TEQ concentration predictions in 9-year-old boys in Finland

Jouni Tuomisto

2020-05-15

## Model initiation

This code initiates the model estimating TEQ values of individual boys with only PCB measurements.

NOTE! This model has a problem that it first logtransforms the data and then makes a regression. The problem is that the sum of logtransformed regression parameters equals the product of original parameters. In the case of TEQ, this should be the sum or original paramters. Therefore, we should actually use the original rathern than logtransformed values and tell the model in another way that the values (PCBs and TEQs) are lognormally distributed. However, I haven’t yet figured out a correct mathematical way to do this.

library(OpasnetUtils)  
library(reshape2)  
library(ggplot2)  
library(MASS)  
library(thlGraphs)

##   
## Attaching package: 'thlGraphs'

## The following object is masked \_by\_ '.GlobalEnv':  
##   
## theme\_thl

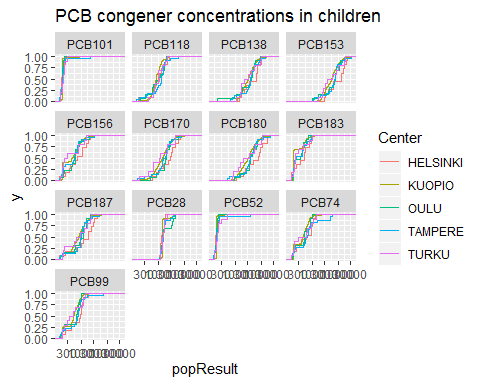
use\_database <- FALSE  
  
if(use\_database) {  
 library(thlConnect)  
 con <- thlDbConnect("pubhealth",dbengine = "postgresql")  
   
 #dbi <- thlDbInfo("pubhealth",dbengine="postgresql",driver="JDBC")  
   
 pops <- thlJdbcQuery(con, "SELECT \* FROM envhealth.dioxdisthuman;")  
 colnames(pops) <- paste0(toupper(substring(colnames(pops),1,1)), substring(colnames(pops),2))  
 colnames(pops)[match(c("Popsresult","Popssource"),colnames(pops))] <- c("popsResult","popsSource")  
 pops <- Ovariable("pops", output=pops, marginal=(colnames(pops)!="popsResult"), unit="pg/g")  
} else {  
 load(file="../Dioxdistboys/pops\_ovariable")  
}  
  
# PCBs of interest (the order of decreasing correlation with SUM-TEQ will be determined automatically later)  
pcb9 <- c("PCB118","PCB138","PCB74","PCB156","PCB153","PCB99","PCB187","PCB170","PCB180")  
  
teq3 <- c("PCDDF\_TEQ", "PCB\_TEQ", "Total\_TEQ")  
  
# "Seven marker PCBs": 28, 52, 101, 118, 138, 153, and 180  
# "Six marker PCBs": 28, 52, 101, 138, 153, and 180  
  
# Why are 28, 52, 101 missing from nine marker PCBs?  
# 52 and 101 were measured but too much <LOQ that they were omitted.  
# 28 was measured, too. Not known why it was not used.  
  
# Adjust for LOQ if not detected. This applies to mothers only as men's <LOQs were replaced already and none in boys  
  
LOQ <- oapply(pops, c("Compound","Year"), function(x) min(x[x>0]))  
colnames(LOQ@output)[colnames(LOQ@output)=="popsResult"] <- "LOQResult"  
LOQ@name <- "LOQ"  
  
# LOQ <- LOQ # Use upper-bound estimate  
# LOQ <- LOQ / 2 # Use medium-bound estimate  
LOQ <- LOQ / 10 # Use lower-bound estimate  
   
pops <- pops + LOQ \* (pops==0)  
  
# Calculate and add TEQ values to the data  
  
objects.latest("Op\_en4017", "initiate") # [[Toxic equivalency factor]] TEF

## Loading objects:  
## TEFraw  
## TEF

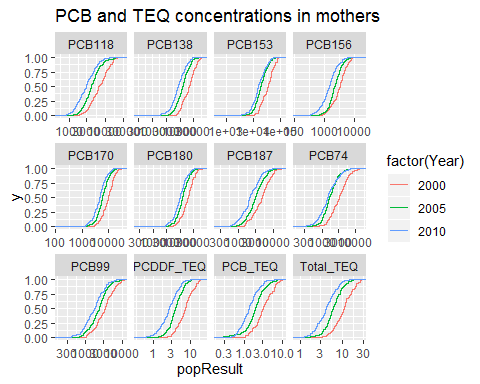
TEF <- EvalOutput(TEF)  
levels(TEF$Compound) <- toupper(levels(TEF$Compound))  
  
popteq <- pops \* TEF  
  
tmp1 <- oapply(popteq[popteq$Group %in% c("Chlorinated dibenzo-p-dioxins", "Chlorinated dibenzofurans")],NULL,sum,c("Compound","Group"))  
tmp2 <- oapply(popteq[popteq$Group %in% c("Non-ortho-substituted PCBs", "Mono-ortho-substituted PCBs")],NULL,sum,c("Compound","Group"))  
tmp3 <- tmp1 + tmp2  
tmp1$Compound <- "PCDDF\_TEQ"  
tmp2$Compound <- "PCB\_TEQ"  
tmp3$Compound <- "Total\_TEQ"  
  
pop <- OpasnetUtils::combine(pops, tmp1, tmp2, tmp3, name="pop")  
pop <- pop[!(pop$Subgroup=="Child" & pop$Compound=="PCB\_TEQ") , ] # Remove because children do not have all PCBs  
  
#### Make pop\_w with wide format congener table  
  
pop\_w <- log(pop)  
  
pop\_w <- reshape(  
 pop\_w@output[  
 pop\_w$Age != "17-19" & pop\_w$Compound %in% c(pcb9, teq3) , # Remove adult patients from LASERI  
 c("Id","Subgroup","Compound","popResult")],  
 v.names = "popResult",  
 idvar="Id",  
 timevar = "Compound",  
 direction="wide")  
colnames(pop\_w) <- gsub("popResult.", "", colnames(pop\_w))  
  
###### Add rows for Women and Donors without TEQ so that predictions are made  
  
pop\_w <- rbind(  
 pop\_w,  
 cbind(Subgroup=paste0(pop\_w$Subgroup,"\_pred"), pop\_w[colnames(pop\_w)!="Subgroup"])  
)  
pop\_w[grepl("\_pred",pop\_w$Subgroup),teq3] <- NA

## Descriptive statistics with adult patients from LASERI

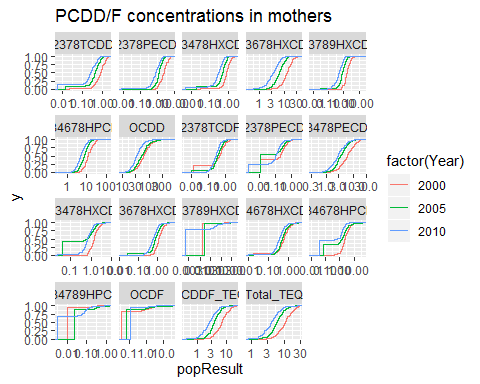
ggplot(pop@output[pop$Subgroup=="Child",], aes(x=popResult, colour=Center))+stat\_ecdf()+scale\_x\_log10()+  
 facet\_wrap(~Compound)+  
 labs(title="PCB congener concentrations in children")



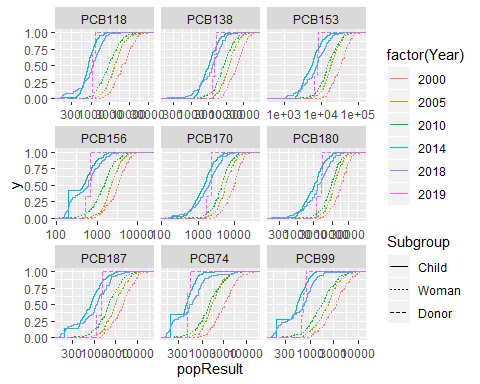
ggplot(pop@output[pop$Subgroup=="Woman" & pop$Compound %in% c(pcb9, teq3),], aes(x=popResult, colour=factor(Year)))+stat\_ecdf()+scale\_x\_log10()+  
 facet\_wrap(~Compound, scales="free\_x")+  
 labs(title="PCB and TEQ concentrations in mothers")



ggplot(pop@output[pop$Subgroup=="Woman" & !grepl("PCB",pop$Compound),], aes(x=popResult, colour=factor(Year)))+stat\_ecdf()+scale\_x\_log10()+  
 facet\_wrap(~Compound, scales="free\_x")+  
 labs(title="PCDD/F concentrations in mothers")



ggplot(pop@output[pop$Compound %in% pcb9 , ], aes(x=popResult, colour=factor(Year), linetype=Subgroup))+stat\_ecdf()+scale\_x\_log10()+  
 facet\_wrap(~Compound, scales="free\_x")

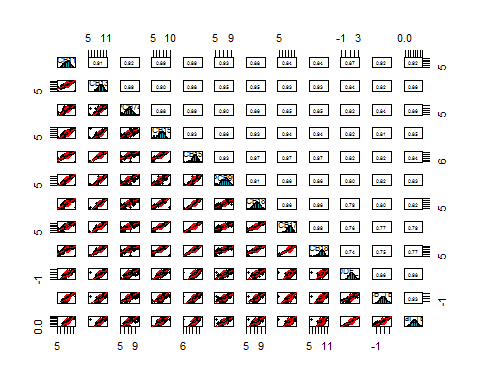


library(psych)

##   
## Attaching package: 'psych'

## The following objects are masked from 'package:ggplot2':  
##   
## %+%, alpha

pairs.panels(pop\_w[c(pcb9,teq3)],   
 method = "pearson", # correlation method  
 hist.col = "#00AFBB",  
 density = TRUE, # show density plots  
 ellipses = TRUE # show correlation ellipses  
 )



We can see that the distributions are mostly in line with lognormal distributions, but with some problems related to missing values. In boys’ data, many congeners show that although the lowest values are comparable in Kuopio and elsewhere, other centers have clearly less of moderately low values, and the median is 50-100 % higher than in Kuopio.

