PFHxS 1 compartment Plots (v8)

Weihsueh Chiu, Claire Lay, Parker Malek

2022-01-23

library(coda)  
library(bayesplot)

## This is bayesplot version 1.8.0

## - Online documentation and vignettes at mc-stan.org/bayesplot

## - bayesplot theme set to bayesplot::theme\_default()

## \* Does \_not\_ affect other ggplot2 plots

## \* See ?bayesplot\_theme\_set for details on theme setting

library(ggplot2)  
library(ggsci)  
library(khroma)  
library(tidyverse)

## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.1 ──

## ✓ tibble 3.1.6 ✓ dplyr 1.0.7  
## ✓ tidyr 1.1.4 ✓ stringr 1.4.0  
## ✓ readr 2.1.1 ✓ forcats 0.5.1  
## ✓ purrr 0.3.4

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

library(reshape2)

##   
## Attaching package: 'reshape2'

## The following object is masked from 'package:tidyr':  
##   
## smiths

library(yardstick) # for quick rmse

## For binary classification, the first factor level is assumed to be the event.  
## Use the argument `event\_level = "second"` to alter this as needed.

##   
## Attaching package: 'yardstick'

## The following object is masked from 'package:readr':  
##   
## spec

library(here)

## here() starts at /media/projects/projects/PFAS\_PBPK/user/weihsueh\_2021/PFAS\_1cpt\_v8-main/PFHxS\_1cpt\_v8

knitr::opts\_chunk$set(echo = TRUE, dpi = 300 )

Set up MCSim file

# this markdown file must be saved in top level directory for the following to work; the mcsim code depends on getwd results.  
mdir <- "MCSim"  
source(here::here(mdir,"setup\_MCSim.R"))  
# Make mod.exe (used to create mcsim executable from model file)  
makemod()

