```
#############################################################
#####    Probabilistic estimates of Daily Intake    #######
#####       Using LitStream data                #######
#############################################################
#####                                  #######
#####    This R code is only for BCEP             #######
#####                                  #######
#############################################################

#########################################################
### Author: Dr. Wei-Wen Hsu          ##############
### University of Cincinnati         ##############
### Division of Biostatistics and Bioinformatics ###
### hsuwe@uc.edu                     ##############
### Latest modification: 2/27/2024     #####
#############################################
#### Updated the FUE values     ##########
#############################################

rm(list=ls(all=TRUE))

library(dplyr)
library(tidyr)
library(openxlsx)
library("writexl")
library("readxl")
library(basictabler)
library(psych)
library(goft)
```

```
library(ggplot2)

#install.packages("goft")

#install.packages("EnvStats")

library(EnvStats);


################################################


## Location for the results

path_results ="D:\\UC-OneDrive\\OneDrive - University of Cincinnati\\Desktop\\F\\Dr. Lynne - biomonitoring task\\Results";

#path_results ="C:\\Users\\hsuwe\\OneDrive - University of Cincinnati\\Desktop\\F\\Dr. Lynne - biomonitoring task\\Results";


## As the results in the file named:

exceldata_name = "\\4. LitStream_Distributional_data_results.xlsx";



############################

####  FUE parameters  ####

############################

# Old values

# c(best estimate, high, low)

#FUE_BCEP = c(0.2, 0.41, 0.04)

#FUE_BCIPP = c(0.39, 0.78, 0.08)

#FUE_BDCIPP = c(0.42, 0.5, 0.08)

#FUE_BCIPP_BCIPHIPP = c(0.78, 0.94, 0.16)


## New values as of 2/24/2024

FUE_BCEP   = c(0.13, 0.27, 0.03)
```

```
FUE_BCIPP  = c(0.23, 0.47, 0.05)
FUE_BDCIPP = c(0.23, 0.28, 0.05)
FUE_BCIPP_BCIPHIPP = c(0.58, 0.90, 0.12)



############################
#### UFRBW parameters   ####
############################
# From NHANES 2017-2018 data
# NOTE: the position of the value is important

groups    = c("3-5 yrs","6-11 yrs","12-17 yrs","18+ yrs")
UFRBW_GM  = c(0.063917438,0.024195805,0.016092702,0.01293934)
GSD       = c(2.639638606,2.241928384,2.129355164,2.072937724)

parameters_UFRBW = as.data.frame(cbind(groups, UFRBW_GM,GSD) )
parameters_UFRBW$UFRBW_GM = as.numeric(parameters_UFRBW$UFRBW_GM)
parameters_UFRBW$GSD = as.numeric(parameters_UFRBW$GSD)



########################################
## Estimated Chemicals from LitStream ##
## By age groups                     ##
########################################
# NOTE: the position of value should be in this order : c("6-11 years", "20-39 years")

############################
#### Use the following concentration levels from the studies for each age group  ####
```

```
## Study: "379-Percy et al-2022", "379-Percy et al-2022" and "382-Yang et al-2023"

age_groups =c("3 years","5 years", "20-39 years");

GM_conc_BCEP  = c(1  , 0.74, 0.51)
GSD_conc_BCEP = c(4.6, 3.64, 4.33)

##############################################################################################
####

#install.packages("EnvStats")
library(EnvStats); # a package for generating random values from a triangular distribution

##############################
### By Chemicals         ###
##############################

# a random sample of 1,000
n = 1000


## BCEP

BCEP_matrix=matrix(,1000,3); ## for 3 age-groups