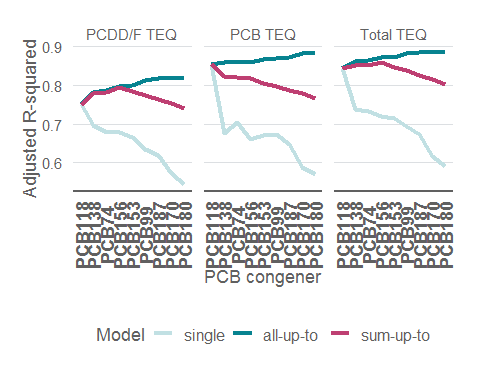
The fraction of values below LOQ is very large in 12378PeCDF 123478HxCDD, 1234789HpCDF, and OCDF. The detection level has improved for several congeners and brought the fraction of samples below LOQ down: 2378TCDF, 12378PECDF, 123789HXCDF, 1234789HPCDF.

## Linear regression analyses for model comparison

We wanted to find an effective way to predict total TEQ (PCDD/F+PCB) concentration in situations where only PCB concentrations are available. This was done by studying properties of several linear regressions and identifying the best one for the data available.

Because the concentration distributions had a reasonable fit with lognormal distributions, all concentrations we log-transformed. Three different TEQ values were used as the dependent variable: PCDD/F TEQ, PCB TEQ and total TEQ. First, TEQ values were predicted using linear regression (lm function in R statistical software version 3.5.3) and PCB concentrations as independent variables, one at a time. Based on the adjusted R squared values of the total TEQ models, PCB congeners were sorted to decreasing order. Second, PCB congeners were added to the regressions as independent variables one by one until all congeners were in a model. Third, rather than using individual congeners, the congeners were summed up and the sum was used as the only independent variable. Fourth, all congeners and all different sums were added to a regression model and two-way inclusion-exclusion approach with Aitken Information Criteria was used to identify the most informative set of independent variables.

#Explanatory power of different models (linear regression)  
  
stati <- data.frame()  
  
# 1. Model using a single independent variable  
  
for(j in teq3) {  
 for(i in pcb9) {  
 dep <- i  
 stati <- rbind(  
 stati,  
 data.frame(  
 Model = "single",  
 Dependent = j,  
 Independents = dep,  
 AdjR = summary(lm(as.formula(paste(j, "~", dep)), data = pop\_w))[["adj.r.squared"]]  
 )  
 )  
 }  
}  
  
##### Sort pbc9 to match the explanatory power  
  
pcb9 <- stati[stati$Model=="single" & stati$Dependent=="Total\_TEQ",]  
pcb9 <- as.character(pcb9$Independents[order(-pcb9$AdjR)])  
  
stati$Independents <- factor(stati$Independents, levels=pcb9)  
  
  
# 2. Model using all independent variables up to the one mentioned  
  
for(j in teq3) {  
 for(i in 1:length(pcb9)) {  
 dep <- pcb9[1:i]  
 stati <- rbind(  
 stati,  
 data.frame(  
 Model = "all-up-to",  
 Dependent = j,  
 Independents = pcb9[i],  
 AdjR = summary(lm(as.formula(paste(j, "~", paste(dep,collapse=" + "))), data = pop\_w))[["adj.r.squared"]]  
 )  
 )  
 }  
}  
  
# 3. Model using a sum of all up to the one mentioned  
  
for(j in teq3) {  
 for(i in 1:length(pcb9)) {  
 pop\_w$tmp <- rowSums(pop\_w[pcb9[1:i]])  
 stati <- rbind(  
 stati,  
 data.frame(  
 Model="sum-up-to",  
 Dependent = j,  
 Independents = pcb9[i],  
 AdjR = summary(lm(as.formula(paste(j, "~ tmp")), data = pop\_w))[["adj.r.squared"]]  
 )  
 )  
 }  
}  
  
# [1] "PCDDF\_TEQ" "PCB\_TEQ" "Total\_TEQ"  
levels(stati$Dependent) <- c("PCDD/F TEQ", "PCB TEQ", "Total TEQ")  
  
ggplot(stati, aes(x=Independents, y=AdjR, color=Model, group=Model))+geom\_line(size=1.5)+  
 facet\_grid(.~Dependent)+  
 scale\_color\_manual(values=c(light,dark,ruby))+  
 labs(  
# title="Explanatory power of PCB combinations on TEQ",  
 y="Adjusted R-squared",  
 x="PCB congener"  
 )+  
 theme\_thl(legend.position="bottom", base\_size=18, base\_family = "source", axis.text.x=element\_text(angle=90, vjust=0.5))



ggsave("Explanatory\_power\_of\_PCB\_on\_TEQ.pdf",width=16/2.54, height=18/2.54)  
  
# 4. Model using AIC to find the best set of variables from congeners and congener sums  
  
for(j in teq3) {  
 dep <- pcb9  
 mod <- lm(as.formula(paste(j, "~", paste(dep,collapse=" + "))), data=na.omit(pop\_w[c(j, pcb9)]))  
 print(summary(stepAIC(mod, direction = "both", trace = FALSE)))  
}

##   
## Call:  
## lm(formula = PCDDF\_TEQ ~ PCB118 + PCB138 + PCB74 + PCB156 + PCB99 +   
## PCB187 + PCB180, data = na.omit(pop\_w[c(j, pcb9)]))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.01371 -0.16759 -0.01129 0.15111 1.31176   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -6.61775 0.24370 -27.155 < 2e-16 \*\*\*  
## PCB118 0.46871 0.05674 8.261 1.16e-15 \*\*\*  
## PCB138 0.79359 0.08488 9.349 < 2e-16 \*\*\*  
## PCB74 0.12819 0.04183 3.064 0.00229 \*\*   
## PCB156 0.26711 0.05217 5.120 4.29e-07 \*\*\*  
## PCB99 -0.43355 0.07612 -5.695 2.04e-08 \*\*\*  
## PCB187 -0.22889 0.07495 -3.054 0.00237 \*\*   
## PCB180 -0.12732 0.07477 -1.703 0.08920 .   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.2785 on 530 degrees of freedom  
## Multiple R-squared: 0.8204, Adjusted R-squared: 0.818   
## F-statistic: 345.8 on 7 and 530 DF, p-value: < 2.2e-16  
##   
##   
## Call:  
## lm(formula = PCB\_TEQ ~ PCB118 + PCB138 + PCB156 + PCB153 + PCB99 +   
## PCB187 + PCB170 + PCB180, data = na.omit(pop\_w[c(j, pcb9)]))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.25072 -0.09858 0.00846 0.11755 1.24713   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -8.05376 0.27429 -29.362 < 2e-16 \*\*\*  
## PCB118 1.05560 0.04469 23.622 < 2e-16 \*\*\*  
## PCB138 -0.17581 0.10073 -1.745 0.08148 .   
## PCB156 -0.21411 0.04714 -4.542 6.91e-06 \*\*\*  
## PCB153 0.16873 0.11015 1.532 0.12618   
## PCB99 -0.11641 0.06457 -1.803 0.07197 .   
## PCB187 -0.18801 0.06903 -2.723 0.00667 \*\*   
## PCB170 0.75706 0.16628 4.553 6.58e-06 \*\*\*  
## PCB180 -0.28048 0.18329 -1.530 0.12655   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.2369 on 529 degrees of freedom  
## Multiple R-squared: 0.8844, Adjusted R-squared: 0.8826   
## F-statistic: 505.8 on 8 and 529 DF, p-value: < 2.2e-16  
##   
##   
## Call:  
## lm(formula = Total\_TEQ ~ PCB118 + PCB138 + PCB74 + PCB156 + PCB99 +   
## PCB187 + PCB170, data = na.omit(pop\_w[c(j, pcb9)]))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.61405 -0.13817 -0.00782 0.11391 1.28612   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -6.53627 0.15675 -41.699 < 2e-16 \*\*\*  
## PCB118 0.67019 0.04442 15.088 < 2e-16 \*\*\*  
## PCB138 0.51399 0.07033 7.308 1.00e-12 \*\*\*  
## PCB74 0.08240 0.03272 2.518 0.0121 \*   
## PCB156 0.10681 0.04244 2.517 0.0121 \*   
## PCB99 -0.33108 0.05981 -5.536 4.88e-08 \*\*\*  
## PCB187 -0.23383 0.05153 -4.538 7.04e-06 \*\*\*  
## PCB170 0.10392 0.05653 1.838 0.0666 .   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.2165 on 530 degrees of freedom  
## Multiple R-squared: 0.8865, Adjusted R-squared: 0.885   
## F-statistic: 591.5 on 7 and 530 DF, p-value: < 2.2e-16

