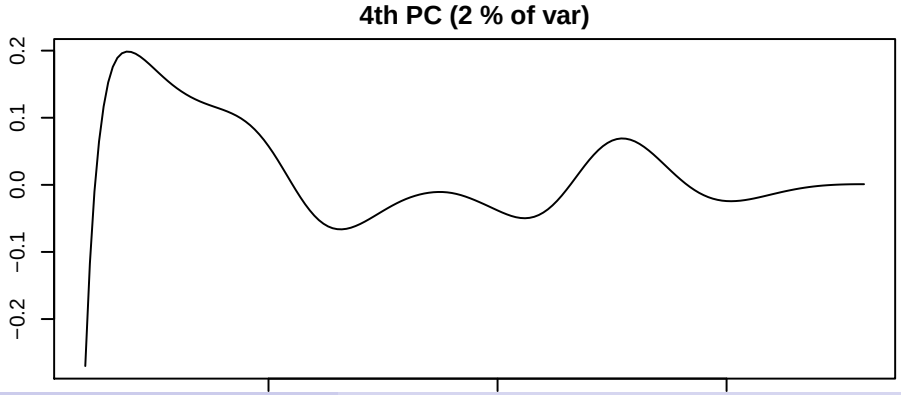








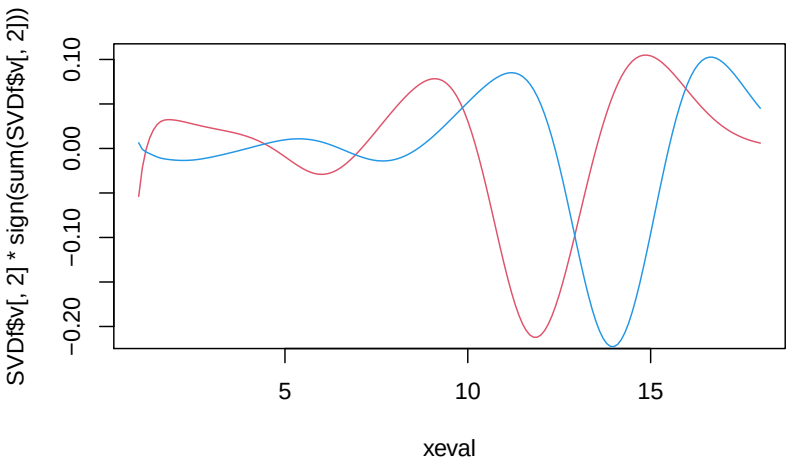


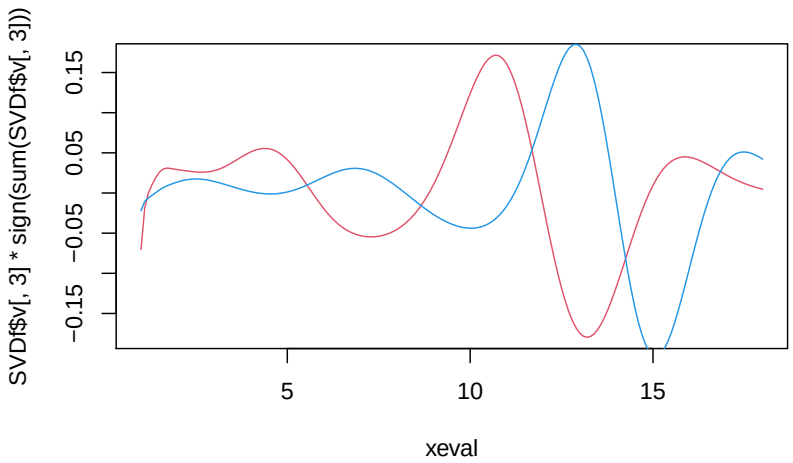


PCA of Acceleration (Girls, 2nd Derivative of Height)

- here things are again similar for girls and boys
- 1st PC captures the variability in the infant age (remember, these are acceleration curves)
- 2nd PC contrasts PGS period against pre- and after-PGS period
- 3rd PC is very hard to interpret due to the sign change in the middle of the PGS period

PGS can be seen clearly on 2nd and 3rd PC:



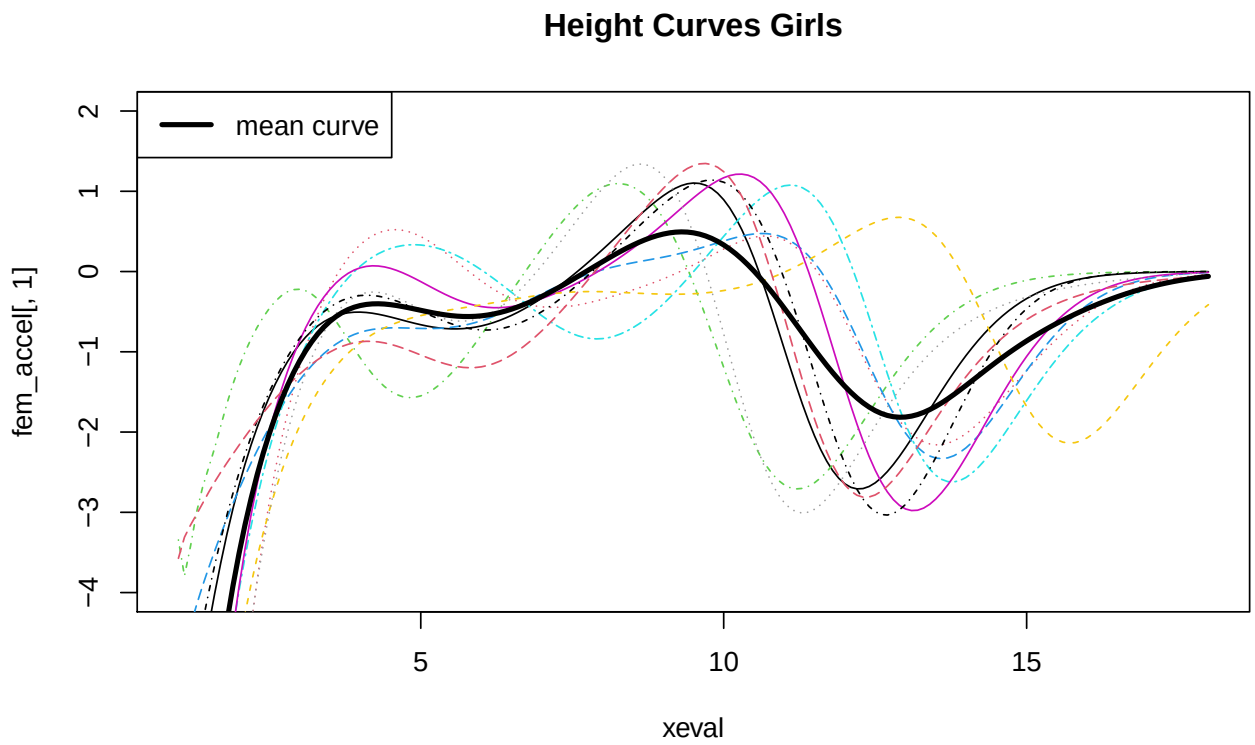


Section 3

Time Warping and Registration

Warping Problem: Example

Let's zoom in on the first 10 female acceleration curves:



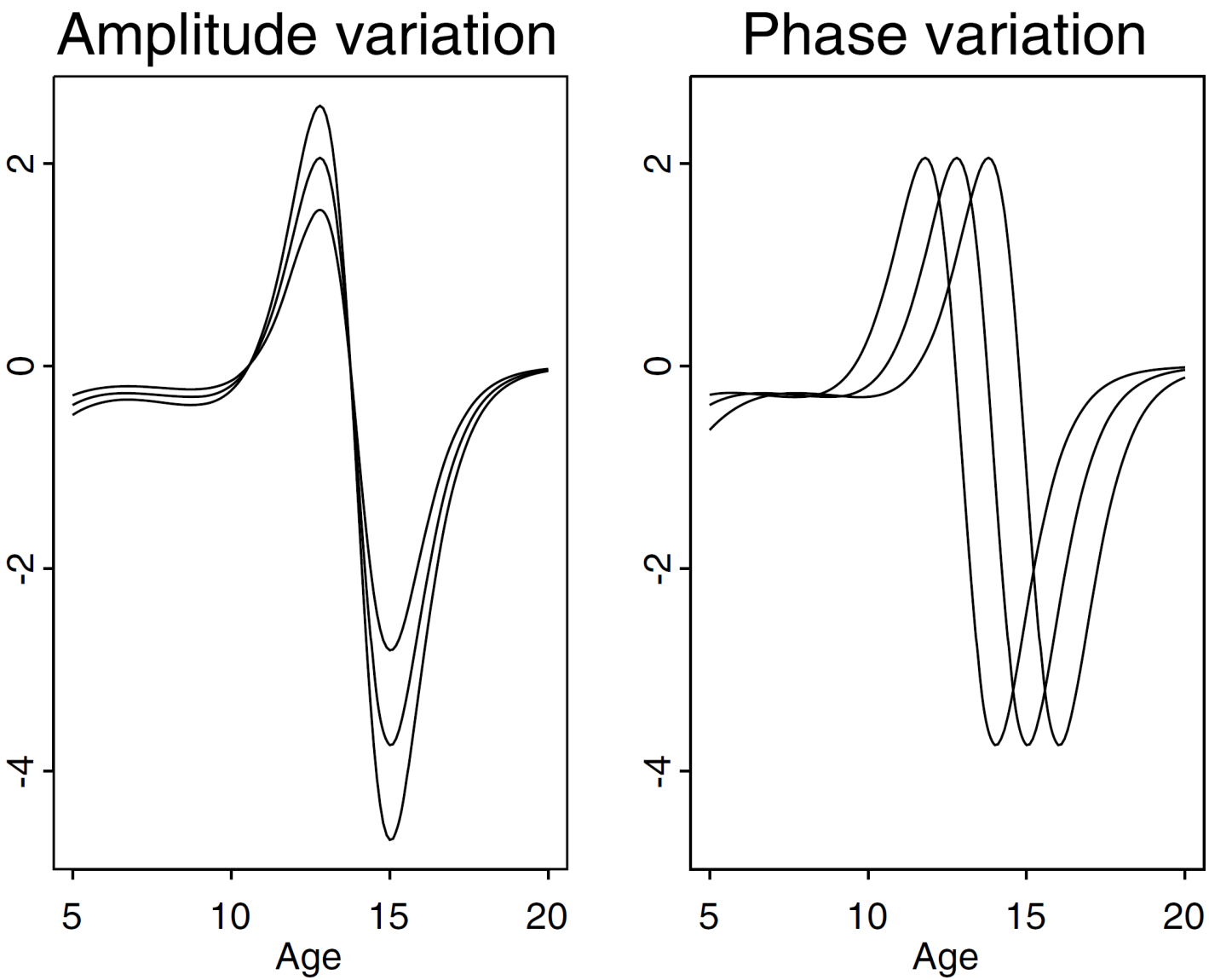
- the mean suggests much longer PGS duration than any single curve
- the peaks and valleys are much smaller than for any single curve

Warping Problem: Example

This is due to PGS occuring at different times for different individuals:

- **growth age** (age w.r.t to the growth process, unique for every individual depending e.g. on secretion of hormones) is a warped version of the
- **objective age** (age in which we are taking measurements – given by the objective time flow)
  - for example, the peak of the PGS (i.e. where the acceleration curves go down to zero in the 10-15 age window) is a well-defined landmark in the growth process but two individuals experience it in a different **objective age**, though their **growth ages** are the same at that landmark

Phase (x) vs. Amplitude (y) Variation



Registration

- re-map the observation interval to $[0, 1]$ just for the sake of presentation
- consider the following model for the observed curves $A_n^{(obs)}(t)$, $n = 1, \ldots, N$:

$$A_n^{(obs)}(t) = A_n^{\star}(F_n(t))$$

where $F_n : [0, 1] \to [0, 1]$ is a non-decreasing *time-warping* function and $A_n^{\star}$ is a realization of a growth-acceleration process $A^{\star}$ that we actually wish to study

