LASERI-trajectories

Jouni Tuomisto

2020-09-30

## R Markdown

This analysis produces a model for trajectories of human body burdens of persistent organic pollutants (POPs).

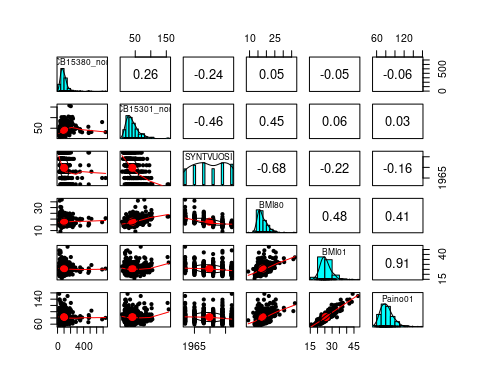
library(foreign)  
library(psych)  
  
df <- read.spss("~/Documents/priv/VS \_Henkilökohtaiset\_POP-trajektorit\_LASERI-aineistossa/updated\_korjattu\_LASERI\_data\_2020\_05\_WORKCOPY.sav", to.data.frame = TRUE)

## Warning in read.spss("~/Documents/priv/VS  
## \_Henkilökohtaiset\_POP-trajektorit\_LASERI-aineistossa/  
## updated\_korjattu\_LASERI\_data\_2020\_05\_WORKCOPY.sav", : ~/Documents/  
## priv/VS \_Henkilökohtaiset\_POP-trajektorit\_LASERI-aineistossa/  
## updated\_korjattu\_LASERI\_data\_2020\_05\_WORKCOPY.sav: Compression bias (0) is not  
## the usual value of 100

## Warning in read.spss("~/Documents/priv/VS  
## \_Henkilökohtaiset\_POP-trajektorit\_LASERI-aineistossa/  
## updated\_korjattu\_LASERI\_data\_2020\_05\_WORKCOPY.sav", : Undeclared level(s) 0  
## added in variable: bintup80

## Warning in read.spss("~/Documents/priv/VS  
## \_Henkilökohtaiset\_POP-trajektorit\_LASERI-aineistossa/  
## updated\_korjattu\_LASERI\_data\_2020\_05\_WORKCOPY.sav", : Undeclared level(s) 0  
## added in variable: bintup01

#colnames(df)[grep("PI",toupper(colnames(df)))]  
  
units <- read.csv("POP\_units.csv")  
  
# Take all-male panel of interesting variables  
df <- df[df$SP=="Poika", c("PCB15380\_norm","PCB15301\_norm","SYNTVUOSI", "BMI80","BMI01","Paino01")] # ,"bodyfat80","bodyfat92"  
df <- na.omit(df)  
# FAKE DATA FOR MODEL CHECKING  
#df <- data.frame(SYNTVUOSI=1962:1966,BMI80=24,BMI01=26,Paino01=c(90,80,70,60,50))  
pairs.panels(df)



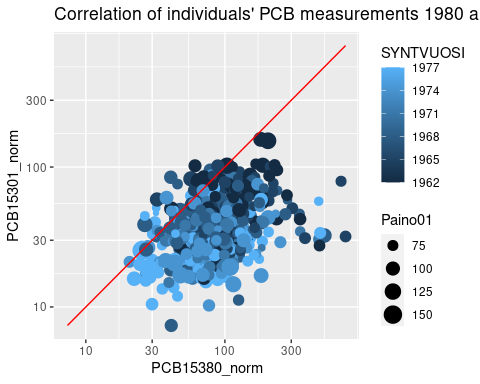
## Indices used  
  
AGE <- 0:40  
H <- length(AGE)  
YEAR <- 1960:2020  
INDIVIDUAL <- 1:nrow(df)  
I <- length(INDIVIDUAL)  
SYNT <- df$SYNTVUOSI[1:I]  
U <- 365 # unit conversion factor: 1/d --> 1/a  
A <- 0.9 # fraction, absorption fraction  
k <- 0.22 # 1/a, elimination constant of the POP  
a <- 1.6 # ug/kg/d, average intake at the peak exposure  
N <- 100  
b <- 0.07 # 1/a, rate of decrease of the POP in the environment and intake  
c <- 1900 # year of peak exposure  
P <- rep(1,length(AGE)) # age-specific intake relative to lifetime average  
  