# 5. Model using AIC to find the best set of variables from congeners and all two-way interaction terms  
  
for(j in teq3) {  
 dep <- c(pcb9)  
 mod <- lm(as.formula(paste(j, "~ (", paste(dep,collapse=" + "), ")^2")), data=pop\_w)  
 print(summary(stepAIC(mod, direction = "both", trace = FALSE)))  
  
}

##   
## Call:  
## lm(formula = PCDDF\_TEQ ~ PCB118 + PCB138 + PCB74 + PCB156 + PCB153 +   
## PCB99 + PCB187 + PCB170 + PCB180 + PCB118:PCB138 + PCB118:PCB153 +   
## PCB118:PCB187 + PCB138:PCB74 + PCB138:PCB153 + PCB138:PCB99 +   
## PCB138:PCB170 + PCB138:PCB180 + PCB74:PCB153 + PCB74:PCB99 +   
## PCB74:PCB180 + PCB156:PCB153 + PCB156:PCB187 + PCB156:PCB170 +   
## PCB156:PCB180 + PCB153:PCB187 + PCB153:PCB170 + PCB153:PCB180 +   
## PCB99:PCB180 + PCB187:PCB180 + PCB170:PCB180, data = pop\_w)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.06695 -0.15218 -0.00318 0.12743 1.31427   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -6.8393 5.0914 -1.343 0.179776   
## PCB118 -2.9240 1.0113 -2.891 0.004003 \*\*   
## PCB138 2.1534 2.0699 1.040 0.298686   
## PCB74 0.4622 0.8582 0.539 0.590427   
## PCB156 2.5827 1.0347 2.496 0.012872 \*   
## PCB153 0.2552 2.3770 0.107 0.914527   
## PCB99 -0.6340 1.2182 -0.520 0.603012   
## PCB187 -0.1232 1.4418 -0.085 0.931914   
## PCB170 -7.0449 3.8635 -1.823 0.068820 .   
## PCB180 5.7705 3.9699 1.454 0.146688   
## PCB118:PCB138 -1.0899 0.3175 -3.433 0.000647 \*\*\*  
## PCB118:PCB153 1.7809 0.3475 5.124 4.25e-07 \*\*\*  
## PCB118:PCB187 -0.5121 0.1955 -2.620 0.009067 \*\*   
## PCB138:PCB74 1.1829 0.3281 3.605 0.000343 \*\*\*  
## PCB138:PCB153 -1.0518 0.3805 -2.764 0.005914 \*\*   
## PCB138:PCB99 0.7182 0.1938 3.705 0.000235 \*\*\*  
## PCB138:PCB170 -2.6276 1.0308 -2.549 0.011096 \*   
## PCB138:PCB180 2.8192 1.0178 2.770 0.005812 \*\*   
## PCB74:PCB153 -1.3225 0.3840 -3.444 0.000621 \*\*\*  
## PCB74:PCB99 -0.3392 0.1563 -2.170 0.030453 \*   
## PCB74:PCB180 0.4614 0.2120 2.176 0.030005 \*   
## PCB156:PCB153 -1.3765 0.3947 -3.488 0.000530 \*\*\*  
## PCB156:PCB187 0.7384 0.2636 2.801 0.005288 \*\*   
## PCB156:PCB170 -1.1419 0.6692 -1.706 0.088584 .   
## PCB156:PCB180 1.6238 0.7433 2.184 0.029386 \*   
## PCB153:PCB187 0.5564 0.3835 1.451 0.147458   
## PCB153:PCB170 4.7985 1.3638 3.519 0.000473 \*\*\*  
## PCB153:PCB180 -3.2932 1.3072 -2.519 0.012066 \*   
## PCB99:PCB180 -0.4157 0.2122 -1.959 0.050693 .   
## PCB187:PCB180 -0.7422 0.3989 -1.861 0.063388 .   
## PCB170:PCB180 -0.7465 0.3829 -1.949 0.051812 .   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.2647 on 507 degrees of freedom  
## (906 observations deleted due to missingness)  
## Multiple R-squared: 0.8448, Adjusted R-squared: 0.8356   
## F-statistic: 91.98 on 30 and 507 DF, p-value: < 2.2e-16  
##   
##   
## Call:  
## lm(formula = PCB\_TEQ ~ PCB118 + PCB138 + PCB74 + PCB156 + PCB153 +   
## PCB99 + PCB187 + PCB170 + PCB180 + PCB118:PCB74 + PCB118:PCB156 +   
## PCB118:PCB99 + PCB118:PCB187 + PCB118:PCB170 + PCB118:PCB180 +   
## PCB138:PCB153 + PCB138:PCB170 + PCB74:PCB99 + PCB156:PCB187 +   
## PCB156:PCB180 + PCB153:PCB187 + PCB99:PCB187 + PCB99:PCB180 +   
## PCB187:PCB170 + PCB170:PCB180, data = pop\_w)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.14958 -0.10123 0.00363 0.10943 1.19802   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -18.79767 3.82317 -4.917 1.19e-06 \*\*\*  
## PCB118 -0.29293 0.82348 -0.356 0.722195   
## PCB138 -1.55715 1.26226 -1.234 0.217909   
## PCB74 -0.13619 0.47571 -0.286 0.774770   
## PCB156 -6.84583 1.18450 -5.780 1.30e-08 \*\*\*  
## PCB153 1.39804 1.36298 1.026 0.305507   
## PCB99 3.00430 1.06633 2.817 0.005028 \*\*   
## PCB187 -2.19345 1.09248 -2.008 0.045193 \*   
## PCB170 9.19680 2.36682 3.886 0.000115 \*\*\*  
## PCB180 0.14008 2.27255 0.062 0.950873   
## PCB118:PCB74 0.25319 0.10716 2.363 0.018512 \*   
## PCB118:PCB156 0.72805 0.13140 5.541 4.83e-08 \*\*\*  
## PCB118:PCB99 -0.16723 0.09404 -1.778 0.075954 .   
## PCB118:PCB187 -0.31791 0.21171 -1.502 0.133824   
## PCB118:PCB170 -1.03194 0.32592 -3.166 0.001636 \*\*   
## PCB118:PCB180 0.70477 0.37018 1.904 0.057489 .   
## PCB138:PCB153 0.52350 0.18605 2.814 0.005084 \*\*   
## PCB138:PCB170 -0.41874 0.23758 -1.762 0.078586 .   
## PCB74:PCB99 -0.25318 0.10891 -2.325 0.020479 \*   
## PCB156:PCB187 -1.29755 0.32420 -4.002 7.20e-05 \*\*\*  
## PCB156:PCB180 1.18301 0.28403 4.165 3.65e-05 \*\*\*  
## PCB153:PCB187 -0.78137 0.23372 -3.343 0.000889 \*\*\*  
## PCB99:PCB187 0.83212 0.19684 4.227 2.80e-05 \*\*\*  
## PCB99:PCB180 -0.69269 0.21520 -3.219 0.001369 \*\*   
## PCB187:PCB170 1.83387 0.36878 4.973 9.02e-07 \*\*\*  
## PCB170:PCB180 -1.14849 0.27060 -4.244 2.61e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.2242 on 512 degrees of freedom  
## (906 observations deleted due to missingness)  
## Multiple R-squared: 0.8997, Adjusted R-squared: 0.8948   
## F-statistic: 183.7 on 25 and 512 DF, p-value: < 2.2e-16  
##   
##   
## Call:  
## lm(formula = Total\_TEQ ~ PCB118 + PCB138 + PCB74 + PCB156 + PCB153 +   
## PCB99 + PCB187 + PCB170 + PCB180 + PCB118:PCB138 + PCB118:PCB74 +   
## PCB118:PCB153 + PCB118:PCB187 + PCB138:PCB74 + PCB138:PCB153 +   
## PCB138:PCB99 + PCB138:PCB170 + PCB138:PCB180 + PCB74:PCB153 +   
## PCB74:PCB99 + PCB74:PCB180 + PCB156:PCB153 + PCB156:PCB187 +   
## PCB153:PCB170 + PCB153:PCB180 + PCB99:PCB180 + PCB170:PCB180,   
## data = pop\_w)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.5142 -0.1293 -0.0079 0.1121 1.2486   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -7.74440 3.76393 -2.058 0.040143 \*   
## PCB118 -2.17167 0.75249 -2.886 0.004067 \*\*   
## PCB138 1.66312 1.57438 1.056 0.291301   
## PCB74 0.41299 0.66733 0.619 0.536282   
## PCB156 1.37955 0.77405 1.782 0.075303 .   
## PCB153 -0.06377 1.69968 -0.038 0.970088   
## PCB99 -0.14971 0.95538 -0.157 0.875543   
## PCB187 -0.74167 0.73760 -1.006 0.315123   
## PCB170 -2.86226 2.69670 -1.061 0.289014   
## PCB180 3.46349 2.64881 1.308 0.191610   
## PCB118:PCB138 -0.97823 0.24986 -3.915 0.000103 \*\*\*  
## PCB118:PCB74 0.16907 0.09283 1.821 0.069148 .   
## PCB118:PCB153 1.27700 0.25403 5.027 6.91e-07 \*\*\*  
## PCB118:PCB187 -0.24636 0.13303 -1.852 0.064606 .   
## PCB138:PCB74 0.79493 0.25485 3.119 0.001916 \*\*   
## PCB138:PCB153 -0.53881 0.28408 -1.897 0.058436 .   
## PCB138:PCB99 0.65177 0.15265 4.270 2.34e-05 \*\*\*  
## PCB138:PCB170 -2.16956 0.77265 -2.808 0.005177 \*\*   
## PCB138:PCB180 2.14851 0.77569 2.770 0.005813 \*\*   
## PCB74:PCB153 -0.90703 0.29827 -3.041 0.002479 \*\*   
## PCB74:PCB99 -0.37395 0.13625 -2.745 0.006271 \*\*   
## PCB74:PCB180 0.28969 0.16399 1.767 0.077906 .   
## PCB156:PCB153 -0.40343 0.16407 -2.459 0.014269 \*   
## PCB156:PCB187 0.34209 0.14028 2.439 0.015080 \*   
## PCB153:PCB170 2.69569 0.87540 3.079 0.002186 \*\*   
## PCB153:PCB180 -2.02836 0.77919 -2.603 0.009506 \*\*   
## PCB99:PCB180 -0.36876 0.16385 -2.251 0.024836 \*   
## PCB170:PCB180 -0.34892 0.17697 -1.972 0.049184 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.2074 on 510 degrees of freedom  
## (906 observations deleted due to missingness)  
## Multiple R-squared: 0.8998, Adjusted R-squared: 0.8945   
## F-statistic: 169.7 on 27 and 510 DF, p-value: < 2.2e-16