yr=0;

for (agrp in c(1,1,4)){  ## select the first (3-5), 1st (3-5) and 4th (18+) age groups from NHANES 2017-18
cycle
```

```
yr=yr+1;

FUE_BCEP_rand = rtri(n, min=FUE_BCEP[[3]], max=FUE_BCEP[[2]], mode=FUE_BCEP[[1]])

UFRBW_BCEP_rand = rlnorm(n, meanlog = log(parameters_UFRBW$UFRBW_GM[[agrp]]), sdlog =
log(parameters_UFRBW$GSD[[agrp]]))

GM_conc_BCEP_rand = rlnorm(n, meanlog = log(GM_conc_BCEP[[yr]]), sdlog =
log(GSD_conc_BCEP[[yr]]))

DI_BCEP = (GM_conc_BCEP_rand * UFRBW_BCEP_rand) / FUE_BCEP_rand; ## pay attention to the unit

BCEP_matrix[,yr]=DI_BCEP

}

BCEP_matrix = as.data.frame(BCEP_matrix)

names(BCEP_matrix)= age_groups

BCEP_summary = BCEP_matrix %>%  summarise(

    mean_grp1 = (mean(BCEP_matrix[[1]], na.rm = TRUE)),

    mean_grp2 = (mean(BCEP_matrix[[2]], na.rm = TRUE)),

    mean_grp3 = (mean(BCEP_matrix[[3]], na.rm = TRUE)),


    SD_grp1 = (sd(BCEP_matrix[[1]], na.rm = TRUE)),

    SD_grp2 = (sd(BCEP_matrix[[2]], na.rm = TRUE)),

    SD_grp3 = (sd(BCEP_matrix[[3]], na.rm = TRUE)),


    Gmean_grp1 = exp(mean(log(BCEP_matrix[[1]]), na.rm = TRUE)),
```

```
    Gmean_grp2 = exp(mean(log(BCEP_matrix[[2]]), na.rm = TRUE)),
    Gmean_grp3 = exp(mean(log(BCEP_matrix[[3]]), na.rm = TRUE)),

    GSD_grp1 = exp(sd(log(BCEP_matrix[[1]]), na.rm = TRUE)),
    GSD_grp2 = exp(sd(log(BCEP_matrix[[2]]), na.rm = TRUE)),
    GSD_grp3 = exp(sd(log(BCEP_matrix[[3]]), na.rm = TRUE)),

    p25_grp1 = quantile(BCEP_matrix[[1]],0.25),
    p25_grp2 = quantile(BCEP_matrix[[2]],0.25),
    p25_grp3 = quantile(BCEP_matrix[[3]],0.25),

    p50_grp1 = quantile(BCEP_matrix[[1]],0.5),
    p50_grp2 = quantile(BCEP_matrix[[2]],0.5),
    p50_grp3 = quantile(BCEP_matrix[[3]],0.5),

    p75_grp1 = quantile(BCEP_matrix[[1]],0.75),
    p75_grp2 = quantile(BCEP_matrix[[2]],0.75),
    p75_grp3 = quantile(BCEP_matrix[[3]],0.75),

    p90_grp1 = quantile(BCEP_matrix[[1]],0.9),
    p90_grp2 = quantile(BCEP_matrix[[2]],0.9),
    p90_grp3 = quantile(BCEP_matrix[[3]],0.9),

    p95_grp1 = quantile(BCEP_matrix[[1]],0.95),
    p95_grp2 = quantile(BCEP_matrix[[2]],0.95),
    p95_grp3 = quantile(BCEP_matrix[[3]],0.95)

)
```

```
names(BCEP_summary) =
c(paste0("Mean_",age_groups),paste0("SD_",age_groups),paste0("GM_",age_groups),paste0("GSD_",age_groups),
                  paste0("p25_",age_groups),
paste0("p50_",age_groups),paste0("p75_",age_groups),paste0("p90_",age_groups),paste0("p95_",age_groups))
```

```
##########################################
#### Save the summary results     ########
##########################################

study = c("379-Percy et al-2022","379-Percy et al-2022","382-Yang et al-2023")
Age_grp = c("3 years","5 years", "20-39 years")

Mean = c(BCEP_summary[1:3])
SD  = c(BCEP_summary[4:6])
GM  = c(BCEP_summary[7:9])
GSD = c(BCEP_summary[10:12])
p25 = c(BCEP_summary[13:15])
p50 = c(BCEP_summary[16:18])
p75 = c(BCEP_summary[19:21])
p90 = c(BCEP_summary[22:24])
p95 = c(BCEP_summary[25:27])

NH_result_matrix = as.data.frame(cbind(study ,Age_grp,Mean, SD, GM, GSD, p25, p50, p75, p90, p95))
```

```
names(NH_result_matrix) = c("Study", "Age Group", "Arithmetic mean", "Arithmetic SD", "Geometric Mean", "Geometric SD", "25th percentile", "50th percentile", "75th percentile", "90th percentile", "95th percentile");

NH_result_matrix



#########################################################

### Save results as a worksheet in the Excel file  ###

#########################################################

Readme="Readme";

write.xlsx(Readme, file=paste0(path_results,exceldata_name),