# https://www.kasvuseula.fi/  
W <- t(matrix(rep(c(5,10,13,15,17,20,22,26,28,31,35,38,44,48,55,60,65,68,70,71,72,  
 rep(75,10),rep(80,10)),length(INDIVIDUAL)),  
 nrow=length(AGE), dimnames=list(Age=AGE, Individual = INDIVIDUAL)))  
  
# A simplistic model uses the population average as default until the weight reaches the  
# observed value of that individual. Then, actual value is used to replace 80 kg (the max default),  
# if larger. This leads to situation where people suddenly gain weight when at 31 or at 2001.  
W <- pmin(W, df$Paino01, na.rm = TRUE)  
W <- ifelse(W==80, pmax(W, df$Paino01, na.rm=TRUE), W)  
tmp <- df$SYNTVUOSI + matrix(rep(AGE,each=length(INDIVIDUAL)),nrow=length(INDIVIDUAL))>=2001  
W <- ifelse(!is.na(df$Paino01) & tmp, df$Paino01, W)  
  
# A simplistic model that assumes that average BMI is equal to lifetime fat percentage.  
M <- W \* ifelse(is.na(df$BMI01), df$BMI80, (df$BMI80 + df$BMI01)/2) / 100  
  
#aggregate(df$Paino01, by=df[c("Age80")], FUN=function(x) mean(x, na.rm=TRUE))

library(ggplot2)

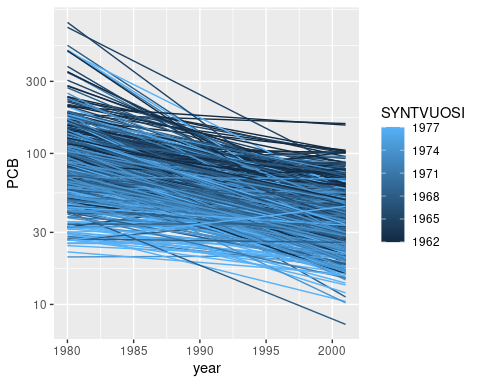
##   
## Attaching package: 'ggplot2'

## The following objects are masked from 'package:psych':  
##   
## %+%, alpha

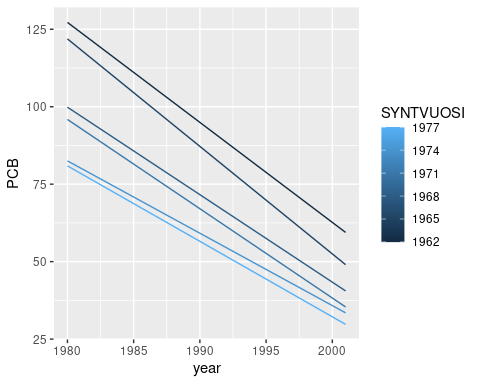
mimax <- c(min(df[c("PCB15380\_norm","PCB15301\_norm")]),max(df[c("PCB15380\_norm","PCB15301\_norm")]))  
ggplot()+  
 geom\_point(data=df, aes(x=PCB15380\_norm, y=PCB15301\_norm, size=Paino01, colour=SYNTVUOSI))+  
 geom\_line(data=data.frame(x=mimax, y=mimax), aes(x=x, y=y),color="red")+  
 scale\_x\_log10()+scale\_y\_log10()+  
 labs(title="Correlation of individuals' PCB measurements 1980 and 2001")



