Laseri fat trajectories

Jouni Tuomisto

2021-03-31

## Overview and tasks

This project aims to create individual trajectories of fat mass change in individuals in the Laseri study.

### Tasks already performed

1. Create a Bayesian model that predicts the mean trajectory for girls and boys.
2. First, use simple polynomial fit with three parameters ax^2 + bx + c.
3. Use multivariable normal for parameters a, b, c.
4. Use PFDWB for 1980-1992 and rasvapros01\_lasklog for 2001.
5. R package sme checked but no longer available at CRAN, so other solutions are needed.

### To be done:

1. Decide on the function for fat percentage change.
2. Decide on the function for individual fat trajectory.
3. Consider using splines for modelling individual changes in fat percentage. Noora’s master’s thesis about smoothing splines for non-linear responses may be of help. Possibly useful articles <https://doi.org/10.1214/ss/1038425655> <https://doi.org/10.1198/1061860043010>

library(reshape2)  
library(tidyverse)

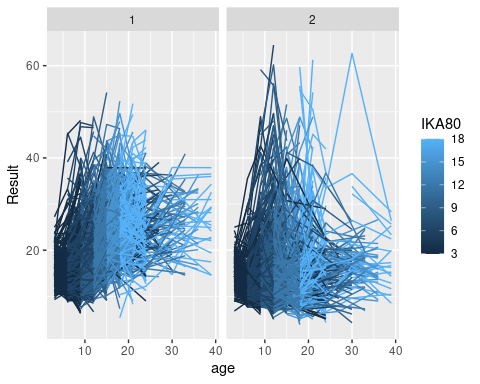
## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.0 ──

## ✓ ggplot2 3.3.2 ✓ purrr 0.3.4  
## ✓ tibble 3.0.4 ✓ dplyr 1.0.2  
## ✓ tidyr 1.1.2 ✓ stringr 1.4.0  
## ✓ readr 1.4.0 ✓ forcats 0.5.0

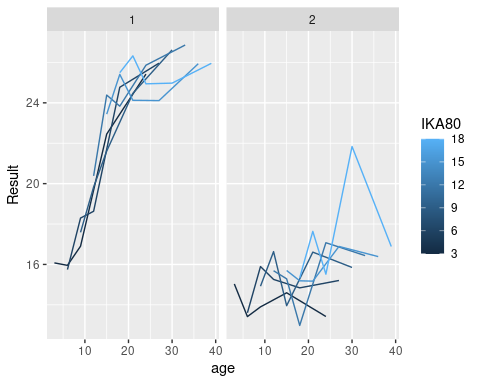
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

df <- read.csv("data/priv/rasvapopperus.csv")[-(11:85)] # Remove POP variables  
# Each row is an individual  
dfl <- melt(df, id.vars = c("tutkno80","SP","IKA80"), variable.name = "Parameter", value.name = "Result")  
dfl$year[grepl("80",dfl$Parameter)] <- 1980  
dfl$year[grepl("83",dfl$Parameter)] <- 1983  
dfl$year[grepl("86",dfl$Parameter)] <- 1986  
dfl$year[grepl("92",dfl$Parameter)] <- 1992  
dfl$year[grepl("01",dfl$Parameter)] <- 2001  
dfl$Parameter <- gsub("(80|83|86|92|01)","",dfl$Parameter)   
dfl$age <- dfl$year - 1980 + dfl$IKA80  
  
ggplot(dfl[dfl$Parameter %in% c("PFDWB","rasvapros\_lasklog"),], aes(x=age, y=Result, colour=IKA80, group=tutkno80))+geom\_line()+facet\_wrap(~SP)

## Warning: Removed 3599 row(s) containing missing values (geom\_path).



tmp <- dfl[dfl$Parameter %in% c("PFDWB","rasvapros\_lasklog"),]  
tmp <- aggregate(tmp["Result"], by = tmp[c("age","SP","IKA80")], FUN=function(x) mean(x, na.rm = TRUE))  
ggplot(tmp, aes(x=age, y=Result, colour=IKA80, group=IKA80))+geom\_line()+facet\_wrap(~SP)