## The mod.exe had been created.

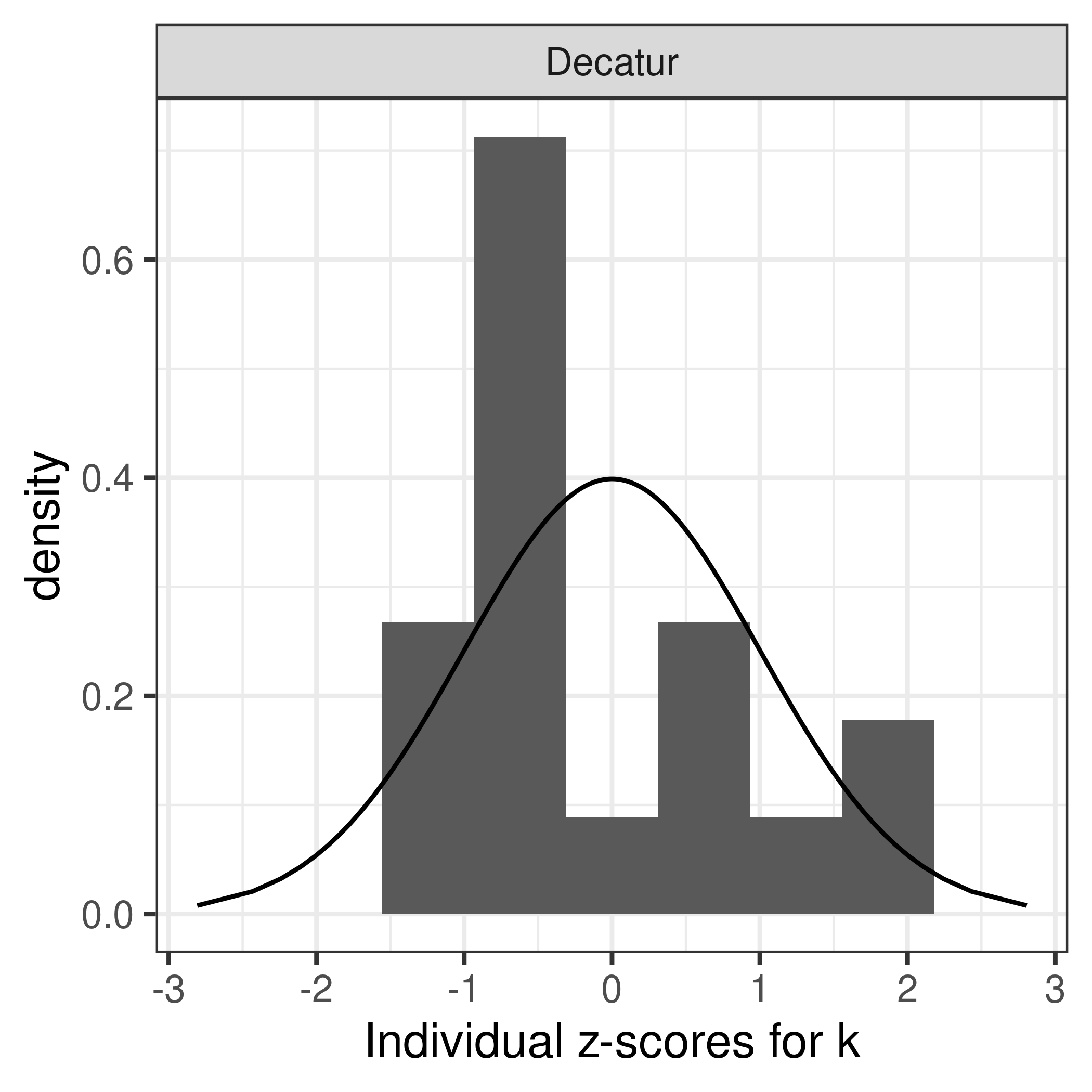
## Set filenames and load data

## Set up dataset

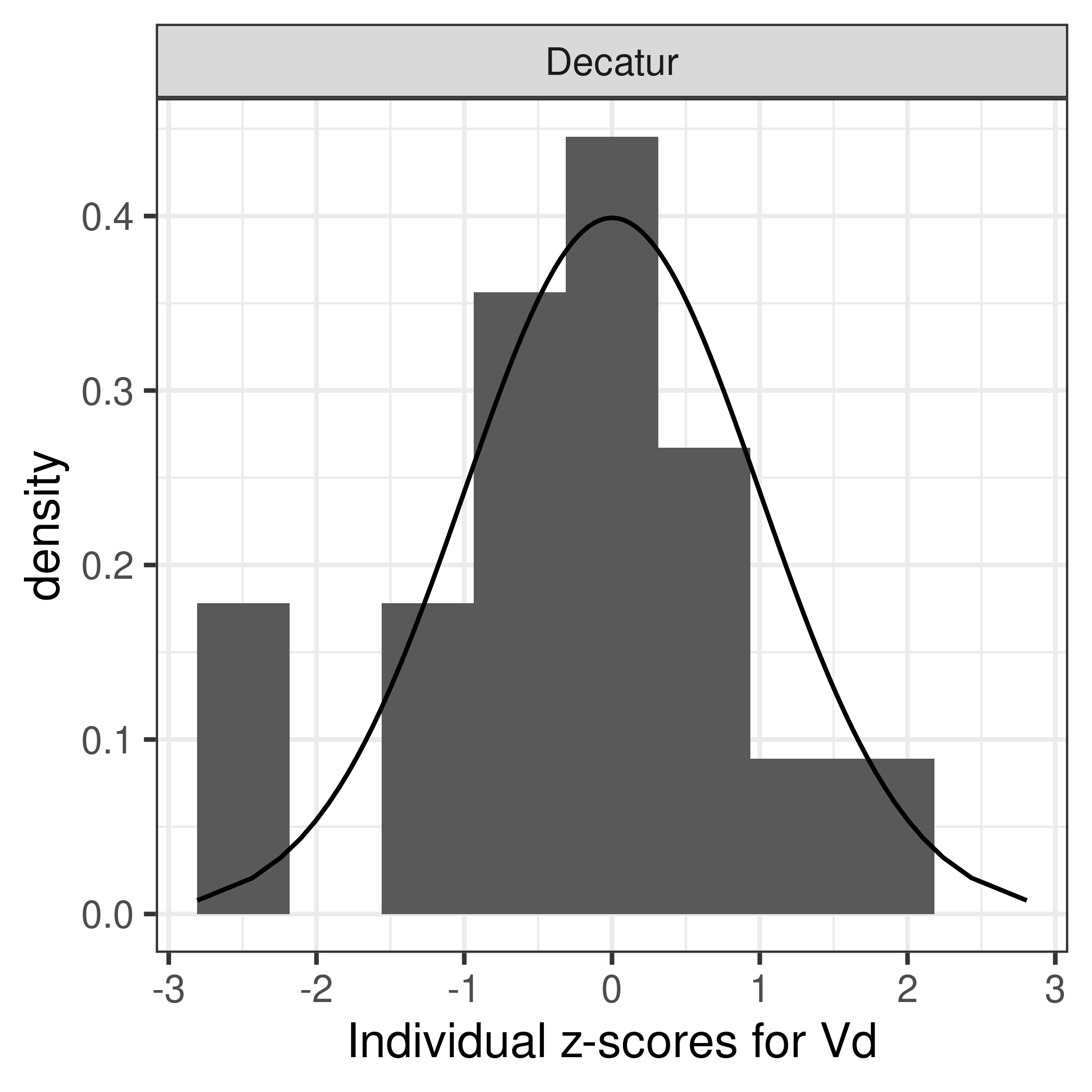
id\_lut <- multicheck$df\_check %>% select(Level) %>% unique () %>%  
 mutate(dataset = c(   
 rep("Decatur M Train", 9),  
 rep("Decatur F Train", 9),  
 rep("Decatur M Test", 9),  
 rep("Decatur F Test", 10),  
 'Paulsboro-Train','Horsham-Train',  
 'Warminster-Test','Warrington-Train'),   
 Sex = c(   
 rep("M", 9),  
 rep("F", 9),  
 rep("M", 9),  
 rep("F", 10),  
 'Mixed', 'Mixed', 'Mixed', 'Mixed'),  
 City = c(   
 rep("Decatur", 18),  
 rep("Decatur", 19),  
 'Paulsboro','Horsham','Warminster','Warrington'),   
 Train\_Test = c(   
 rep("Train", 9),  
 rep("Train", 9),  
 rep("Test", 9),  
 rep("Test", 10),  
 'Train', 'Train', 'Test', 'Test'),  
 datatype = c(  
 rep("Individual",9+9+9+10),  
 rep("Summary",4)),  
 Simulation = row\_number(),  
 variable = paste0(dataset, " ",Simulation))  
  