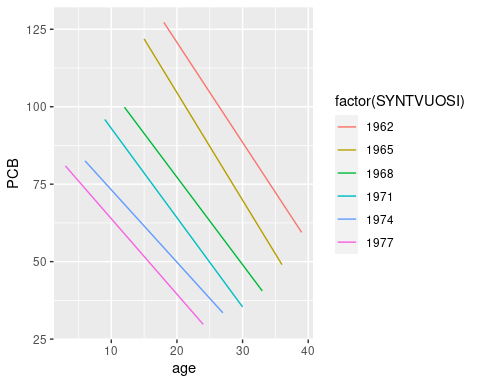
df$id <- 1:nrow(df)  
df$Paino80 <- NA  
long <- reshape(df,  
 idvar=c("SYNTVUOSI","id"),  
 varying=list(  
 PCB=c("PCB15380\_norm","PCB15301\_norm"),  
 BMI= c("BMI80","BMI01"),  
 weight=c("Paino80","Paino01")),  
 v.names =c("PCB","BMI","weight"),  
 timevar="year",  
 times=c(1980,2001),  
 direction="long")  
  
ggplot(long, aes(x=year, y=PCB, colour=SYNTVUOSI, group=id))+geom\_line()+  
 scale\_y\_log10()



ggplot(aggregate(long["PCB"], by=long[c("SYNTVUOSI","year")], FUN=mean),  
 aes(x=year, y=PCB, colour=SYNTVUOSI, group=SYNTVUOSI))+geom\_line()



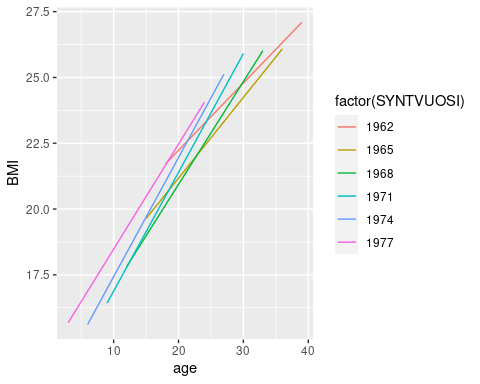
long$age <- long$year - long$SYNTVUOSI  
ggplot(aggregate(long["PCB"], by=long[c("SYNTVUOSI","age")], FUN=mean),  
 aes(x=age, y=PCB, colour=factor(SYNTVUOSI), group=SYNTVUOSI))+geom\_line()



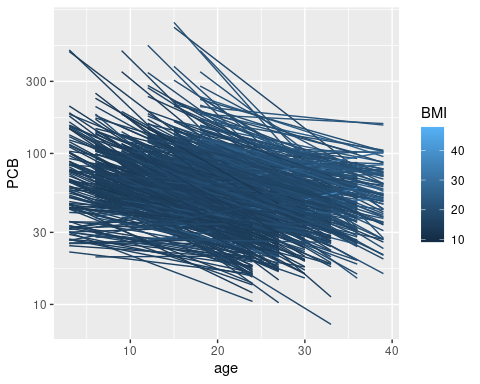
cat("Concentrations in 20-year-old people decrease", signif((1-(40/120)^(1/(1977-1962)))\*100,2), " % per year\n")

## Concentrations in 20-year-old people decrease 7.1 % per year

ggplot(aggregate(long["BMI"], by=long[c("SYNTVUOSI","age")], FUN=mean),  
 aes(x=age, y=BMI, colour=factor(SYNTVUOSI), group=SYNTVUOSI))+geom\_line()



tmp <- aggregate(long["PCB"], by=long[c("age","year")], FUN=mean)  
ggplot(long,  
 aes(x=age, y=PCB, colour=BMI, group=id))+geom\_line()+  
 scale\_y\_log10()

 ## Bayesian hierachical model

# This was forked from Dioxdistboys\_individual\_rows.Rmd and  
# originally from code Op\_en3104/bayes on page [[EU-kalat]]  
  
## This code builds an iterative model with varying intake  
## https://github.com/jtuomist/dioxdisthuman/wiki#iterative-model-with-varying-intake-primary-choice  
  
#if(params$run\_bayes) {  
 library(OpasnetUtils)  
 library(reshape2)  
 library(rjags) # JAGS

