```
##############################################################
#####     Probabilistic estimates of Daily Intake    #######
#####       Using 2017-2018 NHANES data            #######
##############################################################

######################################################
### Author: Dr. Wei-Wen Hsu           #############
### University of Cincinnati          #############
### Division of Biostatistics and Bioinformatics ###
### hsuwe@uc.edu                      #############
### Latest modification: 2/24/2024     #############
######################################################

######################################################
###   Update the FUE values | R Code               #######
######################################################



rm(list=ls(all=TRUE))


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

```r
library(ggplot2)
#install.packages("goft")
#install.packages("EnvStats")
library(EnvStats); # a package for generating random values from a triangular distribution

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

## 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 = "\\3. NHANES_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)

# 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)



############################
#### UFRBW parameters   ####
############################
# 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 NHANES ##
## By age groups                 ##
## 2017-2018 cycle                ##
#####################################
# NOTE: the position of value should be in this order : c("3-5 yrs","6-11 yrs","12-17 yrs","18+ yrs")

GM_conc_BCEP  = c(0.442551665,0.437878092,0.396797044,0.341750027)
GSD_conc_BCEP = c(2.483254634,2.385635741,2.23609415,2.305328885)

GM_conc_BCIPP  = c(0.162008359,0.157505146,0.149855623,0.137686884)
GSD_conc_BCIPP = c(2.094779969,2.101866146,1.991535006,2.047386104)
```

```
GM_conc_BDCIPP  = c(2.114038246,2.469496702,1.582861204,1.094851566)

GSD_conc_BDCIPP = c(2.319589615,2.336829698,2.177486106,2.336394394)



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

### By Chemicals         ###

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


# a random sample of 1,000

n = 1000

gg= 4; # 4 age groups


## BCEP


BCEP_matrix=matrix(,1000,gg); ## for 4 age-groups


for (yr in c(1:length(parameters_UFRBW$groups))){


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$UFRBW_GM[[yr]]), sdlog = log(parameters_UFRBW$GSD[[yr]]))


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)= parameters_UFRBW$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)),

        mean_grp4 = (mean(BCEP_matrix[[4]], 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)),

        SD_grp4 = (sd(BCEP_matrix[[4]], 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)),

        Gmean_grp4 = exp(mean(log(BCEP_matrix[[4]]), 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)),

        GSD_grp4 = exp(sd(log(BCEP_matrix[[4]]), 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),
        p25_grp4 = quantile(BCEP_matrix[[4]],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),
        p50_grp4 = quantile(BCEP_matrix[[4]],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),
        p75_grp4 = quantile(BCEP_matrix[[4]],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),
        p90_grp4 = quantile(BCEP_matrix[[4]],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),
        p95_grp4 = quantile(BCEP_matrix[[4]],0.95),
)

names(BCEP_summary) =
c(paste0("Mean_",parameters_UFRBW$groups),paste0("SD_",parameters_UFRBW$groups),paste0("GM
_",parameters_UFRBW$groups),paste0("GSD_",parameters_UFRBW$groups),
                paste0("p25_",parameters_UFRBW$groups),
paste0("p50_",parameters_UFRBW$groups),paste0("p75_",parameters_UFRBW$groups),paste0("p90_",
parameters_UFRBW$groups),paste0("p95_",parameters_UFRBW$groups))
```

```
## BDCIPP

BDCIPP_matrix=matrix(,1000,gg); ## for 4 age-groups

for (yr in c(1:length(parameters_UFRBW$groups))){

FUE_BDCIPP_rand = rtri(n, min=FUE_BDCIPP[[3]], max=FUE_BDCIPP[[2]], mode=FUE_BDCIPP[[1]])

UFRBW_BDCIPP_rand = rlnorm(n, meanlog = log(parameters_UFRBW$UFRBW_GM[[yr]]), sdlog = log(parameters_UFRBW$GSD[[yr]]))

GM_conc_BDCIPP_rand = rlnorm(n, meanlog = log(GM_conc_BDCIPP[[yr]]), sdlog = log(GSD_conc_BDCIPP[[yr]]))

DI_BDCIPP = (GM_conc_BDCIPP_rand * UFRBW_BDCIPP_rand) / FUE_BDCIPP_rand; ## pay attention to the unit