id\_lut$dataset <- factor(id\_lut$dataset,levels=  
 c("Decatur M Train","Decatur F Train","Arnsberg M Train",  
 "Arnsberg F Train","Decatur M Test","Decatur F Test","Arnsberg M Test",  
 "Arnsberg F Test","Minnesota Train","Minnesota Test",  
 'Lubeck-Bartell-Train', 'Lubeck-Bartell-Test',  
 'Little Hocking-Bartell-Train', 'Little Hocking-Bartell-Test',  
 'Little Hocking-Emmett-Test','Paulsboro-Train','Horsham-Train',  
 'Warminster-Test','Warrington-Train'))  
id\_lut$City <- factor(id\_lut$City,levels =   
 c("Decatur","Arnsberg","Minnesota",'Lubeck-Bartell',  
 'Little Hocking-Bartell','Little Hocking-Emmett',  
 'Paulsboro','Horsham','Warminster','Warrington'))  
   
  
indiv\_lut <- id\_lut %>%   
 filter(City %in% c("Decatur")) %>%  
 mutate( dataset = as.factor(dataset))  
  
nv <- data.frame(dataset =unique(indiv\_lut$dataset),   
 variable= rep("Pop GM", 4),  
 type= rep("Pop GM", 4), stringsAsFactors = FALSE)

## Individual parameters

set.seed(314159)  
  
indiv\_parms <- indiv\_lut  
lnkparmnames <- paste("ln\_k.",gsub("\_",".",indiv\_parms$Level),".",sep="")  
lnVdparmnames <- paste("ln\_Vd.",gsub("\_",".",indiv\_parms$Level),".",sep="")  
  
parmsamp <- apply(multicheck$parms.samp,2,sample,1)  
  
## Random z-score estimate of each parameter  
indiv\_parms$ln\_k.z.samp <- parmsamp[lnkparmnames]  
indiv\_parms$ln\_Vd.z.samp <- parmsamp[lnVdparmnames]  
  
normd <- data.frame(x=qnorm(ppoints(200)))  
normd$y <- dnorm(normd$x)  
  
iplotk<-  
 ggplot(subset(indiv\_parms,Train\_Test=="Train"))+  
 geom\_histogram(aes(x=ln\_k.z.samp,after\_stat(density)),bins=10)+facet\_wrap(~City,ncol=1)+  
 geom\_line(aes(x=x,y=y),data=normd)+  
 xlab("Individual z-scores for k") + theme\_bw()  
  
iplotVd<-  
 ggplot(subset(indiv\_parms,Train\_Test=="Train"))+  
 geom\_histogram(aes(x=ln\_Vd.z.samp,after\_stat(density)),bins=10)+facet\_wrap(~City,ncol=1)+  
 geom\_line(aes(x=x,y=y),data=normd)+  
 xlab("Individual z-scores for Vd") + theme\_bw()  
  
print(iplotk)



print(iplotVd)



ggsave(file.path("output-plots",  
 paste0( sa,"Indiv\_zscores\_k\_PFHxS.pdf")),iplotk,dpi=600)

## Saving 3.5 x 3.5 in image

ggsave(file.path("output-plots",  
 paste0( sa,"Indiv\_zscores\_k\_PFHxS.png")),iplotk,dpi=600)

## Saving 3.5 x 3.5 in image

ggsave(file.path("output-plots",  
 paste0( sa,"Indiv\_zscores\_Vd\_PFHxS.pdf")),iplotVd,dpi=600)

## Saving 3.5 x 3.5 in image

ggsave(file.path("output-plots",  
 paste0( sa,"Indiv\_zscores\_Vd\_PFHxS.png")),iplotVd,dpi=600)

## Saving 3.5 x 3.5 in image

## Scatter plot of predictions (median of multicheck samples) versus data.

This is a Figure 2 panel. Needed to use “scale=1.1” in ggsave to match PFOA.

nrow(multicheck$df\_check)

## [1] 39000

nrow(id\_lut)

## [1] 41

multicheck$df\_check %>% left\_join(id\_lut) %>% nrow()

## Joining, by = c("Level", "Simulation")

## [1] 39000

names(multicheck$df\_check)

## [1] "Level" "Simulation" "Output\_Var" "Time" "Data"   
## [6] "Prediction"

multicheck2 <- multicheck$df\_check %>%   
 left\_join(id\_lut, by = c("Level", "Simulation"))%>%   
 group\_by\_at ( vars(-Prediction)) %>%   
 summarise(Prediction = median(Prediction)) %>%  
 ungroup() %>%  
 group\_by(City) %>%   
 mutate(Train\_Test = factor(Train\_Test, levels = c("Train", "Test")),  
 `City (datatype)` = factor (paste0("\n", City, "\n(", datatype, ")\n") ),  
 label = case\_when(Train\_Test=="Train" ~ "G: PFHxS Train",  
 Train\_Test=="Test" ~ "H: PFHxS Test",  
 TRUE ~ ""))

## `summarise()` has grouped output by 'Level', 'Simulation', 'Output\_Var', 'Time', 'Data', 'dataset', 'Sex', 'City', 'Train\_Test', 'datatype'. You can override using the `.groups` argument.