## Loading required package: coda

## Linked to JAGS 4.3.0

## Loaded modules: basemod,bugs

library(MASS) # mvrnorm  
 library(car) # scatterplotMatrix

## Loading required package: carData

##   
## Attaching package: 'car'

## The following object is masked from 'package:psych':  
##   
## logit

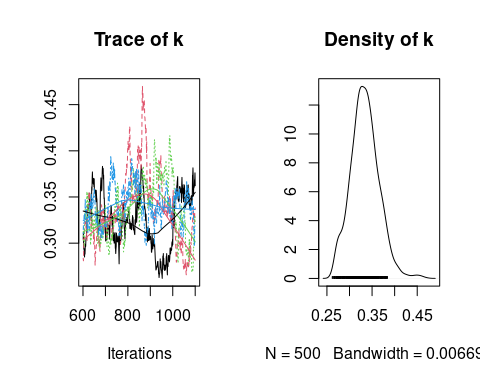
N <- params$N # defined as render parameter  
   
 # Hierarchical Bayes model.  
   
 mod <- textConnection(  
 "  
 var k, a, b, U, E[61], C[I,H], B[I,H], M[I,H], C1, C2, conc[I,2], SYNT[I],  
 B1[I], B2[I], B3[I], B4[I], B5[I], B6[I], B7[I], B8[I], B9[I], B10[I], B11[I],  
 B12[I], B13[I], B14[I], B15[I], B16[I], B17[I], B18[I], B19[I], B20[I], B21[I], B22[I],  
 B23[I], B24[I], B25[I], B26[I], B27[I], B28[I], B29[I], B30[I], B31[I], B32[I], B33[I],  
 B34[I], B35[I], B36[I], B37[I], B38[I], B39[I], B40[I], B41[I];  
 model{  
 # below.LOQ[i,j] ~ dinterval(-conc[i,j], -LOQ[j])  
 k ~ dnorm(0.1, 0.001)  
 a ~ dnorm(2, 0.001)  
# b ~ dnorm(0.1, 0.001)  
 tau[1] ~ dgamma(1.0,1.0)   
 tau[2] ~ dgamma(1.0,1.0)   
 for(year in YEAR) {  
 E[year-1959] <- max(0,a\*b\*exp(1)\*(year-c+1/b)\*exp(-b\*(year-c+1/b)))  
 }  
 B2[] <- B1[] \* (1 - k) + U \* A \* W[,1] \* E[SYNT[]-1960+1] \* P[1]  
 B3[] <- B2[] \* (1 - k) + U \* A \* W[,2] \* E[SYNT[]-1960+2] \* P[2]  
 B4[] <- B3[] \* (1 - k) + U \* A \* W[,3] \* E[SYNT[]-1960+3] \* P[3]  
 B5[] <- B4[] \* (1 - k) + U \* A \* W[,4] \* E[SYNT[]-1960+4] \* P[4]  
 B6[] <- B5[] \* (1 - k) + U \* A \* W[,5] \* E[SYNT[]-1960+5] \* P[5]  
 B7[] <- B6[] \* (1 - k) + U \* A \* W[,6] \* E[SYNT[]-1960+6] \* P[6]  
 B8[] <- B7[] \* (1 - k) + U \* A \* W[,7] \* E[SYNT[]-1960+7] \* P[7]  
 B9[] <- B8[] \* (1 - k) + U \* A \* W[,8] \* E[SYNT[]-1960+8] \* P[8]  
 B10[] <- B9[] \* (1 - k) + U \* A \* W[,9] \* E[SYNT[]-1960+9] \* P[9]  
 B11[] <- B10[] \* (1 - k) + U \* A \* W[,10] \* E[SYNT[]-1960+10] \* P[10]  
 B12[] <- B11[] \* (1 - k) + U \* A \* W[,11] \* E[SYNT[]-1960+11] \* P[11]  
 B13[] <- B12[] \* (1 - k) + U \* A \* W[,12] \* E[SYNT[]-1960+12] \* P[12]  
 B14[] <- B13[] \* (1 - k) + U \* A \* W[,13] \* E[SYNT[]-1960+13] \* P[13]  