## Hierarchical Bayes

library(rjags) # JAGS

## Loading required package: coda

## Linked to JAGS 4.3.0

## Loaded modules: basemod,bugs

library(MASS) # mvrnorm

##   
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':  
##   
## select

library(car) # scatterplotMatrix

## Loading required package: carData

##   
## Attaching package: 'car'

## The following object is masked from 'package:dplyr':  
##   
## recode

## The following object is masked from 'package:purrr':  
##   
## some

library(mcmcr) # as.mcarray

## Registered S3 method overwritten by 'mcmcr':  
## method from   
## as.mcmc.list.mcarray rjags

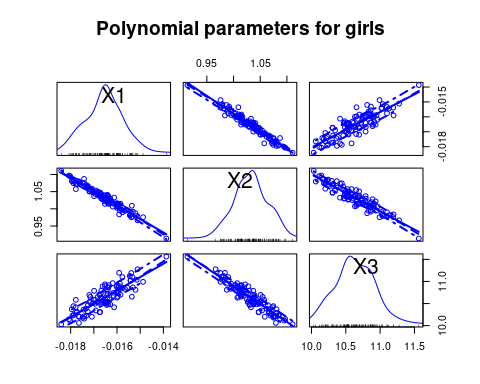
N <- 100 # params$N # defined as render parameter  
  
################ Hierarchical Bayes model.  
## The model estimates individual fat percentage trajectories in the LASERI cohort.  
## The assumption is that each trajectory follows a second-order polynomial, i.e. mu = a1\*x^2 + a2\*x + a3.  
## Parameters a1, a2, and a3 are individually adjusted. The actual observed fat percentage is assumed  
## to be normally distributed with mean mu and precision tau, which is sex-dependent.  
  
# Parameters in the model:  
# a1, a2, a3 = parameters of the polynomial function; sex-dependent  
# fat = observed individual fat percentages for five years (based on e.g. skin fold measurements)  
# s = gender of individual  
# age = age of individual at year 1980  
# mu = individually estimated mean (expected value) of fat percentage; depends also on year  
# taum = precision for mu  
# tauf = precision for the actual fat percentage measurement; depends on gender  
# pred = model predictions of individual trajectories; it is not used in estimation  
  
mod <- textConnection(  
 "  
 var a[2,3], fat[I,5], s[I], age[I];   
 model{  
 taum ~ dgamma(1.0,1.0) # tau for mu  
 tauf[1] ~ dgamma(1.0,1.0) # tau for fat percentage in girls  
 tauf[2] ~ dgamma(1.0,1.0) # tau for fat percentage in boys  
 a[1,1:3] ~ dmnorm(c(0,0,0), Omega) # girls  
 a[2,1:3] ~ dmnorm(c(0,0,0), Omega) # boys  
  
 for(i in 1:I) {   
 for(y in 1:length(YEAR)) {  
 mu[i,y] ~ dnorm(a[s[i],1]\*(YEAR[y]+age[i])^2 + a[s[i],2]\*(YEAR[y]+age[i]) + a[s[i],3], taum)  
 fat[i,y] ~ dnorm(mu[i,y] , tauf[s[i]])  
 pred[i,y] ~ dnorm(mu[i,y] , tauf[s[i]])  
 }  
 }  
 }  
")  
  
jags <- jags.model(  
 mod,  
 data = list(  
 YEAR = c(0,3,6,12,21), # Year since 1980  
 age = df$IKA80, # Age at 1980  
 fat = df[c("PFDWB80","PFDWB83","PFDWB86","PFDWB92","rasvapros01\_lasklog")], # Fat percentage  
 s = df$SP,  
 I = nrow(df),  
 Omega = diag(3)/1000  
 ),  
 n.chains = 4,  
 n.adapt = 1000  
)

## Compiling model graph  
## Declaring variables  
## Resolving undeclared variables  
## Allocating nodes  
## Graph information:  
## Observed stochastic nodes: 11483  
## Unobserved stochastic nodes: 42462  
## Total graph size: 61284  
##   
## Initializing model

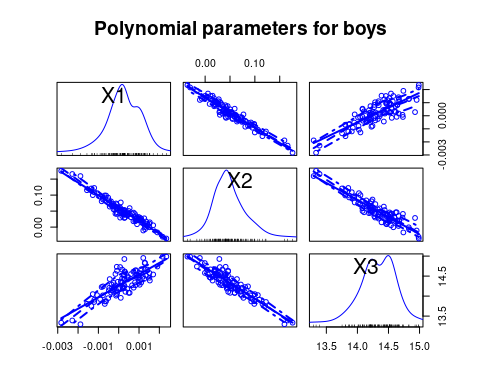
update(jags, 1000)  
  
samps.c <- coda.samples(  
 jags,   
 variable.names=c("a","tau"),  
 n.iter=N\*10,  
 thin=10  
)