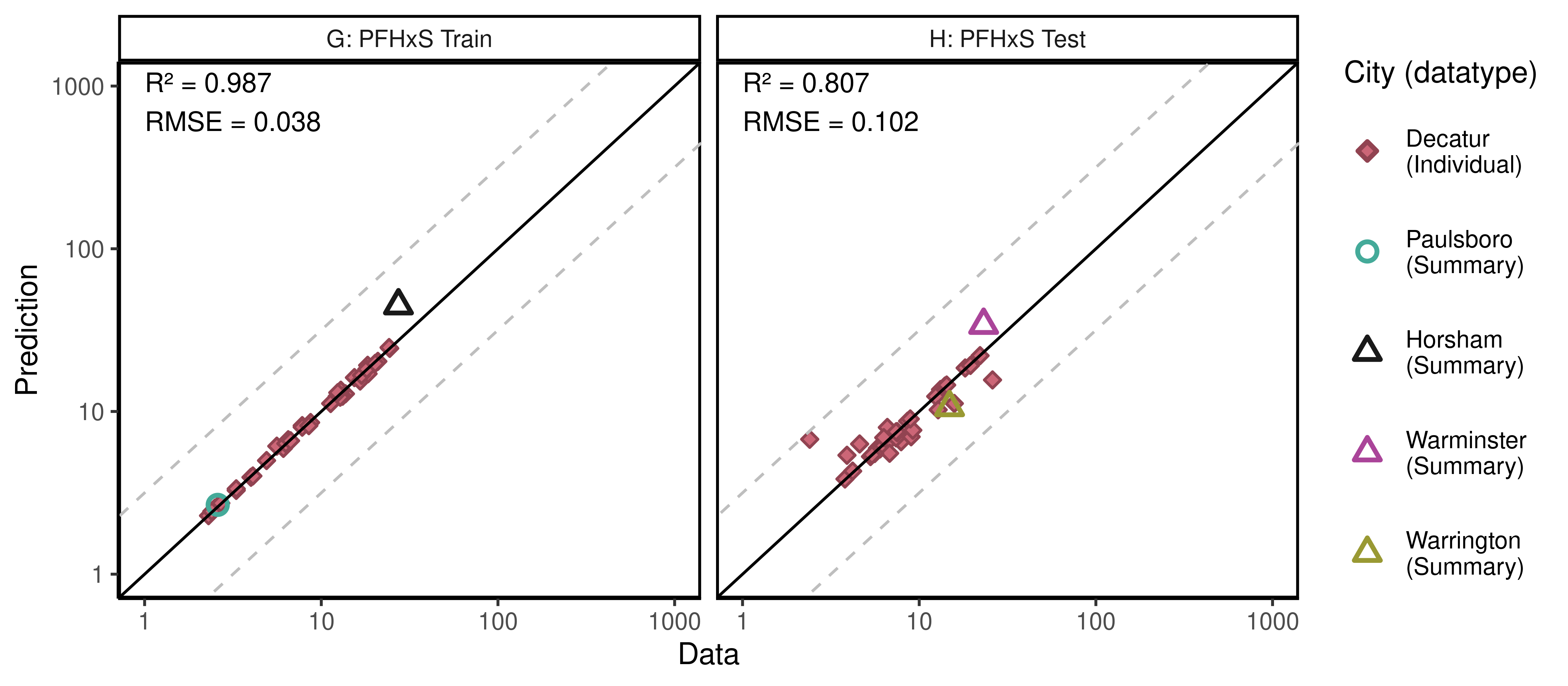
#define color for testing boxplots  
bp\_cols <- c (as.character (khroma::colour("muted")(9)) , "#191919")   
bp\_cols <-bp\_cols[c(1,7,10:8)]# plot\_scheme\_colourblind(bp\_cols)   
  
### Create aesthetics lookup  
aes\_lut <- multicheck2 %>% ungroup() %>%   
 group\_by(City, datatype, `City (datatype)` ) %>% summarise () %>% ungroup() %>%  
 mutate( cols = bp\_cols, city\_fills = bp\_cols ,   
 # for individual level on point plot (multicheck2), darken outlines for visibility, use standard colors otherwise  
 city\_outlines = if\_else(datatype == "Individual" , colorspace::darken(city\_fills, 0.3), city\_fills) ,   
 shapes = case\_when(datatype == "Individual" & `City` %in% c('Decatur', 'Arnsberg', 'Minnesota') ~ 23,  
 datatype == "Summary" &`City` %in% c("Horsham", "Warminster", "Warrington") ~ 2,  
 datatype == "Summary" & `City` == "Paulsboro" ~ 1,  
 TRUE ~ 18 ),   
 size = if\_else(datatype =="Individual", 1.75, 2.5 ) )