 B15[] <- B14[] \* (1 - k) + U \* A \* W[,14] \* E[SYNT[]-1960+14] \* P[14]  
 B16[] <- B15[] \* (1 - k) + U \* A \* W[,15] \* E[SYNT[]-1960+15] \* P[15]  
 B17[] <- B16[] \* (1 - k) + U \* A \* W[,16] \* E[SYNT[]-1960+16] \* P[16]  
 B18[] <- B17[] \* (1 - k) + U \* A \* W[,17] \* E[SYNT[]-1960+17] \* P[17]  
 B19[] <- B18[] \* (1 - k) + U \* A \* W[,18] \* E[SYNT[]-1960+18] \* P[18]  
 B20[] <- B19[] \* (1 - k) + U \* A \* W[,19] \* E[SYNT[]-1960+19] \* P[19]  
 B21[] <- B20[] \* (1 - k) + U \* A \* W[,20] \* E[SYNT[]-1960+20] \* P[20]  
 B22[] <- B21[] \* (1 - k) + U \* A \* W[,21] \* E[SYNT[]-1960+21] \* P[21]  
 B23[] <- B22[] \* (1 - k) + U \* A \* W[,22] \* E[SYNT[]-1960+22] \* P[22]  
 B24[] <- B23[] \* (1 - k) + U \* A \* W[,23] \* E[SYNT[]-1960+23] \* P[23]  
 B25[] <- B24[] \* (1 - k) + U \* A \* W[,24] \* E[SYNT[]-1960+24] \* P[24]  
 B26[] <- B25[] \* (1 - k) + U \* A \* W[,25] \* E[SYNT[]-1960+25] \* P[25]  
 B27[] <- B26[] \* (1 - k) + U \* A \* W[,26] \* E[SYNT[]-1960+26] \* P[26]  
 B28[] <- B27[] \* (1 - k) + U \* A \* W[,27] \* E[SYNT[]-1960+27] \* P[27]  
 B29[] <- B28[] \* (1 - k) + U \* A \* W[,28] \* E[SYNT[]-1960+28] \* P[28]  
 B30[] <- B29[] \* (1 - k) + U \* A \* W[,29] \* E[SYNT[]-1960+29] \* P[29]  
 B31[] <- B30[] \* (1 - k) + U \* A \* W[,30] \* E[SYNT[]-1960+30] \* P[30]  
 B32[] <- B31[] \* (1 - k) + U \* A \* W[,31] \* E[SYNT[]-1960+31] \* P[31]  
 B33[] <- B32[] \* (1 - k) + U \* A \* W[,32] \* E[SYNT[]-1960+32] \* P[32]  
 B34[] <- B33[] \* (1 - k) + U \* A \* W[,33] \* E[SYNT[]-1960+33] \* P[33]  
 B35[] <- B34[] \* (1 - k) + U \* A \* W[,34] \* E[SYNT[]-1960+34] \* P[34]  
 B36[] <- B35[] \* (1 - k) + U \* A \* W[,35] \* E[SYNT[]-1960+35] \* P[35]  
 B37[] <- B36[] \* (1 - k) + U \* A \* W[,36] \* E[SYNT[]-1960+36] \* P[36]  
 B38[] <- B37[] \* (1 - k) + U \* A \* W[,37] \* E[SYNT[]-1960+37] \* P[37]  
 B39[] <- B38[] \* (1 - k) + U \* A \* W[,38] \* E[SYNT[]-1960+38] \* P[38]  
 B40[] <- B39[] \* (1 - k) + U \* A \* W[,39] \* E[SYNT[]-1960+39] \* P[39]  
 B41[] <- B40[] \* (1 - k) + U \* A \* W[,40] \* E[SYNT[]-1960+40] \* P[40]  
   