- if we knew the functions F_n, we would prefer to work with the registered data

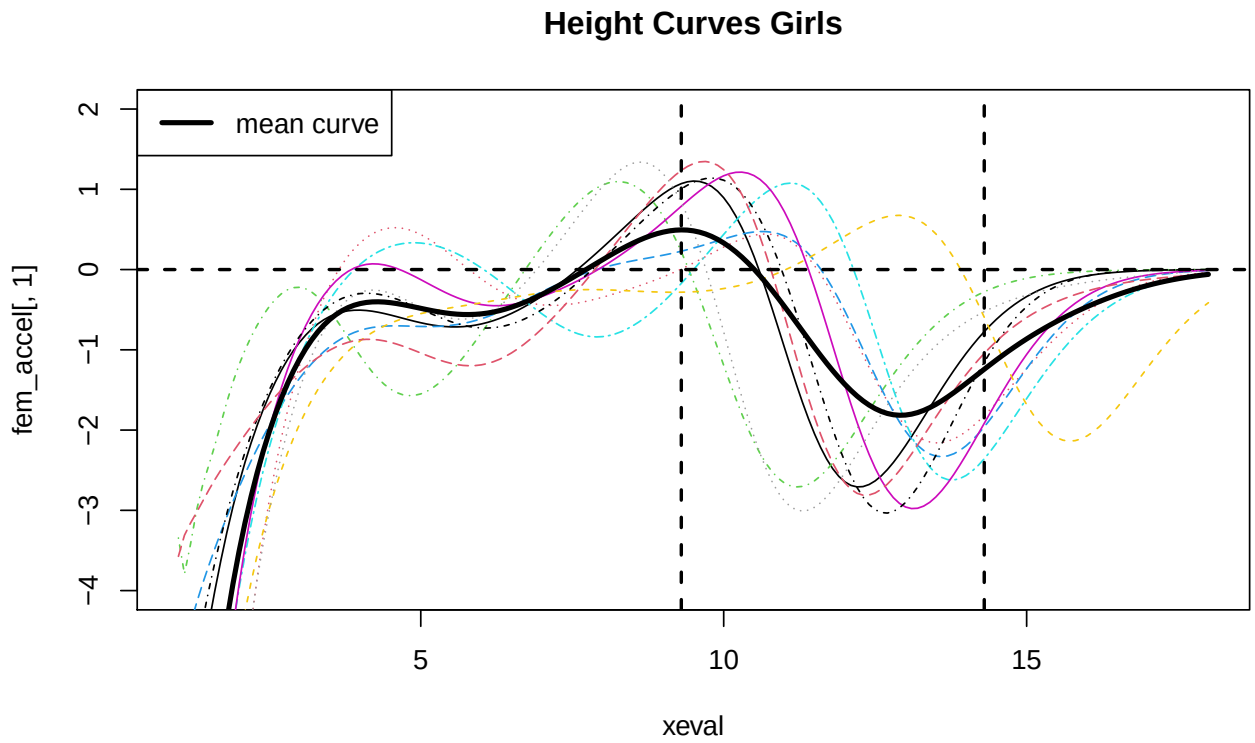
$$A_n^{(reg)}(t) = A_n^{(obs)}(F_n^{-1}(t))$$

Registration

- we want to register the curves, i.e. create a new time flow that is objective w.r.t. the growth process and find the functions F_n
- then we want to **register** every curve in a way such that the growth process follows the objective time
- there are issues with this:
  1. any procedure (and there are many that look mathematically sound) inferring this automatically is doomed to fail unless registered processes are rank one
     - obviously not true here, since this would mean e.g. that someone who is born larger will be taller in the adulthood as well
  2. we need to observe the whole process from the beginning until the end
     - also not true here, but approximately...
- so let us just register the data in a way that our landmark (peak PGS) is registered to a fixed point (e.g. mean PGS)

Localizing Landmarks: Example

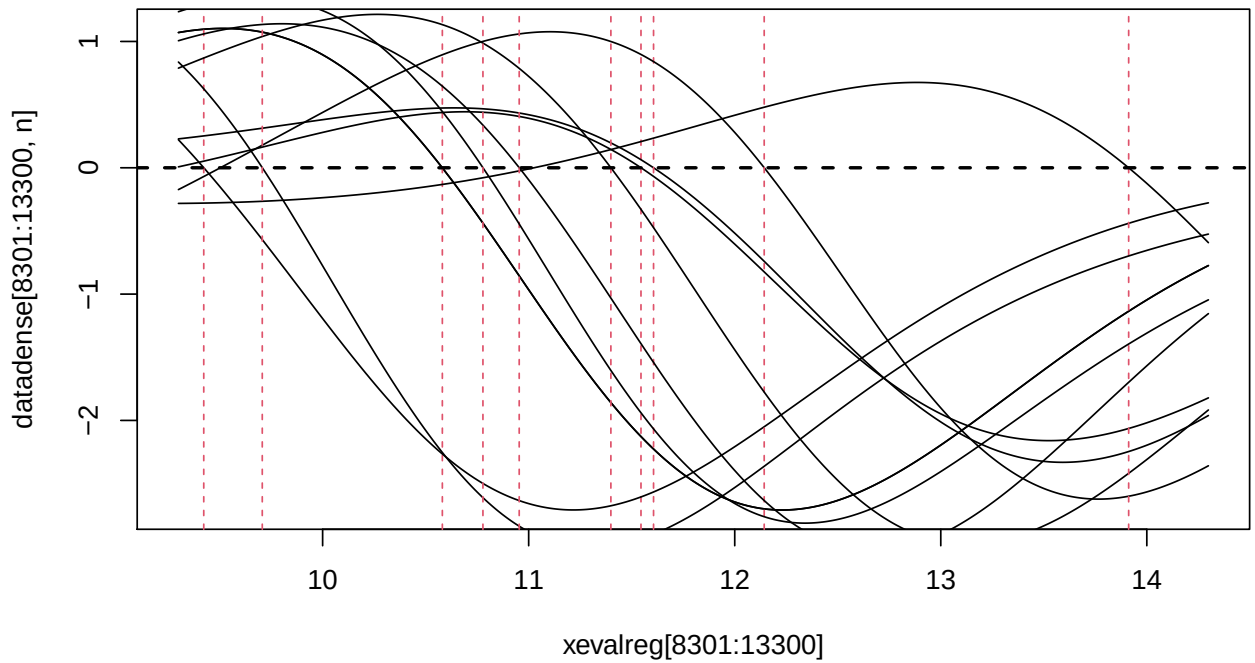
Let's zoom in again:



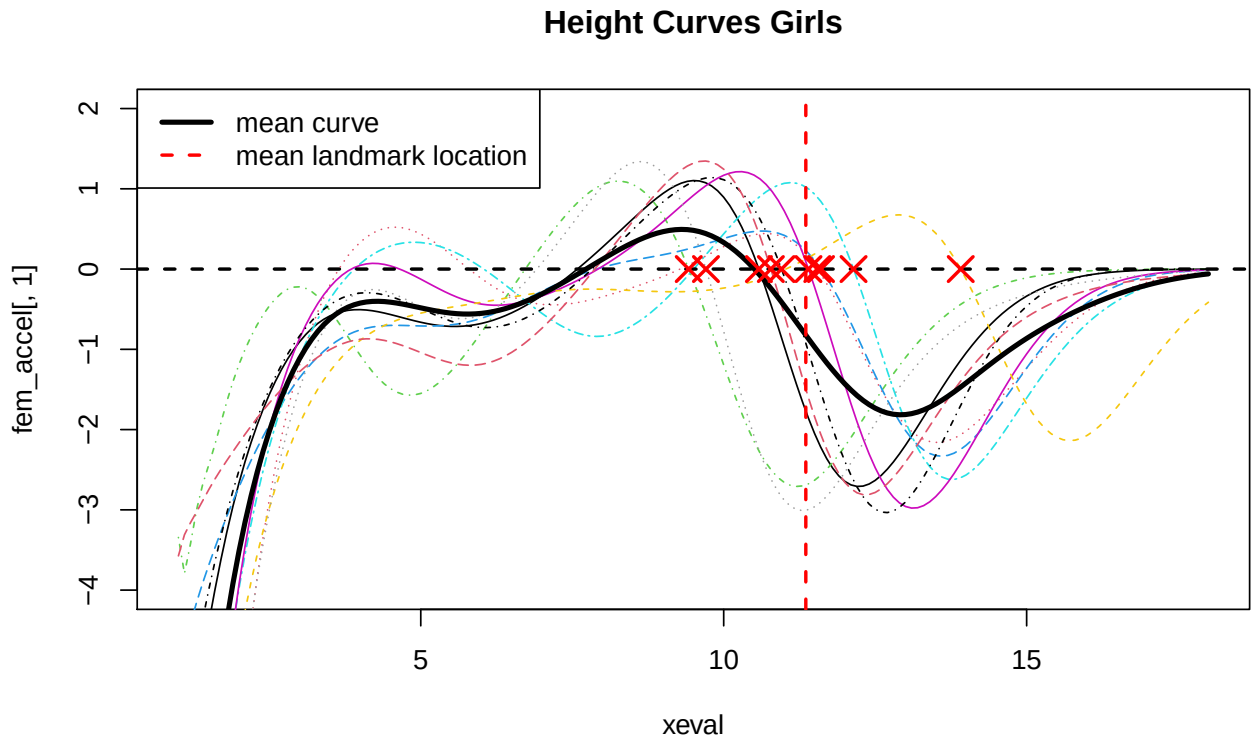
Finding the locations at which acceleration curves go through from above around puberty is a straightforward programming exercise.