## `summarise()` has grouped output by 'City', 'datatype'. You can override using the `.groups` argument.

source( paste0(gsub(basename(here()), 'shared\_functions', here()), '/plot\_scatter\_mcheck.r'))  
  
p2 <- plot\_scatter\_mcheck(dframe = multicheck2, pfas\_nom = pfas\_name, aes\_lut\_fn = aes\_lut )

## Joining, by = "facet\_label"

print(p2)



ggsave(here ("output-plots", paste0( sa,"multicheckplot\_", pfas\_name,  
 ".pdf")),p2,dpi=600, scale=1.1)

## Saving 8.8 x 3.85 in image

ggsave(here ("output-plots", paste0( sa,"multicheckplot\_", pfas\_name,  
 ".png")),p2,dpi=600, scale=1.1)

## Saving 8.8 x 3.85 in image

multicheck2$PFAS <- "PFHxS"  
multiplot\_scatter\_dat <- list(aes\_lut = aes\_lut, multicheck2 = multicheck2) # save out for use in commbined plot  
save(multiplot\_scatter\_dat, file = "pfhxs\_multiplot\_scatter.Rdata")

## Parse multicheck

df\_check <- multicheck$df\_check  
df\_check <- subset(df\_check,Data > 0)   
  
n1 <- nrow(df\_check)  
id\_chks <- df\_check %>% select(Level) %>% unique() %>% bind\_cols(id\_lut) %>%  
 mutate(dataset = as.factor(dataset), Sex = as.factor(Sex), City = as.factor(City),   
 Train\_Test = as.factor(Train\_Test))

## New names:  
## \* Level -> Level...1  
## \* Level -> Level...2

df\_check <- df\_check %>% left\_join(id\_chks)%>%  
 mutate(Dataset = paste(as.character(dataset), Simulation),  
 Sex = ordered(Sex, levels = c("M", "F", "Mixed"),   
 labels = c("Male", "Female", "Mixed (all sexes)")))

## Joining, by = "Simulation"

n2 <- nrow(df\_check)  
if(n1 != n2)print("duplicates created in id-lut join")

df\_check$Time.desc <- as.character(paste0("T=",df\_check$Time))  
df\_check$Time.desc[df\_check$Time.desc == "T=1e-06"] <- "SteadyState"  
df\_check$Dataset.Time <- interaction(df\_check$Dataset,  
 df\_check$Time.desc,lex.order=TRUE)  
df\_check$Dataset.Time <- factor(df\_check$Dataset.Time,  
 levels=levels(df\_check$Dataset.Time))  
calibdata <- df\_check[,names(df\_check) != "Prediction"]  
calibdata <- calibdata[!duplicated(calibdata),]  
print(calibdata)