      sheetName = "Readme", colNames = TRUE, rowNames = F, append = T)



################  BCEP  #################

wb <- loadWorkbook(paste0(path_results,exceldata_name));

addWorksheet(wb,"Simulated_BCEP")

writeData(wb, sheet = "Simulated_BCEP", x =BCEP_matrix, borders = "columns", startCol = 1, startRow = 1)

### graphs

BCEP_data_long <- gather(BCEP_matrix, age_groups, BCEP, age_groups, factor_key=TRUE)

g2.s = ggplot(BCEP_data_long, aes(x=BCEP, color=age_groups, fill=age_groups)) +

geom_histogram(alpha=0.6, show.legend = FALSE) +

  scale_x_continuous(name = "BCEP level") +

  scale_y_continuous(name = "Frequency")+

facet_wrap(~ age_groups, scale="free", ncol=2) +
```

```
  theme(
      axis.text.y = element_text(size=10),
      axis.title.y = element_text(size=10),
      axis.text.x = element_text(size=10),
      axis.title = element_text( size = 10, face = "bold" ),
      strip.text = element_text(size = 10),
      axis.title.x = element_text(size=10)
      )
#g2.s

plot(g2.s)
insertPlot(wb, "Simulated_BCEP", startCol = 6, startRow = 2)

### log-normal distribution test
BCEP_lnorm_test_results1 = lnorm_test(BCEP_matrix[["3 years"]])
BCEP_lnorm_test_results2 = lnorm_test(BCEP_matrix[["5 years"]])
BCEP_lnorm_test_results3 = lnorm_test(BCEP_matrix[["20-39 years"]])

BCEP_lnorm = rbind(

cbind(BCEP_lnorm_test_results1$method,BCEP_lnorm_test_results1$data.name,BCEP_lnorm_test_resu
lts1$p.value),

cbind(BCEP_lnorm_test_results2$method,BCEP_lnorm_test_results2$data.name,BCEP_lnorm_test_resu
lts2$p.value),

cbind(BCEP_lnorm_test_results3$method,BCEP_lnorm_test_results3$data.name,BCEP_lnorm_test_resu
lts3$p.value)
            )

colnames(BCEP_lnorm) = c("Method:", "Data:","p-value=")
```

```
BCEP_lnorm=transform(BCEP_lnorm)

BCEP_lnorm[[3]]=as.numeric(BCEP_lnorm[[3]])


summary_stat_BCEP= rbind(describe(BCEP_matrix[["3 years"]]),describe(BCEP_matrix[["5 years"]]),describe(BCEP_matrix[["20-39 years"]]))


summary_stat_BCEP= summary_stat_BCEP[,-1]

rownames(summary_stat_BCEP) = age_groups;


writeData(wb, sheet = "Simulated_BCEP", x =summary_stat_BCEP, rowNames = T, startCol = 14, startRow = 2)

writeData(wb, sheet = "Simulated_BCEP", x =BCEP_lnorm , startCol = 6, startRow = 22)


sheet_name = "BCEP_Estimates"


addWorksheet(wb,paste0(sheet_name))


writeData(wb, sheet = paste0(sheet_name), x ="Table. Daily intake - Estimated parameters based on probabilistic models using LitStream data", borders = "columns", startCol = 1, startRow = 1)



## Table Caption

writeData(wb, sheet = sheet_name, x = "Unit: ug/kg-day", borders = "columns", startCol = 1, startRow = 2)


row_headers = paste0(rownames(NH_result_matrix))


tbl <- BasicTable$new()

tbl$addData(NH_result_matrix, firstColumnAsRowHeaders=T,
```

```
        explicitColumnHeaders=c("Study", "Age Group", "Arithmetic mean", "Arithmetic SD", "Geometric Mean", "Geometric SD", "25th percentile", "50th percentile", "75th percentile", "90th percentile", "95th percentile"));


# the arguments are (rFrom, cFrom, rTo, cTo, declarations)

tbl$setStyling(2, 3, 16, 11, declarations=list("xl-value-format"="##0.0000"))


tbl$writeToExcelWorksheet(wb=wb, wsName=sheet_name,

                topRowNumber=4, leftMostColumnNumber=1,

                applyStyles=T, mapStylesFromCSS=F, outputValuesAs="rawValue")



saveWorkbook(wb, file=paste0(path_results,exceldata_name), overwrite = TRUE);



#################################################################################################
#######


#################################

#########  THE END   #############

#################################
```