BDCIPP_matrix[,yr]=DI_BDCIPP
}


BDCIPP_matrix = as.data.frame(BDCIPP_matrix)
names(BDCIPP_matrix)= parameters_UFRBW$groups

BDCIPP_summary = BDCIPP_matrix %>%   summarise(
        mean_grp1 = (mean(BDCIPP_matrix[[1]], na.rm = TRUE)),
```

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

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

mean_grp4 = (mean(BDCIPP_matrix[[4]], na.rm = TRUE)),


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

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

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

SD_grp4 = (sd(BDCIPP_matrix[[4]], na.rm = TRUE)),


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

Gmean_grp2 = exp(mean(log(BDCIPP_matrix[[2]]), na.rm = TRUE)),

Gmean_grp3 = exp(mean(log(BDCIPP_matrix[[3]]), na.rm = TRUE)),

Gmean_grp4 = exp(mean(log(BDCIPP_matrix[[4]]), na.rm = TRUE)),


GSD_grp1 = exp(sd(log(BDCIPP_matrix[[1]]), na.rm = TRUE)),

GSD_grp2 = exp(sd(log(BDCIPP_matrix[[2]]), na.rm = TRUE)),

GSD_grp3 = exp(sd(log(BDCIPP_matrix[[3]]), na.rm = TRUE)),

GSD_grp4 = exp(sd(log(BDCIPP_matrix[[4]]), na.rm = TRUE)),


p25_grp1 = quantile(BDCIPP_matrix[[1]],0.25),

p25_grp2 = quantile(BDCIPP_matrix[[2]],0.25),

p25_grp3 = quantile(BDCIPP_matrix[[3]],0.25),

p25_grp4 = quantile(BDCIPP_matrix[[4]],0.25),


p50_grp1 = quantile(BDCIPP_matrix[[1]],0.5),

p50_grp2 = quantile(BDCIPP_matrix[[2]],0.5),

p50_grp3 = quantile(BDCIPP_matrix[[3]],0.5),

p50_grp4 = quantile(BDCIPP_matrix[[4]],0.5),
```

```
        p75_grp1 = quantile(BDCIPP_matrix[[1]],0.75),
        p75_grp2 = quantile(BDCIPP_matrix[[2]],0.75),
        p75_grp3 = quantile(BDCIPP_matrix[[3]],0.75),
        p75_grp4 = quantile(BDCIPP_matrix[[4]],0.75),

        p90_grp1 = quantile(BDCIPP_matrix[[1]],0.9),
        p90_grp2 = quantile(BDCIPP_matrix[[2]],0.9),
        p90_grp3 = quantile(BDCIPP_matrix[[3]],0.9),
        p90_grp4 = quantile(BDCIPP_matrix[[4]],0.9),

        p95_grp1 = quantile(BDCIPP_matrix[[1]],0.95),
        p95_grp2 = quantile(BDCIPP_matrix[[2]],0.95),
        p95_grp3 = quantile(BDCIPP_matrix[[3]],0.95),
        p95_grp4 = quantile(BDCIPP_matrix[[4]],0.95),
)

names(BDCIPP_summary) = c(paste0("Mean_",parameters_UFRBW$groups),paste0("SD_",parameters_UFRBW$groups),paste0("GM_",parameters_UFRBW$groups),paste0("GSD_",parameters_UFRBW$groups),
                    paste0("p25_",parameters_UFRBW$groups), paste0("p50_",parameters_UFRBW$groups),paste0("p75_",parameters_UFRBW$groups),paste0("p90_", parameters_UFRBW$groups),paste0("p95_",parameters_UFRBW$groups))





## BCIPP


BCIPP_matrix=matrix(,1000,gg); ## for age-groups
```

```
for (yr in c(1:length(parameters_UFRBW$groups))){

FUE_BCIPP_rand = rtri(n, min=FUE_BCIPP[[3]], max=FUE_BCIPP[[2]], mode=FUE_BCIPP[[1]])

UFRBW_BCIPP_rand = rlnorm(n, meanlog = log(parameters_UFRBW$UFRBW_GM[[yr]]), sdlog =
log(parameters_UFRBW$GSD[[yr]]))

GM_conc_BCIPP_rand = rlnorm(n, meanlog = log(GM_conc_BCIPP[[yr]]), sdlog =
log(GSD_conc_BCIPP[[yr]]))

DI_BCIPP = (GM_conc_BCIPP_rand * UFRBW_BCIPP_rand) / FUE_BCIPP_rand;  ## pay attention to the
unit

BCIPP_matrix[,yr]=DI_BCIPP
}

BCIPP_matrix = as.data.frame(BCIPP_matrix)
names(BCIPP_matrix)= parameters_UFRBW$groups

#hist(BCIPP_matrix[[5]])

BCIPP_summary = BCIPP_matrix %>%   summarise(

    mean_grp1 = (mean(BCIPP_matrix[[1]], na.rm = TRUE)),
    mean_grp2 = (mean(BCIPP_matrix[[2]], na.rm = TRUE)),
    mean_grp3 = (mean(BCIPP_matrix[[3]], na.rm = TRUE)),