## Level Simulation Output\_Var Time Data Level...1 Level...2  
## 1 1\_1\_1\_1\_1 1 Cserum 0.000 15.400 1\_1\_1\_1\_1 1\_1\_1\_1\_1  
## 2 1\_1\_1\_1\_1 1 Cserum 5.802 16.600 1\_1\_1\_1\_1 1\_1\_1\_1\_1  
## 3 1\_1\_1\_1\_2 2 Cserum 0.000 4.900 1\_1\_1\_1\_2 1\_1\_1\_1\_2  
## 4 1\_1\_1\_1\_2 2 Cserum 5.802 6.100 1\_1\_1\_1\_2 1\_1\_1\_1\_2  
## 5 1\_1\_1\_1\_3 3 Cserum 0.000 11.300 1\_1\_1\_1\_3 1\_1\_1\_1\_3  
## 6 1\_1\_1\_1\_3 3 Cserum 5.802 7.800 1\_1\_1\_1\_3 1\_1\_1\_1\_3  
## 7 1\_1\_1\_1\_4 4 Cserum 0.000 12.800 1\_1\_1\_1\_4 1\_1\_1\_1\_4  
## 8 1\_1\_1\_1\_4 4 Cserum 5.802 5.600 1\_1\_1\_1\_4 1\_1\_1\_1\_4  
## 9 1\_1\_1\_1\_5 5 Cserum 0.000 20.900 1\_1\_1\_1\_5 1\_1\_1\_1\_5  
## 10 1\_1\_1\_1\_5 5 Cserum 5.802 12.400 1\_1\_1\_1\_5 1\_1\_1\_1\_5  
## 11 1\_1\_1\_1\_6 6 Cserum 0.000 6.300 1\_1\_1\_1\_6 1\_1\_1\_1\_6  
## 12 1\_1\_1\_1\_6 6 Cserum 5.802 6.700 1\_1\_1\_1\_6 1\_1\_1\_1\_6  
## 13 1\_1\_1\_1\_7 7 Cserum 0.000 24.400 1\_1\_1\_1\_7 1\_1\_1\_1\_7  
## 14 1\_1\_1\_1\_7 7 Cserum 5.802 18.100 1\_1\_1\_1\_7 1\_1\_1\_1\_7  
## 15 1\_1\_1\_1\_8 8 Cserum 0.000 24.200 1\_1\_1\_1\_8 1\_1\_1\_1\_8  
## 16 1\_1\_1\_1\_8 8 Cserum 5.802 20.200 1\_1\_1\_1\_8 1\_1\_1\_1\_8  
## 17 1\_1\_1\_1\_9 9 Cserum 0.000 7.800 1\_1\_1\_1\_9 1\_1\_1\_1\_9  
## 18 1\_1\_1\_1\_9 9 Cserum 5.802 8.500 1\_1\_1\_1\_9 1\_1\_1\_1\_9  
## 19 1\_1\_1\_1\_10 10 Cserum 0.000 2.300 1\_1\_1\_1\_10 1\_1\_1\_1\_10  
## 20 1\_1\_1\_1\_10 10 Cserum 5.802 2.500 1\_1\_1\_1\_10 1\_1\_1\_1\_10  
## 21 1\_1\_1\_1\_11 11 Cserum 0.000 12.900 1\_1\_1\_1\_11 1\_1\_1\_1\_11  
## 22 1\_1\_1\_1\_11 11 Cserum 5.802 13.700 1\_1\_1\_1\_11 1\_1\_1\_1\_11  
## 23 1\_1\_1\_1\_12 12 Cserum 0.000 2.400 1\_1\_1\_1\_12 1\_1\_1\_1\_12  
## 24 1\_1\_1\_1\_12 12 Cserum 5.802 3.300 1\_1\_1\_1\_12 1\_1\_1\_1\_12  
## 25 1\_1\_1\_1\_13 13 Cserum 0.000 2.700 1\_1\_1\_1\_13 1\_1\_1\_1\_13  
## 26 1\_1\_1\_1\_13 13 Cserum 5.802 4.000 1\_1\_1\_1\_13 1\_1\_1\_1\_13  
## 27 1\_1\_1\_1\_14 14 Cserum 0.000 18.300 1\_1\_1\_1\_14 1\_1\_1\_1\_14  
## 28 1\_1\_1\_1\_14 14 Cserum 5.802 18.400 1\_1\_1\_1\_14 1\_1\_1\_1\_14  
## 29 1\_1\_1\_1\_15 15 Cserum 0.000 12.900 1\_1\_1\_1\_15 1\_1\_1\_1\_15  
## 30 1\_1\_1\_1\_15 15 Cserum 5.802 13.200 1\_1\_1\_1\_15 1\_1\_1\_1\_15  
## 31 1\_1\_1\_1\_16 16 Cserum 0.000 3.300 1\_1\_1\_1\_16 1\_1\_1\_1\_16  
## 32 1\_1\_1\_1\_16 16 Cserum 5.802 4.100 1\_1\_1\_1\_16 1\_1\_1\_1\_16  
## 33 1\_1\_1\_1\_17 17 Cserum 0.000 16.900 1\_1\_1\_1\_17 1\_1\_1\_1\_17  
## 34 1\_1\_1\_1\_17 17 Cserum 5.802 12.300 1\_1\_1\_1\_17 1\_1\_1\_1\_17  
## 35 1\_1\_1\_1\_18 18 Cserum 0.000 8.700 1\_1\_1\_1\_18 1\_1\_1\_1\_18  
## 36 1\_1\_1\_1\_18 18 Cserum 5.802 6.500 1\_1\_1\_1\_18 1\_1\_1\_1\_18  
## 37 1\_1\_1\_2\_1 19 Cserum 0.000 6.300 1\_1\_1\_2\_1 1\_1\_1\_2\_1  
## 38 1\_1\_1\_2\_1 19 Cserum 5.802 4.600 1\_1\_1\_2\_1 1\_1\_1\_2\_1  
## 39 1\_1\_1\_2\_2 20 Cserum 0.000 7.100 1\_1\_1\_2\_2 1\_1\_1\_2\_2  
## 40 1\_1\_1\_2\_2 20 Cserum 5.802 7.400 1\_1\_1\_2\_2 1\_1\_1\_2\_2  
## 41 1\_1\_1\_2\_3 21 Cserum 0.000 19.500 1\_1\_1\_2\_3 1\_1\_1\_2\_3  
## 42 1\_1\_1\_2\_3 21 Cserum 5.802 14.000 1\_1\_1\_2\_3 1\_1\_1\_2\_3  
## 43 1\_1\_1\_2\_4 22 Cserum 0.000 7.500 1\_1\_1\_2\_4 1\_1\_1\_2\_4  
## 44 1\_1\_1\_2\_4 22 Cserum 5.802 9.000 1\_1\_1\_2\_4 1\_1\_1\_2\_4  
## 45 1\_1\_1\_2\_5 23 Cserum 0.000 8.600 1\_1\_1\_2\_5 1\_1\_1\_2\_5  
## 46 1\_1\_1\_2\_5 23 Cserum 5.802 9.200 1\_1\_1\_2\_5 