## Additional regression analyses

## Linear regression sensitivity analysis without Kuopio, Year and Parity

We can make several conslusions based on the additional regression model results. Data is not shown here because we decided to put these model comparison issues to another manuscript and only focus on PCB9 model here.

1. As expected, regression finds a perfect fit when all consituents of a TEQ value are present.
2. Year shows up in every model implying that the PCB profile changes in time.
3. Parity is not important determinant in models given PCB concentrations. This implies that although it affects the absolute concentrations, it does not affect congener profile in a way that would affect TEQ estimates. This is also seen in the graph above: all congeners show the same slightly decreasing trend with parity.
4. City shows up in the models implying that there are differences in congener profiles between cities.
5. If Kuopio and Year are included, PCB138 drops out, but if not, then PCB 153 drops out. This implies that PCB138 contains important information that is associated with Year and location. This should be studied more.
6. SUM9PCB never shows up in any model, indicating that individual congeners do much better work in predicting TEQs. This is not surprising because a lot of information is lost when summing things up.
7. When we look at the distributions of congener concentrations scaled by SUM9PCB, there are very little differences between moms and boys. This implies that the congener profile derived from moms is a fair representation of congener profile in boys. However, this conclusion is based on PCB alone, because there are no data about PCDD/F in boys.
8. A hypothesis is that the most recent mothers’ milk data should be used rather than the whole data, because that would temporally reflect the children’s data more precisely. However, when the regressions model analysis is run with the newest (2010) data only (data not shown), the regression fits are clearly poorer: R2 values are 82, and 76, and 82 % rather than 87, 82, and 87 % for PCB, PCDD/F, and Total TEQ, respectively. Therefore, we will use the whole WHO data.
9. Data about Kuopio and Year are important: linear regression fit drops from 87 to 87 % for PCB TEQ if they are omitted, 82 to 75 % for PCDD/F, and from 87 to 83 % for Total TEQ.
10. The R2 fit is somewhat poorer with Bayesian model because it did not use data about Kuopio and Year in analysis: 67, 67, and 76 % for 2010 data only and 75, 75, and 82 % for all WHO data, respectively.

## Bayesian hierachical model

# This was forked from code Op\_en3104/bayes on page [[EU-kalat]]  
  
library(OpasnetUtils)  
library(reshape2)  
library(rjags) # JAGS

## Loading required package: coda

## Linked to JAGS 4.3.0

## Loaded modules: basemod,bugs

library(ggplot2)  
library(MASS) # mvrnorm  
library(car) # scatterplotMatrix

## Loading required package: carData

##   
## Attaching package: 'car'

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

N <- 500  
  
# Hierarchical Bayes model.  
  
mod <- textConnection(  
 "  
 model{  
 for(j in C) { # TEQ columns (after pcb9 columns)  
 tau[j] ~ dgamma(1.0, 1.0)  
 for(i in S) { # S = human sample  
 # below.LOQ[i,j] ~ dinterval(-conc[i,j], -LOQ[j])  
 teq[i,j] ~ dnorm(mu[i,j], tau[j])  
  
 mu[i,j] <- b[j,1]\*conc[i,1] + b[j,2]\*conc[i,2] + b[j,3]\*conc[i,3] +  
 b[j,4]\*conc[i,4] + b[j,5]\*conc[i,5] + b[j,6]\*conc[i,6] +  
 b[j,7]\*conc[i,7] + b[j,8]\*conc[i,8] + b[j,9]\*conc[i,9]   
 }  
 for(k in K) {  
 b[j,k] ~ dnorm(0, 0.0001) # Congener-specific coefficient for TEQs  
 }  
 }  
 }  
")  
  
jags <- jags.model(  
 mod,  
 data = list(  
 S = 1:nrow(pop\_w),  
 C = 1:2,  
 K = 1:length(pcb9),  
 conc = data.matrix(pop\_w[pcb9]),   
 teq = data.matrix(pop\_w[teq3])  
 ),  
 n.chains = 4,  
 n.adapt = 1000  
)

## Compiling model graph  
## Resolving undeclared variables  
## Allocating nodes  
## Graph information:  
## Observed stochastic nodes: 1076  
## Unobserved stochastic nodes: 1832  
## Total graph size: 31862  
##   
## Initializing model

samps.j <- jags.samples(  
 jags,   
 c(  
 'mu',   
 'tau',  
 'b',  
 'teq'  
 ),   
 thin=100,  
 N\*100  
)

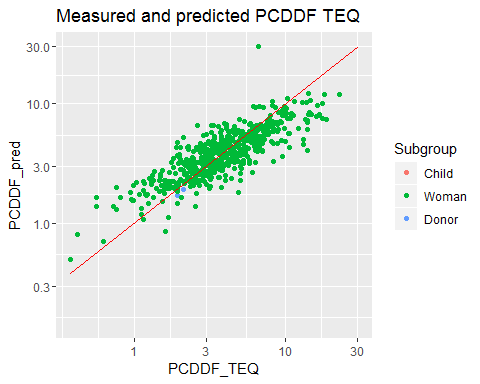
The Bayes model is now run. PCDD/F TEQ is taken as the target TEQ.

# Postprocess the model output  
  
dimnames(samps.j$b) <- list(TEQ = teq3[1:2], Compound = pcb9, Iter = 1:N, Chain = 1:4)  
dimnames(samps.j$mu) <- list(Id = 1:nrow(pop\_w), TEQ = paste0(teq3[1:2], "\_mu"), Iter = 1:N, Chain = 1:4)  
dimnames(samps.j$tau) <- list(TEQ = teq3[1:2], Iter = 1:N, Chain = 1:4)  
dimnames(samps.j$teq) <- list(Subgroup = pop\_w$Subgroup, TEQ = paste0(teq3, "\_pred"), Iter = 1:N, Chain = 1:4)  
  
samps.j$teq[,3,,] <- apply(samps.j$teq[,1:2,,],c(1,3,4), function(x) log(sum(exp(x)))) # Fill in Total\_TEQ slot in TEQ  
  
# The the PCDDF\_TEQ as the main outcome  
  
out <- cbind(  
 pop\_w,  
 exp(apply(samps.j$mu, MARGIN=c("Id","TEQ"), FUN=mean)),  
 exp(apply(samps.j$teq, MARGIN=c("Subgroup","TEQ"), FUN=mean)),  
 exp(t(apply(samps.j$teq[,1,,], MARGIN=c("Subgroup"), FUN=function(x) quantile(x, probs = c(0.05,0.5,0.95))))),  
 exp(t(apply(samps.j$teq[,2,,], MARGIN=c("Subgroup"), FUN=function(x) quantile(x, probs = c(0.05,0.5,0.95))))),  
 exp(t(apply(samps.j$teq[,3,,], MARGIN=c("Subgroup"), FUN=function(x) quantile(x, probs = c(0.05,0.5,0.95))))),  
 Sample = exp(apply(samps.j$teq[,,,], MARGIN=c("Subgroup","TEQ"), FUN=function(x) sample(x, size=1))),  
 P\_exceed = apply(samps.j$teq[,,,], MARGIN=c("Subgroup","TEQ"), FUN=function(x) mean(x>log(7)))  
)  
  
out[c(pcb9,teq3)] <- exp(out[c(pcb9,teq3)]) # Convert the original data back to arithmetic scale  
  