 C[,1] <- B1 / M[,1]  
 C[,2] <- B2 / M[,2]  
 C[,3] <- B3 / M[,3]  
 C[,4] <- B4 / M[,4]  
 C[,5] <- B5 / M[,5]  
 C[,6] <- B6 / M[,6]  
 C[,7] <- B7 / M[,7]  
 C[,8] <- B8 / M[,8]  
 C[,9] <- B9 / M[,9]  
 C[,10] <- B10 / M[,10]  
 C[,11] <- B11 / M[,11]  
 C[,12] <- B12 / M[,12]  
 C[,13] <- B13 / M[,13]  
 C[,14] <- B14 / M[,14]  
 C[,15] <- B15 / M[,15]  
 C[,16] <- B16 / M[,16]  
 C[,17] <- B17 / M[,17]  
 C[,18] <- B18 / M[,18]  
 C[,19] <- B19 / M[,19]  
 C[,20] <- B20 / M[,20]  
 C[,21] <- B21 / M[,21]  
 C[,22] <- B22 / M[,22]  
 C[,23] <- B23 / M[,23]  
 C[,24] <- B24 / M[,24]  
 C[,25] <- B25 / M[,25]  
 C[,26] <- B26 / M[,26]  
 C[,27] <- B27 / M[,27]  
 C[,28] <- B28 / M[,28]  
 C[,29] <- B29 / M[,29]  
 C[,30] <- B30 / M[,30]  
 C[,31] <- B31 / M[,31]  
 C[,32] <- B32 / M[,32]  
 C[,33] <- B33 / M[,33]  
 C[,34] <- B34 / M[,34]  
 C[,35] <- B35 / M[,35]  
 C[,36] <- B36 / M[,36]  
 C[,37] <- B37 / M[,37]  
 C[,38] <- B38 / M[,38]  
 C[,39] <- B39 / M[,39]  
 C[,40] <- B40 / M[,40]  
 C[,41] <- B41 / M[,41]  
   
 for(i in 1:I) {   
 conc[i,1] ~ dnorm(C[i,1980 - SYNT[i]+1], tau[1]) # Age 0 is at position 1  
 conc[i,2] ~ dnorm(C[i,2001 - SYNT[i]+1], tau[2])  
 pred[i,1] ~ dnorm(C[i,1980 - SYNT[i]+1], tau[1])  
 pred[i,2] ~ dnorm(C[i,2001 - SYNT[i]+1], tau[2])  
 }  
 }  
 ")  
  
 jags <- jags.model(  
 mod,  
 data = list(  
 YEAR = YEAR,  
 W = W[1:I,1:H],  
 M = M[1:I,1:H],  
 H = H,  
 I = I,  
 U = U,  
 A = A,  
 b = b,  
 c = c,  
 P = P[1:H],  
 SYNT = SYNT,  
 conc = df[1:I,c("PCB15380\_norm","PCB15301\_norm")],  
 B1 = rep(0,I)  
 ),  
 n.chains = 4,  
 n.adapt = 100  
 )

## Compiling model graph  
## Declaring variables  
## Resolving undeclared variables  
## Allocating nodes  
## Graph information:  
## Observed stochastic nodes: 1130  
## Unobserved stochastic nodes: 1134  
## Total graph size: 51681  
##   
## Initializing model

samps.j <- jags.samples(  
 jags,   
 c(  
 'k',   
 'a',  
 'b',  
 'tau',  
 'E',  
 'pred'  
 ),   
 thin=1,  
 N\*1  
 )  
  
 #scatterplotMatrix(samps.j$k, main="Elimination constant k")  
   
 cat("Elimination constant k (1/a)\n")