    mean_grp4 = (mean(BCIPP_matrix[[4]], na.rm = TRUE)),

    SD_grp1 = (sd(BCIPP_matrix[[1]], na.rm = TRUE)),
    SD_grp2 = (sd(BCIPP_matrix[[2]], na.rm = TRUE)),
```

```
SD_grp3 = (sd(BCIPP_matrix[[3]], na.rm = TRUE)),
SD_grp4 = (sd(BCIPP_matrix[[4]], na.rm = TRUE)),

Gmean_grp1 = exp(mean(log(BCIPP_matrix[[1]]), na.rm = TRUE)),
Gmean_grp2 = exp(mean(log(BCIPP_matrix[[2]]), na.rm = TRUE)),
Gmean_grp3 = exp(mean(log(BCIPP_matrix[[3]]), na.rm = TRUE)),
Gmean_grp4 = exp(mean(log(BCIPP_matrix[[4]]), na.rm = TRUE)),

GSD_grp1 = exp(sd(log(BCIPP_matrix[[1]]), na.rm = TRUE)),
GSD_grp2 = exp(sd(log(BCIPP_matrix[[2]]), na.rm = TRUE)),
GSD_grp3 = exp(sd(log(BCIPP_matrix[[3]]), na.rm = TRUE)),
GSD_grp4 = exp(sd(log(BCIPP_matrix[[4]]), na.rm = TRUE)),

p25_grp1 = quantile(BCIPP_matrix[[1]],0.25),
p25_grp2 = quantile(BCIPP_matrix[[2]],0.25),
p25_grp3 = quantile(BCIPP_matrix[[3]],0.25),
p25_grp4 = quantile(BCIPP_matrix[[4]],0.25),

p50_grp1 = quantile(BCIPP_matrix[[1]],0.5),
p50_grp2 = quantile(BCIPP_matrix[[2]],0.5),
p50_grp3 = quantile(BCIPP_matrix[[3]],0.5),
p50_grp4 = quantile(BCIPP_matrix[[4]],0.5),

p75_grp1 = quantile(BCIPP_matrix[[1]],0.75),
p75_grp2 = quantile(BCIPP_matrix[[2]],0.75),
p75_grp3 = quantile(BCIPP_matrix[[3]],0.75),
p75_grp4 = quantile(BCIPP_matrix[[4]],0.75),

p90_grp1 = quantile(BCIPP_matrix[[1]],0.9),
```

```
        p90_grp2 = quantile(BCIPP_matrix[[2]],0.9),
        p90_grp3 = quantile(BCIPP_matrix[[3]],0.9),
        p90_grp4 = quantile(BCIPP_matrix[[4]],0.9),

        p95_grp1 = quantile(BCIPP_matrix[[1]],0.95),
        p95_grp2 = quantile(BCIPP_matrix[[2]],0.95),
        p95_grp3 = quantile(BCIPP_matrix[[3]],0.95),
        p95_grp4 = quantile(BCIPP_matrix[[4]],0.95)
)

names(BCIPP_summary) = c(paste0("Mean_",parameters_UFRBW$groups),paste0("SD_",parameters_UFRBW$groups),paste0("GM_",parameters_UFRBW$groups),paste0("GSD_",parameters_UFRBW$groups),
                paste0("p25_",parameters_UFRBW$groups), paste0("p50_",parameters_UFRBW$groups),paste0("p75_",parameters_UFRBW$groups),paste0("p90_", parameters_UFRBW$groups),paste0("p95_",parameters_UFRBW$groups))


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

Biomarker = c(rep("BCEP",gg),rep("BDCIPP",gg),rep("BCIPP",gg))
Age_grp = rep(parameters_UFRBW$groups,3)
Mean = c(BCEP_summary[1:4] ,BDCIPP_summary[1:4],BCIPP_summary[1:4])
SD  = c(BCEP_summary[5:8]  ,BDCIPP_summary[5:8],BCIPP_summary[5:8])
GM  = c(BCEP_summary[9:12] ,BDCIPP_summary[9:12] ,BCIPP_summary[9:12])
GSD = c(BCEP_summary[13:16],BDCIPP_summary[13:16],BCIPP_summary[13:16])
p25 = c(BCEP_summary[17:20],BDCIPP_summary[17:20],BCIPP_summary[17:20])
```

```
p50 = c(BCEP_summary[21:24],BDCIPP_summary[21:24],BCIPP_summary[21:24])

p75 = c(BCEP_summary[25:28],BDCIPP_summary[25:28],BCIPP_summary[25:28])

p90 = c(BCEP_summary[29:32],BDCIPP_summary[29:32],BCIPP_summary[29:32])

p95 = c(BCEP_summary[33:36],BDCIPP_summary[33:36],BCIPP_summary[33:36])


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


names(NH_result_matrix) = c("Biomarker", "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, c("3-5 yrs", "6-11 yrs", "12-17 yrs","18+ yrs"), 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[["3-5 yrs"]])
BCEP_lnorm_test_results2 = lnorm_test(BCEP_matrix[["6-11 yrs"]])
BCEP_lnorm_test_results3 = lnorm_test(BCEP_matrix[["12-17 yrs"]])
BCEP_lnorm_test_results4 = lnorm_test(BCEP_matrix[["18+ yrs"]])


BCEP_lnorm = rbind(
```