## Warning in FUN(X[[i]], ...): Failed to set trace monitor for tau  
## Variable tau not found

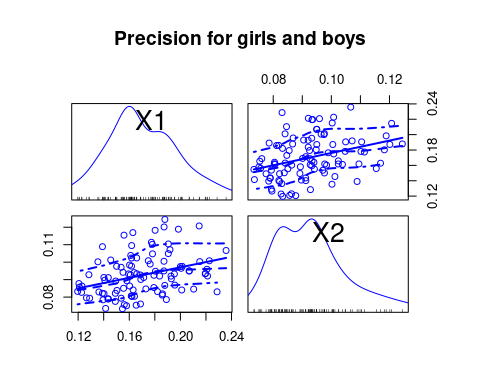
samps.j <- jags.samples(  
 jags,   
 variable.names=c("a","taum", "tauf", "mu","pred"),  
 n.iter=N\*10,  
 thin=10  
)  
  
scatterplotMatrix(t(samps.j$a[1,,,1]), main="Polynomial parameters for girls")



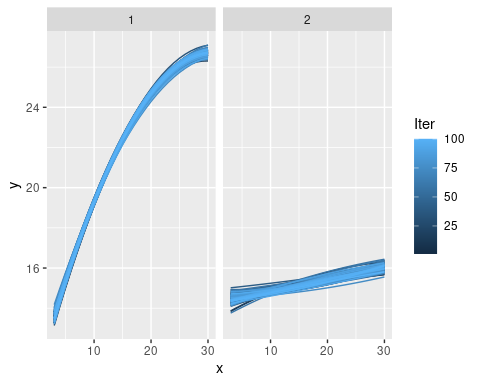
scatterplotMatrix(t(samps.j$a[2,,,1]), main="Polynomial parameters for boys")



scatterplotMatrix(t(samps.j$tauf[,,1]), main="Precision for girls and boys")

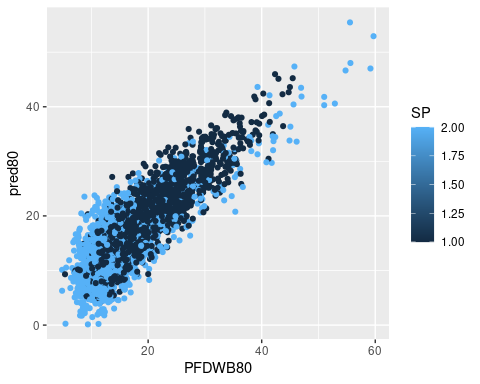


# plot(t(samps.c)) # This works for desktop but not when knitting. Why?  
  
age <- 3:30  
out <- data.frame()  
for(i in 1:100) {  
 for(j in 1:2) {  
 out <- rbind(out, data.frame(  
 Iter = i,  
 sex = j,  
 x = age,  
 y = samps.j$a[j,1,i,1]\*age^2 + samps.j$a[j,2,i,1]\*age + samps.j$a[j,3,i,1]  
 ))  
 }  
}  
  
ggplot(out, aes(x=x,y=y,colour=Iter,group=Iter))+geom\_line()+facet\_wrap(~sex)



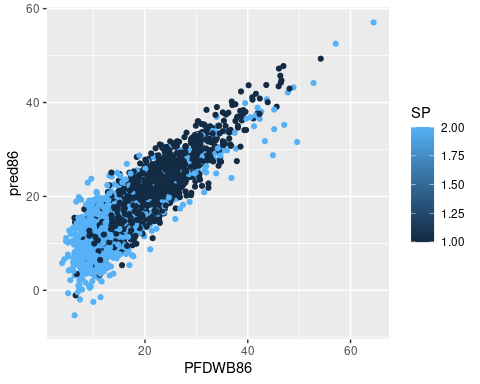
tmp <- apply(samps.j$pred[,,1,1], MARGIN=1:2, FUN=mean)  
colnames(tmp) <- c("pred80","pred83","pred86","pred92","pred01")  
tmp <- data.frame(df,tmp)  
  
ggplot(tmp, aes(x=PFDWB80, y=pred80, colour=SP))+geom\_point()

## Warning: Removed 58 rows containing missing values (geom\_point).



ggplot(tmp, aes(x=PFDWB86, y=pred86, colour=SP))+geom\_point()

## Warning: Removed 1105 rows containing missing values (geom\_point).



Note that the echo = FALSE parameter was added to the code chunk to prevent printing of the R code that generated the plot.