## Elimination constant k (1/a)

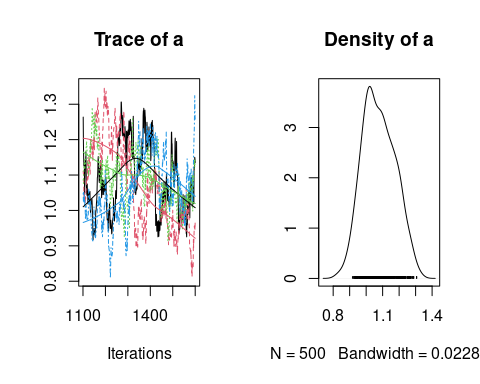
plot(coda.samples(jags, 'k', N))



cat("Average exposure during peak year (ug/kg/d)\n")

## Average exposure during peak year (ug/kg/d)

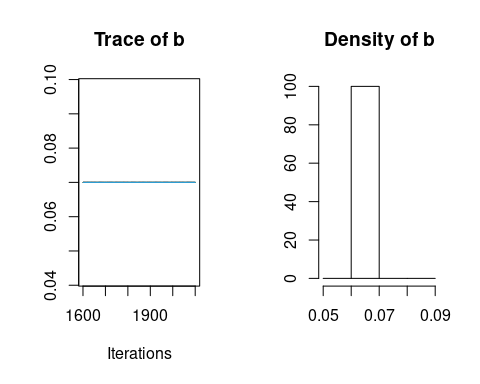
plot(coda.samples(jags, 'a', N))



cat("Exposure reduction rate (1/a)\n")

## Exposure reduction rate (1/a)

plot(coda.samples(jags, 'b', N))



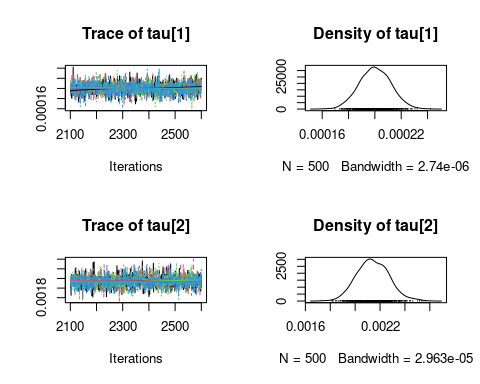
cat("Exposure reduction rate (1/a)\n")

## Exposure reduction rate (1/a)

# plot(coda.samples(jags, 'E', N))  
 cat("Concentration precision\n")

## Concentration precision

plot(coda.samples(jags, 'tau', N))

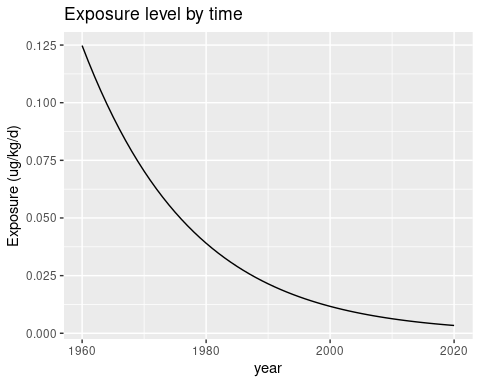


cat("Predicted concentration\n")