1\_1\_1\_2\_5  
## 47 1\_1\_1\_2\_6 24 Cserum 0.000 12.400 1\_1\_1\_2\_6 1\_1\_1\_2\_6  
## 48 1\_1\_1\_2\_6 24 Cserum 5.802 12.800 1\_1\_1\_2\_6 1\_1\_1\_2\_6  
## 49 1\_1\_1\_2\_7 25 Cserum 0.000 8.900 1\_1\_1\_2\_7 1\_1\_1\_2\_7  
## 50 1\_1\_1\_2\_7 25 Cserum 5.802 6.600 1\_1\_1\_2\_7 1\_1\_1\_2\_7  
## 51 1\_1\_1\_2\_8 26 Cserum 0.000 7.400 1\_1\_1\_2\_8 1\_1\_1\_2\_8  
## 52 1\_1\_1\_2\_8 26 Cserum 5.802 6.300 1\_1\_1\_2\_8 1\_1\_1\_2\_8  
## 53 1\_1\_1\_2\_9 27 Cserum 0.000 14.300 1\_1\_1\_2\_9 1\_1\_1\_2\_9  
## 54 1\_1\_1\_2\_9 27 Cserum 5.802 15.800 1\_1\_1\_2\_9 1\_1\_1\_2\_9  
## 55 1\_1\_1\_2\_10 28 Cserum 0.000 3.800 1\_1\_1\_2\_10 1\_1\_1\_2\_10  
## 56 1\_1\_1\_2\_10 28 Cserum 5.802 4.200 1\_1\_1\_2\_10 1\_1\_1\_2\_10  
## 57 1\_1\_1\_2\_11 29 Cserum 0.000 22.100 1\_1\_1\_2\_11 1\_1\_1\_2\_11  
## 58 1\_1\_1\_2\_11 29 Cserum 5.802 26.000 1\_1\_1\_2\_11 1\_1\_1\_2\_11  
## 59 1\_1\_1\_2\_12 30 Cserum 0.000 6.500 1\_1\_1\_2\_12 1\_1\_1\_2\_12  
## 60 1\_1\_1\_2\_12 30 Cserum 5.802 5.800 1\_1\_1\_2\_12 1\_1\_1\_2\_12  
## 61 1\_1\_1\_2\_13 31 Cserum 0.000 7.500 1\_1\_1\_2\_13 1\_1\_1\_2\_13  
## 62 1\_1\_1\_2\_13 31 Cserum 5.802 7.900 1\_1\_1\_2\_13 1\_1\_1\_2\_13  
## 63 1\_1\_1\_2\_14 32 Cserum 0.000 7.500 1\_1\_1\_2\_14 1\_1\_1\_2\_14  
## 64 1\_1\_1\_2\_14 32 Cserum 5.802 6.800 1\_1\_1\_2\_14 1\_1\_1\_2\_14  
## 65 1\_1\_1\_2\_15 33 Cserum 0.000 7.500 1\_1\_1\_2\_15 1\_1\_1\_2\_15  
## 66 1\_1\_1\_2\_15 33 Cserum 5.802 2.400 1\_1\_1\_2\_15 1\_1\_1\_2\_15  
## 67 1\_1\_1\_2\_16 34 Cserum 0.000 6.000 1\_1\_1\_2\_16 1\_1\_1\_2\_16  
## 68 1\_1\_1\_2\_16 34 Cserum 5.802 6.800 1\_1\_1\_2\_16 1\_1\_1\_2\_16  
## 69 1\_1\_1\_2\_17 35 Cserum 0.000 18.200 1\_1\_1\_2\_17 1\_1\_1\_2\_17  
## 70 1\_1\_1\_2\_17 35 Cserum 5.802 13.200 1\_1\_1\_2\_17 1\_1\_1\_2\_17  
## 71 1\_1\_1\_2\_18 36 Cserum 0.000 5.300 1\_1\_1\_2\_18 1\_1\_1\_2\_18  
## 72 1\_1\_1\_2\_18 36 Cserum 5.802 3.900 1\_1\_1\_2\_18 1\_1\_1\_2\_18  
## 73 1\_1\_1\_2\_19 37 Cserum 0.000 5.600 1\_1\_1\_2\_19 1\_1\_1\_2\_19  
## 74 1\_1\_1\_2\_19 37 Cserum 5.802 5.400 1\_1\_1\_2\_19 1\_1\_1\_2\_19  
## 75 1\_2\_1 38 M\_Cbgd\_Css 2.200 2.593 1\_2\_1 1\_2\_1  
## 76 1\_3\_1 39 M\_Cbgd\_Css 2.000 27.340 1\_3\_1 1\_3\_1  
## 77 1\_4\_1 40 M\_Cbgd\_Css 2.000 23.157 1\_4\_1 1\_4\_1  
## 78 1\_4\_2\_1 41 M\_Cbgd\_Css 2.000 14.851 1\_4\_2\_1 1\_4\_2\_1  
## dataset Sex City Train\_Test datatype  
## 1 Decatur M Train Male Decatur Train Individual  
## 2 Decatur M Train Male Decatur Train Individual  
## 3 Decatur M Train Male Decatur Train Individual  
## 4 Decatur M Train Male Decatur Train Individual  
## 5 Decatur M Train Male Decatur Train Individual  
## 6 Decatur M Train Male Decatur Train Individual  
## 7 Decatur M Train Male Decatur Train Individual  
## 8 Decatur M Train Male Decatur Train Individual  
## 9 Decatur M Train Male Decatur Train Individual  
## 10 Decatur M Train Male Decatur Train Individual  
## 11 Decatur M Train Male Decatur Train Individual  
## 12 Decatur M Train Male Decatur Train Individual  
## 13 Decatur M Train Male Decatur Train Individual  
## 14 Decatur M Train Male Decatur Train Individual  
## 15 Decatur M Train Male Decatur Train Individual  
## 16 Decatur M Train Male Decatur Train Individual  
## 17 Decatur M Train Male Decatur Train Individual  
## 18 Decatur M Train Male Decatur Train Individual  
## 19 Decatur F Train Female Decatur Train Individual  