############# Calculate confidence intervals for the cumulative sample distribution  
# This calculation is based on the wrong assumption that you could  
# Sort random samples of n individuals and calculate confidence intervals for each quantile.  
# This cannot be done because you create bias if you fisrt sort and then get a discribution,  
# because the highest quantiles are first selected for being high numbers, and then taking an uppoer  
# limit of that will produce double upper limits  
#  
# Instead, you either have to calculate vertical uncertainties using pointwise or simultaneous bands.  
# Neither of these is what we want here, becaues our focus is on actual individuals more than   
# on the distribution. Therefore, this calculation is not used.  
# https://stats.stackexchange.com/questions/181724/confidence-intervals-for-ecdf  
# https://en.wikipedia.org/wiki/CDF-based\_nonparametric\_confidence\_interval  
  
if(FALSE) {  
samp <- exp(samps.j$teq[,,,1]) #[grepl("Child\_pred", names(samps.j$teq[,1,1,1])),,,1])  
samp <- melt(samp)  
samp$Order <- NA  
tst <- factor(paste(samp$TEQ,samp$Iter))  
  
for(i in unique(tst)) {  
 samp$Order[tst == i] <- rank(samp$value[tst == i])  
}  
  
ggplot(samp[samp$Subgroup == "Child\_pred",], aes(x=value, color=TEQ))+stat\_ecdf()+  
 geom\_vline(xintercept = 7)+  
 scale\_x\_log10()  
  
samp <- t(sapply(unique(samp$Order), FUN = function(x) quantile(samp$value[samp$Order==x], probs = c(0.05, 0.5, 0.95))))  
colnames(samp) <- gsub("%", "", paste0("Cumul", colnames(samp)))  
  
out <- cbind(out, samp)  
} # END IF  
  
#### Move predictions to their right places  
  
out[!grepl("\_pred", out$Subgroup) , grepl("(\_pred|%)", colnames(out))] <-   
 out[grepl("\_pred", out$Subgroup) , grepl("(\_pred|%)", colnames(out))]  
  
colnames(out)[grepl("%", colnames(out))] <- c("PCDDF\_P05", "PCDDF\_P50", "PCDDF\_P95", "PCB\_P05","PCB\_P50", "PCB\_P95",  
 "SUM\_P05", "SUM\_P50","SUM\_P95")  
out <- out[!grepl("\_pred", out$Subgroup),]  
  
colnames(out)[match(c(  
 "PCDDF\_TEQ\_pred","PCB\_TEQ\_pred","Total\_TEQ\_pred",  
 "Sample.PCDDF\_TEQ\_pred","Sample.PCB\_TEQ\_pred", "Sample.Total\_TEQ\_pred",  
 "P\_exceed.PCDDF\_TEQ\_pred", "P\_exceed.PCB\_TEQ\_pred", "P\_exceed.Total\_TEQ\_pred"),colnames(out))] <-  
 c("PCDDF\_pred","PCB\_pred","SUM\_pred","PCDDF\_sample","PCB\_sample","SUM\_sample","PCDDF\_p\_exceed","PCB\_p\_exceed","SUM\_p\_exceed")  
  
out <- merge(out, unique(pop@output[c("Id","Cohort","Sex","Age","Center","Parity","Birthyear","Year")]))

### Comparing measured and predicted values

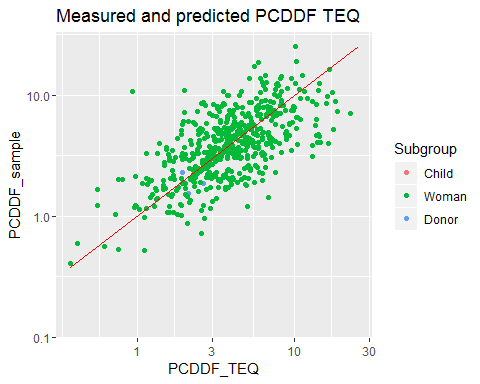
lin <- function(df) {df <- na.omit(df); df <- c(min(df), max(df)); data.frame(x=df, y=df)}  
  
ggplot(out, aes(x=PCDDF\_TEQ, y=PCDDF\_pred, color=Subgroup))+geom\_point()+  
 geom\_line(data=lin(out[c("PCDDF\_TEQ","PCDDF\_pred")]), aes(x=x, y=y),color="red")+  
 scale\_x\_log10()+scale\_y\_log10()+  
 labs(title="Measured and predicted PCDDF TEQ")

## Warning: Removed 184 rows containing missing values (geom\_point).



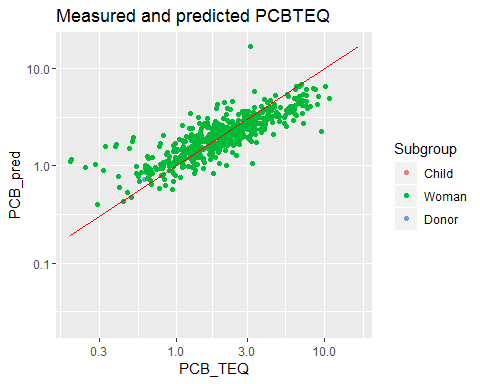
ggplot(out, aes(x=PCDDF\_TEQ, y=PCDDF\_sample, color=Subgroup))+geom\_point()+  
 geom\_line(data=lin(out[c("PCDDF\_TEQ","PCDDF\_sample")]), aes(x=x, y=y),color="red")+  
 scale\_x\_log10()+scale\_y\_log10()+  
 labs(title="Measured and predicted PCDDF TEQ")

## Warning: Removed 184 rows containing missing values (geom\_point).



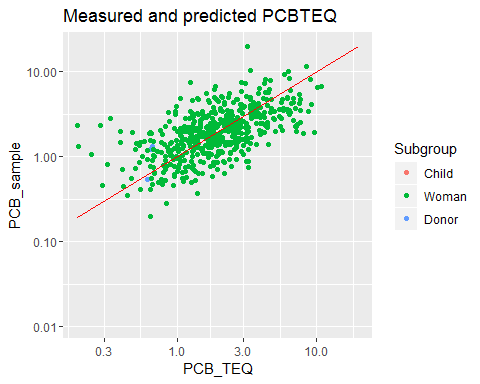
ggplot(out, aes(x=PCB\_TEQ, y=PCB\_pred, color=Subgroup))+geom\_point()+  
 geom\_line(data=lin(out[c("PCB\_TEQ","PCB\_pred")]),aes(x=x, y=y),color="red")+  
 scale\_x\_log10()+scale\_y\_log10()+  
 labs(title="Measured and predicted PCBTEQ")

## Warning: Removed 184 rows containing missing values (geom\_point).



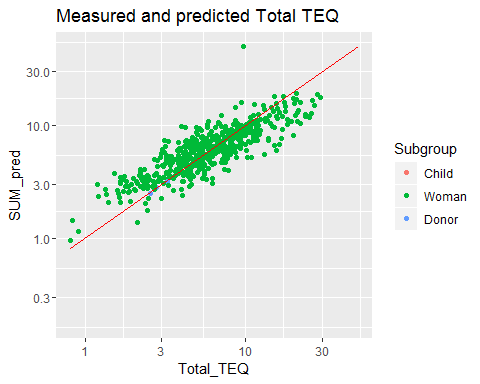
ggplot(out, aes(x=PCB\_TEQ, y=PCB\_sample, color=Subgroup))+geom\_point()+  
 geom\_line(data=lin(out[c("PCB\_TEQ","PCB\_sample")]),aes(x=x, y=y),color="red")+  
 scale\_x\_log10()+scale\_y\_log10()+  
 labs(title="Measured and predicted PCBTEQ")

## Warning: Removed 184 rows containing missing values (geom\_point).



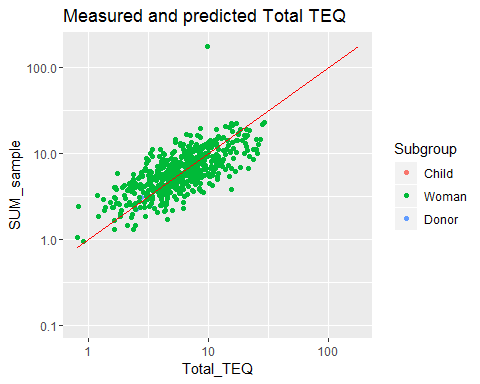
ggplot(out, aes(x=Total\_TEQ, y=SUM\_pred, color=Subgroup))+geom\_point()+  
 geom\_line(data=lin(out[c("Total\_TEQ","SUM\_pred")]),aes(x=x, y=y),color="red")+  
 scale\_x\_log10()+scale\_y\_log10()+  
 labs(title="Measured and predicted Total TEQ")