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

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

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

cbind(BCEP_lnorm_test_results4$method,BCEP_lnorm_test_results4$data.name,BCEP_lnorm_test_results4$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[["3-5 yrs"]]),describe(BCEP_matrix[["6-11 yrs"]]),
    describe(BCEP_matrix[["12-17 yrs"]]),describe(BCEP_matrix[["18+ yrs"]]))


summary_stat_BCEP= summary_stat_BCEP[,-1]
rownames(summary_stat_BCEP) = 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)


##################   BDCIPP  #################


addWorksheet(wb,"Simulated_BDCIPP")
```

```
writeData(wb, sheet = "Simulated_BDCIPP", x =BDCIPP_matrix, borders = "columns", startCol = 1, startRow = 1)


### graphs
BDCIPP_data_long <- gather(BDCIPP_matrix, age_groups, BDCIPP, c("3-5 yrs", "6-11 yrs", "12-17 yrs","18+ yrs"), factor_key=TRUE)
g2.s = ggplot(BDCIPP_data_long, aes(x=BDCIPP, color=age_groups, fill=age_groups)) +
geom_histogram(alpha=0.6, show.legend = FALSE) +
  scale_x_continuous(name = "BDCIPP 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_BDCIPP", startCol = 6, startRow = 2)


### log-normal distribution test
BDCIPP_lnorm_test_results1 = lnorm_test(BDCIPP_matrix[["3-5 yrs"]])
BDCIPP_lnorm_test_results2 = lnorm_test(BDCIPP_matrix[["6-11 yrs"]])
BDCIPP_lnorm_test_results3 = lnorm_test(BDCIPP_matrix[["12-17 yrs"]])
BDCIPP_lnorm_test_results4 = lnorm_test(BDCIPP_matrix[["18+ yrs"]])
```