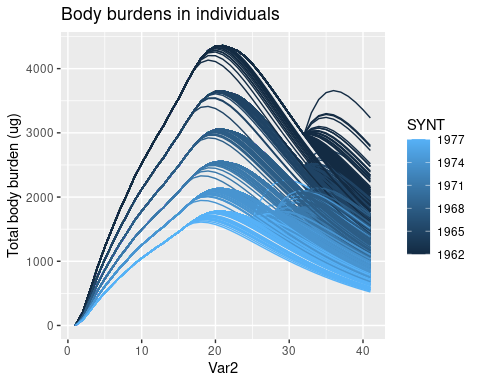
## Predicted concentration

# plot(coda.samples(jags, 'pred', N))  
   
#}

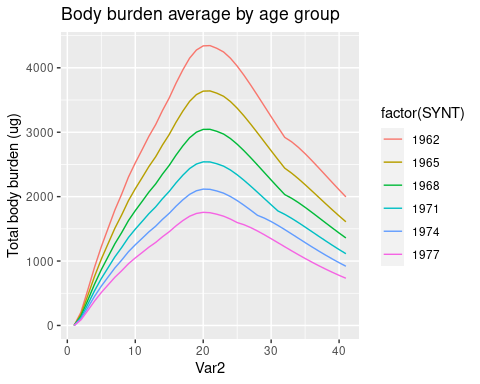
E <- pmax(0,a\*b\*exp(1)\*(YEAR-c+1/b)\*exp(-b\*(YEAR-c+1/b)))  
 B <- matrix(0, nrow=I, ncol=H)  
 for(age in 2:H) {  
 B[,age] <- B[,age-1] \* (1 - k) + U \* A \* W[,age-1] \* E[SYNT-1963+age] \* P[age]  
 }  
 C <- B / M  
   
 ggplot(data.frame(year=YEAR,exposure=E), aes(x=year, y=E))+geom\_line()+  
 labs(title="Exposure level by time",  
 y="Exposure (ug/kg/d)")



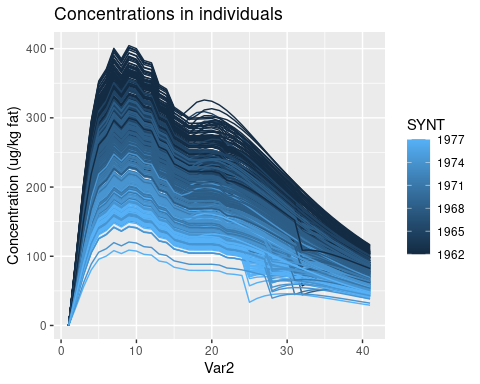
dat <- melt(B)#[seq(1,550, 10),])  
 dat <- merge(dat, cbind(Var1= 1:I, SYNT=SYNT))  
   
 ggplot(dat, aes(x=Var2, y=value, colour=SYNT, group=Var1))+geom\_line()+  
 labs(title="Body burdens in individuals",  
 y="Total body burden (ug)")



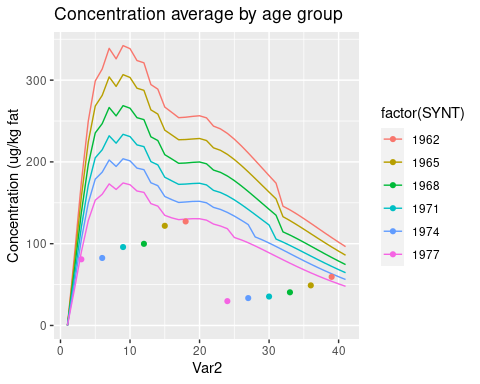
ggplot(aggregate(dat["value"], by=dat[c("Var2","SYNT")], FUN=mean),   
 aes(x=Var2, y=value, colour=factor(SYNT), group=SYNT))+geom\_line()+  
 labs(title="Body burden average by age group",  
 y="Total body burden (ug)")



dat <- melt(C)#[seq(1,550, 10),])  
 dat <- merge(dat, cbind(Var1= 1:I, SYNT=SYNT))  
   
 ggplot(dat, aes(x=Var2, y=value, colour=SYNT, group=Var1))+geom\_line()+  
 labs(title="Concentrations in individuals",  
 y="Concentration (ug/kg fat)")



ggplot()+  
 geom\_line(data=aggregate(dat["value"], by=dat[c("Var2","SYNT")], FUN=mean),   
 aes(x=Var2, y=value, colour=factor(SYNT), group=SYNT))+  
 geom\_point(data=aggregate(long["PCB"],by=long[c("SYNTVUOSI","age")],FUN=mean),  
 aes(x=age, y=PCB,colour=factor(SYNTVUOSI)))+geom\_point()+  
 labs(title="Concentration average by age group",  
 y="Concentration (ug/kg fat")



tst <- rowMeans(samps.j$pred[,,,], dims=2)  
tst <- data.frame(PCB15380\_pred = tst[,1],PCB15301\_pred = tst[,2],  
 df[c("PCB15380\_norm","PCB15301\_norm")])  
pairs.panels(tst)