## Warning: Removed 184 rows containing missing values (geom\_point).



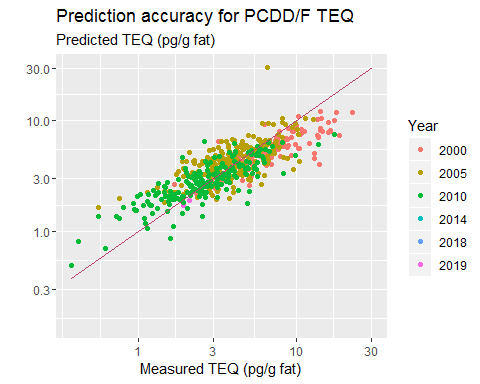
ggplot(out, aes(x=Total\_TEQ, y=SUM\_sample, color=Subgroup))+geom\_point()+  
 geom\_line(data=lin(out[c("Total\_TEQ","SUM\_sample")]),aes(x=x, y=y),color="red")+  
 scale\_x\_log10()+scale\_y\_log10()+  
 labs(title="Measured and predicted Total TEQ")

## Warning: Removed 184 rows containing missing values (geom\_point).

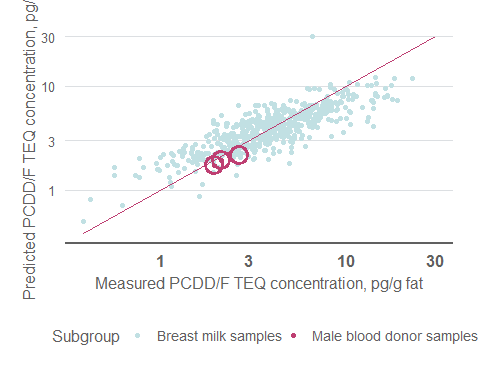


ggplot(out, aes(x=PCDDF\_TEQ, y=PCDDF\_pred, colour=Year))+geom\_point()+  
 geom\_line(data=lin(out[c("PCDDF\_TEQ","PCDDF\_pred")]),aes(x=x, y=y),color=ruby)+  
 scale\_x\_log10()+scale\_y\_log10()+  
 labs(  
 title="Prediction accuracy for PCDD/F TEQ",  
 subtitle="Predicted TEQ (pg/g fat)",  
 y="",  
 x="Measured TEQ (pg/g fat)")

## Warning: Removed 184 rows containing missing values (geom\_point).



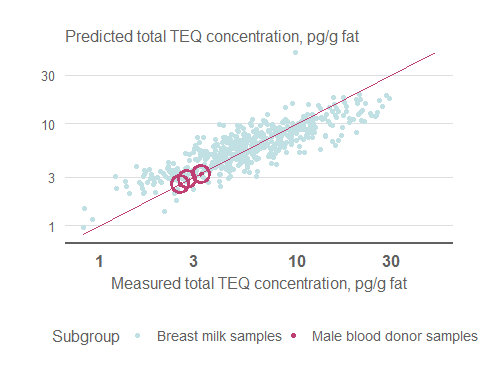
#### Figure b for measured vs predicted TEQs in adults  
  
tmp <- out[out$Subgroup %in% c("Woman","Donor"),]  
tmp$Subgroup <- factor(tmp$Subgroup, levels=c("Woman","Donor"), labels= c("Breast milk samples","Male blood donor samples"))  
  
fig1b <-   
ggplot(tmp, aes(x=PCDDF\_TEQ, y=PCDDF\_pred, colour=Subgroup))+geom\_point()+  
 geom\_line(data=lin(tmp[c("PCDDF\_TEQ","PCDDF\_pred")]),aes(x=x, y=y),color=ruby)+  
 geom\_point(data=out[out$Subgroup=="Donor",], aes(x=PCDDF\_TEQ, y=PCDDF\_pred),  
 colour=ruby, size=5, shape=1, stroke=2)+  
 scale\_color\_manual(values=c(light, ruby))+  
 scale\_x\_log10()+scale\_y\_log10()+  
 theme\_thl(base\_family="source", legend.position="bottom")+  
 labs(  
# title="Prediction accuracy for PCDD/F TEQ",  
 y="Predicted PCDD/F TEQ concentration, pg/g fat",  
 x="Measured PCDD/F TEQ concentration, pg/g fat")  
  
fig1bdata <- fig1b$data[c("Subgroup","PCDDF\_TEQ","PCDDF\_pred")]  
write.csv(fig1bdata, "fig1bdata.csv")  
fig1b



oprint(fig1bdata[fig1bdata$Subgroup=="Male blood donor samples",])

## Subgroup PCDDF\_TEQ PCDDF\_pred  
## 298 Male blood donor samples 2.632522 2.165548  
## 299 Male blood donor samples 1.935719 1.729268  
## 300 Male blood donor samples 2.122546 1.918963

ggsave("Predicted PCDDF TEQ.pdf", width=16/2.54, height=18/2.54)  
  
fig2b <-   
ggplot(tmp, aes(x=Total\_TEQ, y=SUM\_pred, colour=Subgroup))+geom\_point()+  
 geom\_line(data=lin(tmp[c("Total\_TEQ","SUM\_pred")]),aes(x=x, y=y),color=ruby)+  
 geom\_point(data=out[out$Subgroup=="Donor",], aes(x=Total\_TEQ, y=SUM\_pred),  
 colour=ruby, size=5, shape=1, stroke=2)+  
 scale\_color\_manual(values=c(light, ruby))+  
 scale\_x\_log10()+scale\_y\_log10()+  
 theme\_thl(base\_family="source", legend.position="bottom")+  
 labs(  
# title="Prediction accuracy for Total TEQ",  
 subtitle="Predicted total TEQ concentration, pg/g fat",  
 y="",  
 x="Measured total TEQ concentration, pg/g fat")  
  
fig2bdata <- fig2b$data[c("Subgroup","Total\_TEQ","SUM\_pred")]  
write.csv(fig2bdata, "fig2bdata.csv")  
fig2b



oprint(fig2bdata[fig2bdata$Subgroup=="Male blood donor samples",])

## Subgroup Total\_TEQ SUM\_pred  
## 298 Male blood donor samples 3.274491 3.182186  
## 299 Male blood donor samples 2.542641 2.532265  
## 300 Male blood donor samples 2.783409 2.856377

ggsave("Predicted Total TEQ.pdf", width=16/2.54, height=18/2.54)  
  
cat("R^2^ for PCDD/F TEQ measured and predicted\n")

## R^2^ for PCDD/F TEQ measured and predicted

cor(log(out[c("PCDDF\_TEQ", "PCDDF\_pred")]), use="pairwise.complete.obs")^2

## PCDDF\_TEQ PCDDF\_pred  
## PCDDF\_TEQ 1.0000000 0.6817618  
## PCDDF\_pred 0.6817618 1.0000000

cat("R^2^ for PCB TEQ measured and predicted\n")

## R^2^ for PCB TEQ measured and predicted

cor(log(out[c("PCB\_TEQ", "PCB\_pred")]), use="pairwise.complete.obs")^2

## PCB\_TEQ PCB\_pred  
## PCB\_TEQ 1.0000000 0.6812228  
## PCB\_pred 0.6812228 1.0000000

cat("R^2^ for Total TEQ measured and predicted\n")

## R^2^ for Total TEQ measured and predicted

cor(log(out[c("Total\_TEQ", "SUM\_pred")]), use="pairwise.complete.obs")^2

## Total\_TEQ SUM\_pred  
## Total\_TEQ 1.000000 0.716854  
## SUM\_pred 0.716854 1.000000

# Why are the R^2 values so poor? With linear regression, they are ca 0.85, with Bayesian model ca 0.65.

Figure 1. Measured versus predicted PCDD/F TEQ (right), and cumulative distribution of individual predicted values (left; green: median estimate, blue: random draw from individual probability distribution, gray: 90 % confidence interval of individual estimates). The left panel shows adults and the right panel shows children in general Finnish population. A red one-to-one line is shown for comparison.

Figure 2. Measured versus predicted total TEQ (right), and cumulative distribution of individual predicted values (left; green: median estimate, blue: random draw from individual probability distribution, gray: 90 % confidence interval of individual estimates). The left panel shows adults and the right panel shows children in general Finnish population. A red one-to-one line is shown for comparison.

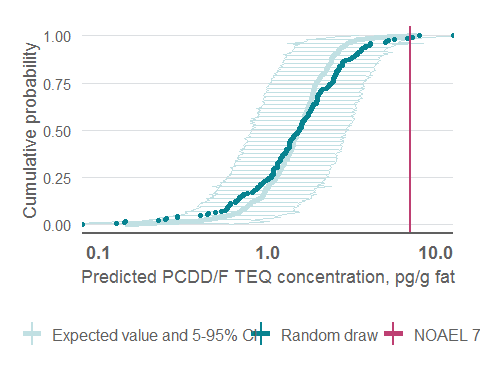
Model is fairly good at predicting measured results, R2 values are 65 % or more.