```
BDCIPP_lnorm = rbind(

cbind(BDCIPP_lnorm_test_results1$method,BDCIPP_lnorm_test_results1$data.name,BDCIPP_lnorm_test_results1$p.value),

cbind(BDCIPP_lnorm_test_results2$method,BDCIPP_lnorm_test_results2$data.name,BDCIPP_lnorm_test_results2$p.value),

cbind(BDCIPP_lnorm_test_results3$method,BDCIPP_lnorm_test_results3$data.name,BDCIPP_lnorm_test_results3$p.value),

cbind(BDCIPP_lnorm_test_results4$method,BDCIPP_lnorm_test_results4$data.name,BDCIPP_lnorm_test_results4$p.value) )


colnames(BDCIPP_lnorm) = c("Method:", "Data:","p-value=")
BDCIPP_lnorm=transform(BDCIPP_lnorm)
BDCIPP_lnorm[[3]]=as.numeric(BDCIPP_lnorm[[3]])


summary_stat_BDCIPP = rbind(describe(BDCIPP_matrix[["3-5 yrs"]]),describe(BDCIPP_matrix[["6-11 yrs"]]),
    describe(BDCIPP_matrix[["12-17 yrs"]]),describe(BDCIPP_matrix[["18+ yrs"]]))


summary_stat_BDCIPP = summary_stat_BDCIPP[,-1]
rownames(summary_stat_BDCIPP) = groups;


writeData(wb, sheet = "Simulated_BDCIPP", x =summary_stat_BDCIPP, rowNames = T , startCol = 14, startRow = 2)
writeData(wb, sheet = "Simulated_BDCIPP", x =BDCIPP_lnorm , startCol = 6, startRow = 22)



#############    BCIPP    ########################
```

```
addWorksheet(wb,"Simulated_BCIPP")


writeData(wb, sheet = "Simulated_BCIPP", x =BCIPP_matrix, borders = "columns", startCol = 1, startRow = 1)


### graphs
BCIPP_data_long <- gather(BCIPP_matrix, age_groups, BCIPP, c("3-5 yrs", "6-11 yrs", "12-17 yrs","18+ yrs"), factor_key=TRUE)
g2.s = ggplot(BCIPP_data_long, aes(x=BCIPP, color=age_groups, fill=age_groups)) +
geom_histogram(alpha=0.6, show.legend = FALSE) +
  scale_x_continuous(name = "BCIPP 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_BCIPP", startCol = 6, startRow = 2)


### log-normal distribution test
```

```
BCIPP_lnorm_test_results1 = lnorm_test(BCIPP_matrix[["3-5 yrs"]])
BCIPP_lnorm_test_results2 = lnorm_test(BCIPP_matrix[["6-11 yrs"]])
BCIPP_lnorm_test_results3 = lnorm_test(BCIPP_matrix[["12-17 yrs"]])
BCIPP_lnorm_test_results4 = lnorm_test(BCIPP_matrix[["18+ yrs"]])

BCIPP_lnorm = rbind(

cbind(BCIPP_lnorm_test_results1$method,BCIPP_lnorm_test_results1$data.name,BCIPP_lnorm_test_results1$p.value),

cbind(BCIPP_lnorm_test_results2$method,BCIPP_lnorm_test_results2$data.name,BCIPP_lnorm_test_results2$p.value),

cbind(BCIPP_lnorm_test_results3$method,BCIPP_lnorm_test_results3$data.name,BCIPP_lnorm_test_results3$p.value),

cbind(BCIPP_lnorm_test_results4$method,BCIPP_lnorm_test_results4$data.name,BCIPP_lnorm_test_results4$p.value) )

colnames(BCIPP_lnorm) = c("Method:", "Data:","p-value=")
BCIPP_lnorm=transform(BCIPP_lnorm)
BCIPP_lnorm[[3]]=as.numeric(BCIPP_lnorm[[3]])

summary_stat_BCIPP= rbind(describe(BCIPP_matrix[["3-5 yrs"]]),describe(BCIPP_matrix[["6-11 yrs"]]),
    describe(BCIPP_matrix[["12-17 yrs"]]),describe(BCIPP_matrix[["18+ yrs"]]))

summary_stat_BCIPP= summary_stat_BCIPP[,-1]
rownames(summary_stat_BCIPP) = groups;

writeData(wb, sheet = "Simulated_BCIPP", x =summary_stat_BCIPP, rowNames = T, startCol = 14, startRow = 2)
writeData(wb, sheet = "Simulated_BCIPP", x =BCIPP_lnorm , startCol = 6, startRow = 22)
```

```
##################################
###### Final estimates  ##########
##################################



sheet_name = "Prob_Estimates_from_NHANES"


addWorksheet(wb,paste0(sheet_name))


writeData(wb, sheet = paste0(sheet_name), x ="Table. Estimated parameters based on probabilistic
models using NHANES 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("Biomarker", "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   ############
#################################
```