Localizing Landmarks: Example

Visual check that the programming exercise was successful:



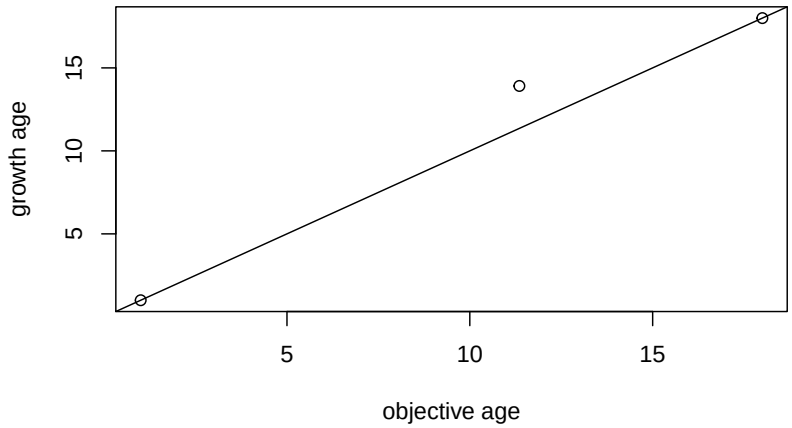
Registering the Landmarks

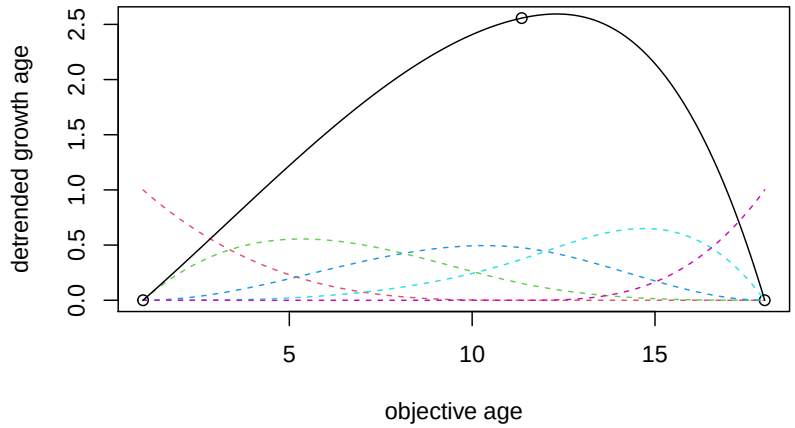


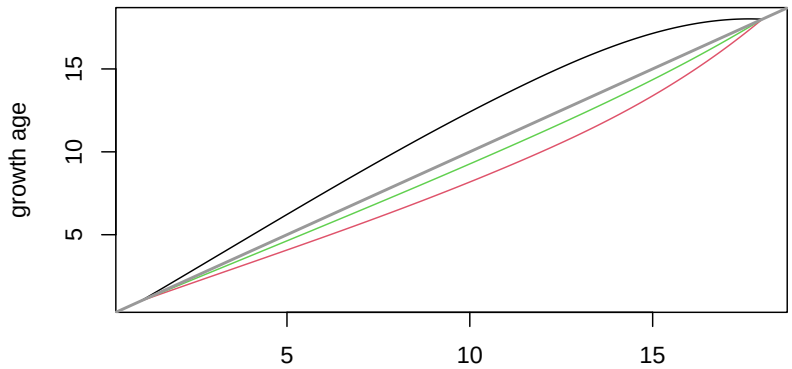
- we want to re-define time-flow for every curve such that
  - all ladmarks (red crosses) align at the mean landmark (the mean location of the PGS peak)
  - we distort time smoothly and monotonically
  - beginning and end remain the same

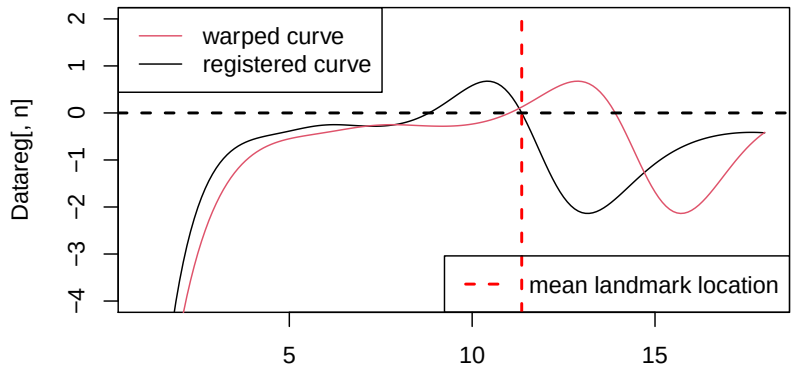
Registering the Landmarks

For illustration, one of the curves has the landmark at 13.9, and this should be registered to the mean landmark at 11.3:



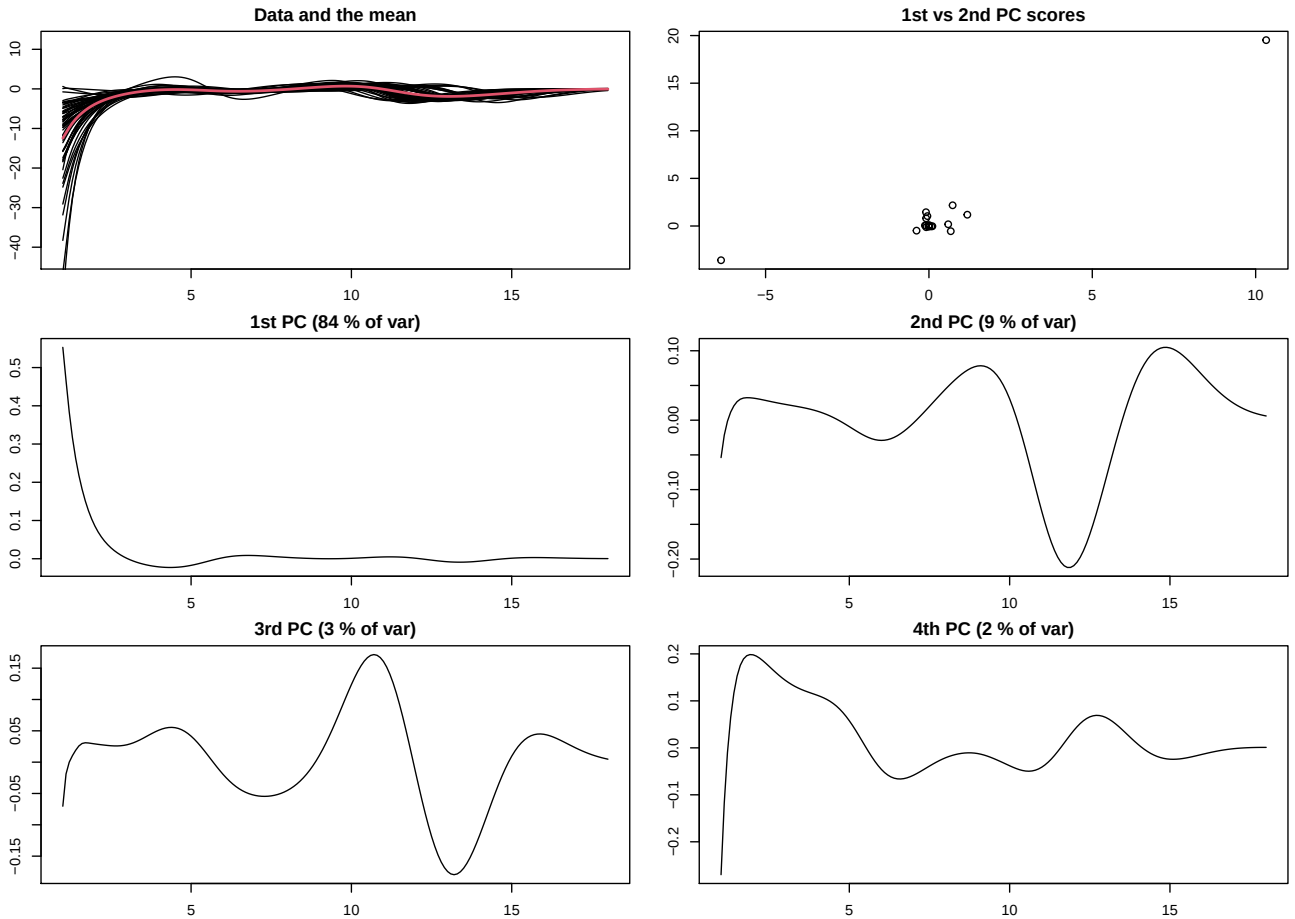






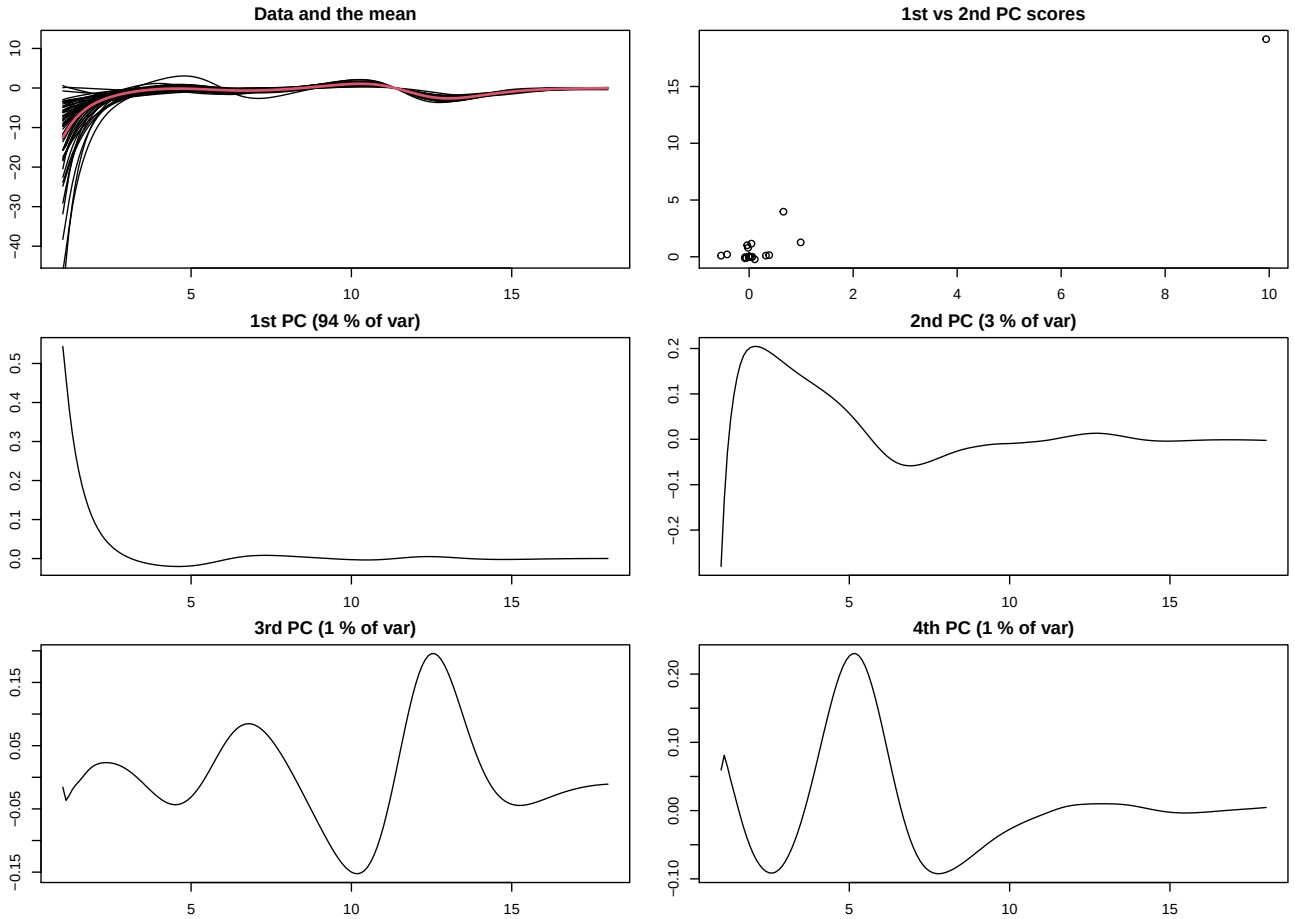
PCA (unregistered)

In the unregistered PCA the 3rd PC is a bit strange, since it captures approximately the same thing as the 2nd one, just a bit shifted.



PCA (registered)

After registration, this PC completely vanishes. Also, registration increased FVE (regardless of how many components we keep). On the other hand most of the variability now at the beginning, which would also need some registration...



Section 4

Project 6

Data

Choose your data set as either

1. covid cases,
2. hospitalizations, or
3. deaths

per capita, and for either the case of

a. the US states, or
b. European states.

Find data on the web, and perform the following tasks. . .

Tasks

- download and check the data
  - think about potential issues
- work with the logarithm of cumulative curves instead of the original daily data
- you will probably need to perform some sort of smoothing
- what is the underlying process you try to study?
  - where does it study and where does it end?
  - after next week, registration might come handy (but not necessarily)
- perform PCA using only `svd()`
- try to interpret your PCs and the low-dimensional plot
  - you might utilize additional info such as GDP for European states or Democratic support for the US states