### Predicted TEQ concentrations

if(FALSE) {  
library(thlGraphs)  
  
ggplot(out)+  
 stat\_ecdf(aes(x=PCDDF\_pred, colour=Subgroup, linetype="Expectation"))+  
 stat\_ecdf(aes(x=PCDDF\_P05, colour=Subgroup, linetype="90 % CI"))+  
 stat\_ecdf(aes(x=PCDDF\_P50, colour=Subgroup, linetype="Median"))+  
 stat\_ecdf(aes(x=PCDDF\_P95, colour=Subgroup, linetype="90 % CI"))+  
 geom\_vline(xintercept=7, colour="red")+  
 scale\_x\_log10()+  
 labs(title="Predicted and confidence interval")  
  
# https://stackoverflow.com/questions/20277658/how-to-combine-stat-ecdf-with-geom-ribbon  
}

### Cumulative distributions of modelled concentrations

tmp <- out[out$Subgroup=="Child",]  
  
#### Figures with cumulative distributions, including individual uncertainties  
  
### Children sorted based on random value  
# These calculations were removed because the audience thought they were confusing  
  
### Children sorted based on expected value  
  
tmp <- tmp[order(tmp$PCDDF\_pred),]  
tmp$y <- seq(0,1,1/(nrow(tmp)-1))  
  
tmp$x <- seq(  
 min(tmp[c("PCDDF\_P05","PCDDF\_sample","PCDDF\_P95")]),  
 max(tmp[c("PCDDF\_P05","PCDDF\_sample","PCDDF\_P95")]),  
 length.out = nrow(tmp))  
  
fig1a <-   
ggplot(tmp) +   
 geom\_path(aes(x=PCDDF\_P05, y=y), colour=light)+  
 geom\_path(aes(x=PCDDF\_P95, y=y), colour=light)+  
 geom\_line(data=data.frame(  
 Id = rep(tmp$Id,2),  
 Conc = c(tmp$PCDDF\_P05, tmp$PCDDF\_P95),  
 y = rep((1:nrow(tmp))/nrow(tmp),2))[c(seq(1,nrow(tmp),4),seq(1,nrow(tmp),4)+nrow(tmp)),], # Thin the CI so that you can see the lines  
 aes(x=Conc, y=y, group=Id),  
 size=0.05, colour=light)+  
 geom\_line(aes(x = PCDDF\_pred, y=y, colour="Expected value and 5-95% CI"), size=2) +   
 stat\_ecdf(aes(x = PCDDF\_sample, colour="Random draw"), size=1.5, geom="point") +   
 geom\_vline(aes(xintercept=7, colour="NOAEL 7 pg/g fat"), size=1)+  
 coord\_cartesian(xlim=c(0.1,10))+  
 scale\_color\_manual(values=c(  
 `Expected value and 5-95% CI`=light,  
 `Random draw`=dark,  
 `NOAEL 7 pg/g fat`=ruby),  
 breaks=c("Expected value and 5-95% CI","Random draw","NOAEL 7 pg/g fat"),  
 guide = guide\_legend(title=""))+  
 scale\_x\_log10()+  
 theme\_thl(base\_size=18, legend.position="bottom", base\_family = "source")+  
 labs(  
# title="TEQ concentration in children in Finland",  
 y="Cumulative probability",  
 x="Predicted PCDD/F TEQ concentration, pg/g fat"  
 )  
  
fig1a



ggsave("PCDDF TEQ concentration distribution.pdf", width=16/2.54, height=18/2.54)

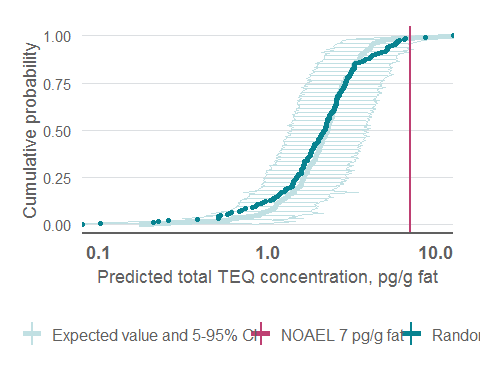
fig1adata <- fig1a$data[c("PCDDF\_P05","PCDDF\_P95","PCDDF\_pred","PCDDF\_sample","y")]  
write.csv(fig1adata, "fig1adata.csv")  
  
cat("Median values of data columns\n")

## Median values of data columns

lapply(fig1adata, median)

## $PCDDF\_P05  
## [1] 0.7903728  
##   
## $PCDDF\_P95  
## [1] 2.945338  
##   
## $PCDDF\_pred  
## [1] 1.52616  
##   
## $PCDDF\_sample  
## [1] 1.532216  
##   
## $y  
## [1] 0.5

#### Figure 2 with Total TEQ  
  
tmp$x <- seq(  
 min(tmp[c("SUM\_P05","SUM\_sample","SUM\_P95")]),  
 max(tmp[c("SUM\_P05","SUM\_sample","SUM\_P95")]),  
 length.out = nrow(tmp))  
  
lo <- ecdf(tmp$SUM\_P95)  
up <- ecdf(tmp$SUM\_P05)  
tmp$lower1 <- lo(tmp$x)  
tmp$upper1 <- up(tmp$x)  
  
tmp <- tmp[order(tmp$SUM\_pred),]  
tmp$y <- seq(0,1,1/(nrow(tmp)-1))  
  
tmp$x <- seq(  
 min(tmp[c("SUM\_P05","SUM\_sample","SUM\_P95")]),  
 max(tmp[c("SUM\_P05","SUM\_sample","SUM\_P95")]),  
 length.out = nrow(tmp))  
  
fig2a <-   
ggplot(tmp) +   
 geom\_path(aes(x=SUM\_P05, y=y), colour=light)+  
 geom\_path(aes(x=SUM\_P95, y=y), colour=light)+  
 geom\_line(data=data.frame(  
 Id = rep(tmp$Id,2),  
 Conc = c(tmp$SUM\_P05, tmp$SUM\_P95),  
 y = rep((1:nrow(tmp))/nrow(tmp),2))[c(seq(1,nrow(tmp),4),seq(1,nrow(tmp),4)+nrow(tmp)),], # Thin the CI so that you can see the lines  
 aes(x=Conc, y=y, group=Id),  
 size=0.05, colour=light)+  
 geom\_line(aes(x = SUM\_pred, y=y, colour="Expected value and 5-95% CI"), size=2) +   
 stat\_ecdf(aes(x = SUM\_sample, colour="Random draw"), size=1.5, geom="point") +   
 geom\_vline(aes(xintercept=7, colour="NOAEL 7 pg/g fat"), size=1)+  
 coord\_cartesian(xlim=c(0.1,10))+  
 scale\_color\_manual(values=c(  
 `Expected value and 5-95% CI`=light,  
 `Random draw`=dark,  
 `NOAEL 7 pg/g fat`=ruby),  
# breaks=c("Expected value and 5-95% CI","Random draw","NOAEL 7 pg/g fat"),  
 guide=guide\_legend(title=""))+  
 scale\_x\_log10()+  
 theme\_thl(legend.position="bottom", base\_size=18, base\_family = "source")+  
 labs(  
 y="Cumulative probability",  
 x="Predicted total TEQ concentration, pg/g fat"  
 )  
  
fig2adata <- fig2a$data[c("SUM\_P05","SUM\_P95","SUM\_pred","SUM\_sample","y")]  
write.csv(fig2adata, "fig2adata.csv")  
fig2a



cat("Median values of data columns\n")

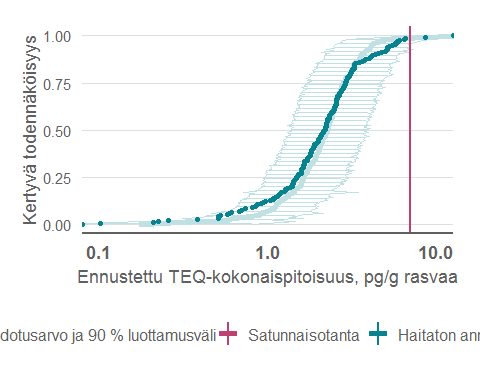
## Median values of data columns

lapply(fig2adata, median)

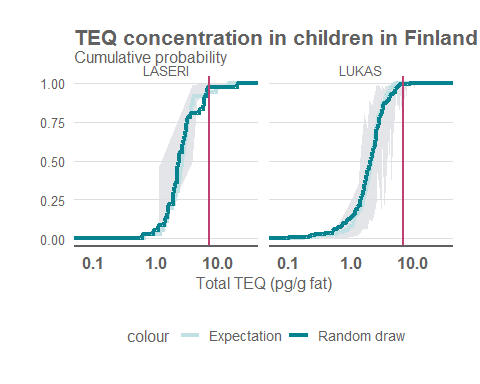
## $SUM\_P05  
## [1] 1.368774  
##   
## $SUM\_P95  
## [1] 3.829768  
##   
## $SUM\_pred  
## [1] 2.26909  
##   
## $SUM\_sample  
## [1] 2.178446  
##   
## $y  
## [1] 0.5

ggsave("Total TEQ concentration distribution.pdf", width=16/2.54, height=18/2.54)  
  
fig2a + labs(  
 y="Kertyvä todennäköisyys",  
 x="Ennustettu TEQ-kokonaispitoisuus, pg/g rasvaa"  
)+  
scale\_color\_manual(values=c(  
 `Expected value and 5-95% CI`=light,  
 `Random draw`=dark,  
 `NOAEL 7 pg/g fat`=ruby),  
 guide=guide\_legend(title=""),  
 labels=c(  
 "Odotusarvo ja 90 % luottamusväli",  
 "Satunnaisotanta",  
 "Haitaton annos 7 pg/g rasvaa"  
))

## Scale for 'colour' is already present. Adding another scale for 'colour',  
## which will replace the existing scale.

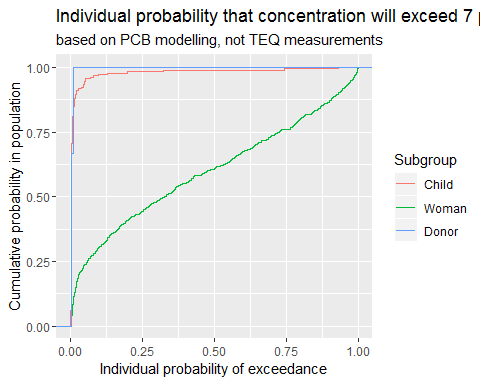


ggsave("TEQ-kokonaispitoisuusjakauma.pdf", width=16/2.54, height=18/2.54)  
  
ggplot(tmp) +   
 geom\_ribbon(aes(x=x, ymin = upper1,ymax = lower1),alpha = 0.8, fill="#dcdfe2") +  
 stat\_ecdf(aes(x = SUM\_pred, colour="Expectation"), size=1.5) +   
 stat\_ecdf(aes(x = SUM\_sample, colour="Random draw"), size=1.5) +   
 geom\_vline(xintercept=7, colour=ruby, size=1)+  
 scale\_color\_manual(values=c(light,dark))+  
 scale\_x\_log10()+  
 theme\_thl(base\_family="source", legend.position="bottom")+  
 facet\_wrap(~Cohort)+  
 labs(  
 title="TEQ concentration in children in Finland",  
 subtitle="Cumulative probability",  
 y="",  
 x="Total TEQ (pg/g fat)"  
 )



### How many individuals exceed the limit value 7 pg/g?

ggplot(out, aes(x=SUM\_p\_exceed, colour=Subgroup))+stat\_ecdf()+  
 labs(  
 title="Individual probability that concentration will exceed 7 pg/g fat",  
 subtitle="based on PCB modelling, not TEQ measurements",  
 y="Cumulative probability in population",  
 x="Individual probability of exceedance")



cat("PCDD/F TEQ values (expected value) exceeding 7 pg/g\n")

## PCDD/F TEQ values (expected value) exceeding 7 pg/g

aggregate(out$PCDDF\_pred, by = out["Subgroup"], FUN= function(x) mean(x>=7))

## Subgroup x  
## 1 Child 0.005434783  
## 2 Woman 0.095327103  
## 3 Donor 0.000000000

cat("PCDD/F TEQ values (random draw) exceeding 7 pg/g\n")

## PCDD/F TEQ values (random draw) exceeding 7 pg/g

aggregate(out$PCDDF\_sample, by = out["Subgroup"], FUN= function(x) mean(x>=7))

## Subgroup x  
## 1 Child 0.01086957  
## 2 Woman 0.19065421  
## 3 Donor 0.00000000

cat("Total TEQ values (expected value) exceeding 7 pg/g\n")

## Total TEQ values (expected value) exceeding 7 pg/g

aggregate(out$SUM\_pred, by = out["Subgroup"], FUN= function(x) mean(x>=7))

## Subgroup x  
## 1 Child 0.01086957  
## 2 Woman 0.38878505  
## 3 Donor 0.00000000

cat("Total TEQ values (random draw) exceeding 7 pg/g\n")

## Total TEQ values (random draw) exceeding 7 pg/g

aggregate(out$SUM\_sample, by = out["Subgroup"], FUN= function(x) mean(x>=7))

## Subgroup x  
## 1 Child 0.01086957  
## 2 Woman 0.37383178  
## 3 Donor 0.00000000

exceedance <- apply(exp(samps.j$teq)>7, MARGIN=c("Subgroup","TEQ"), FUN=mean)  
aggregate(exceedance, by=list(rownames(exceedance)), FUN=mean)

## Group.1 PCDDF\_TEQ\_pred PCB\_TEQ\_pred Total\_TEQ\_pred  
## 1 Child 0.0068070652 0.004179348 0.01900000  
## 2 Child\_pred 0.0065489130 0.004190217 0.01865761  
## 3 Donor 0.0000000000 0.000000000 0.00000000  
## 4 Donor\_pred 0.0006666667 0.000000000 0.00450000  
## 5 Woman 0.1682242991 0.035514019 0.37943925  
## 6 Woman\_pred 0.1679551402 0.025231776 0.40016542

cat("Fraction of children whose 95% fractile exceeds 7 pg/g PCDD/F TEQ")

## Fraction of children whose 95% fractile exceeds 7 pg/g PCDD/F TEQ

mean(fig1adata$PCDDF\_P95>7)

## [1] 0.02173913

cat("Fraction of children whose 95% fractile exceeds 7 pg/g SUM TEQ")

## Fraction of children whose 95% fractile exceeds 7 pg/g SUM TEQ

mean(fig2adata$SUM\_P95>7)

## [1] 0.04891304

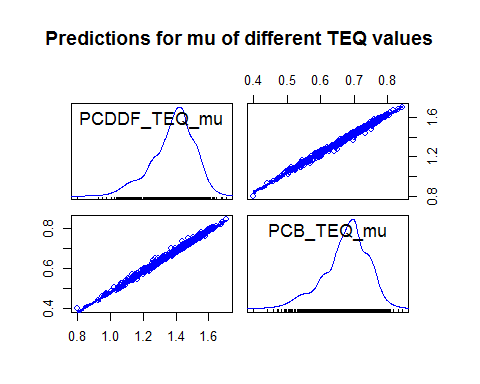
Two individual children had it more likely than not to actually exceed the total TEQ concentration of 7 pg/g fat, estimated based on their PCB concentrations.

Table. Probability of exceeding limit value of 7 pg/g TEQ based on random draws from individual probability distributions (pred) or data (Donor and Woman).

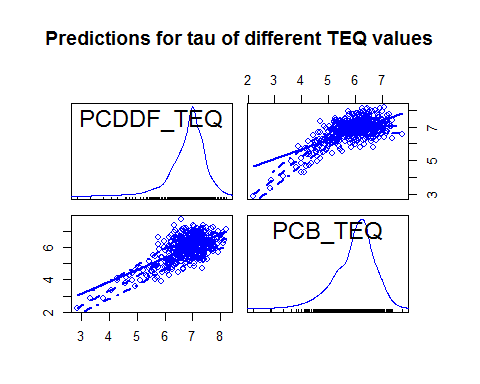
The best estimates of Total TEQs of children practically never exceed the limit of 7 pg/g fat. However, if we look at a random draw from the concentration distribution, about 3 % of children and 40 % of women exceed the limit value. With the 90 % confidence intervals of individual TEQ estimates, one tenth of children have 5 % chance that their concentration actually exceeds the limit value. Note that this does NOT mean that there is 5 % change that one tenth of children exceed the limit value.

### Statistical testing of result

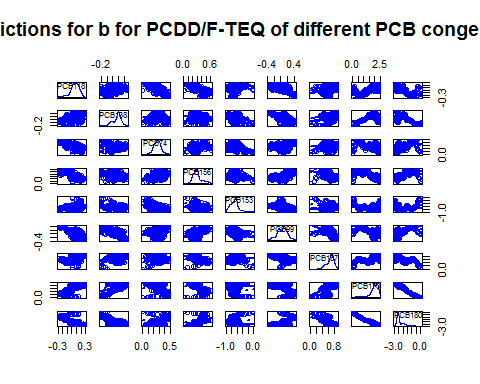
scatterplotMatrix(((samps.j$mu[,,1,1])), main = "Predictions for mu of different TEQ values")



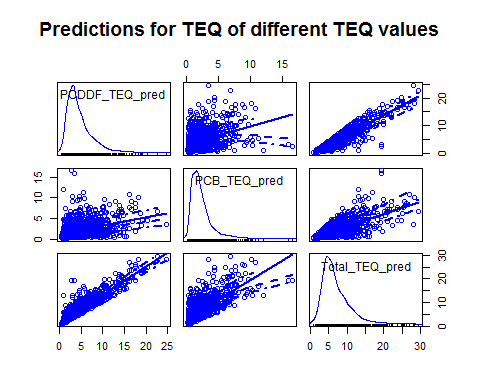
scatterplotMatrix(t((samps.j$tau[,,1])), main = "Predictions for tau of different TEQ values")



scatterplotMatrix(t(samps.j$b[1,,,1]), main = "Predictions for b for PCDD/F-TEQ of different PCB congeners")



scatterplotMatrix(exp(samps.j$teq[,,1,1]), main="Predictions for TEQ of different TEQ values")



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