## 20 Decatur F Train Female Decatur Train Individual  
## 21 Decatur F Train Female Decatur Train Individual  
## 22 Decatur F Train Female Decatur Train Individual  
## 23 Decatur F Train Female Decatur Train Individual  
## 24 Decatur F Train Female Decatur Train Individual  
## 25 Decatur F Train Female Decatur Train Individual  
## 26 Decatur F Train Female Decatur Train Individual  
## 27 Decatur F Train Female Decatur Train Individual  
## 28 Decatur F Train Female Decatur Train Individual  
## 29 Decatur F Train Female Decatur Train Individual  
## 30 Decatur F Train Female Decatur Train Individual  
## 31 Decatur F Train Female Decatur Train Individual  
## 32 Decatur F Train Female Decatur Train Individual  
## 33 Decatur F Train Female Decatur Train Individual  
## 34 Decatur F Train Female Decatur Train Individual  
## 35 Decatur F Train Female Decatur Train Individual  
## 36 Decatur F Train Female Decatur Train Individual  
## 37 Decatur M Test Male Decatur Test Individual  
## 38 Decatur M Test Male Decatur Test Individual  
## 39 Decatur M Test Male Decatur Test Individual  
## 40 Decatur M Test Male Decatur Test Individual  
## 41 Decatur M Test Male Decatur Test Individual  
## 42 Decatur M Test Male Decatur Test Individual  
## 43 Decatur M Test Male Decatur Test Individual  
## 44 Decatur M Test Male Decatur Test Individual  
## 45 Decatur M Test Male Decatur Test Individual  
## 46 Decatur M Test Male Decatur Test Individual  
## 47 Decatur M Test Male Decatur Test Individual  
## 48 Decatur M Test Male Decatur Test Individual  
## 49 Decatur M Test Male Decatur Test Individual  
## 50 Decatur M Test Male Decatur Test Individual  
## 51 Decatur M Test Male Decatur Test Individual  
## 52 Decatur M Test Male Decatur Test Individual  
## 53 Decatur M Test Male Decatur Test Individual  
## 54 Decatur M Test Male Decatur Test Individual  
## 55 Decatur F Test Female Decatur Test Individual  
## 56 Decatur F Test Female Decatur Test Individual  
## 57 Decatur F Test Female Decatur Test Individual  
## 58 Decatur F Test Female Decatur Test Individual  
## 59 Decatur F Test Female Decatur Test Individual  
## 60 Decatur F Test Female Decatur Test Individual  
## 61 Decatur F Test Female Decatur Test Individual  
## 62 Decatur F Test Female Decatur Test Individual  
## 63 Decatur F Test Female Decatur Test Individual  
## 64 Decatur F Test Female Decatur Test Individual  
## 65 Decatur F Test Female Decatur Test Individual  
## 66 Decatur F Test Female Decatur Test Individual  
## 67 Decatur F Test Female Decatur Test Individual  
## 68 Decatur F Test Female Decatur Test Individual  
## 69 Decatur F Test Female Decatur Test Individual  
## 70 Decatur F Test Female Decatur Test Individual  
## 71 Decatur F Test Female Decatur Test Individual  
## 72 Decatur F Test Female Decatur Test Individual  
## 73 Decatur F Test Female Decatur Test Individual  
## 74 Decatur F Test Female Decatur Test Individual  
## 75 Paulsboro-Train Mixed (all sexes) Paulsboro Train Summary  
## 76 Horsham-Train Mixed (all sexes) Horsham Train Summary  
## 77 Warminster-Test Mixed (all sexes) Warminster Test Summary  
## 78 Warrington-Train Mixed (all sexes) Warrington Test Summary  
## variable Dataset Time.desc Dataset.Time  
## 1 Decatur M Train 1 Decatur M Train 1 T=0 Decatur M Train 1.T=0  
## 2 Decatur M Train 1 Decatur M Train 1 T=5.802 Decatur M Train 1.T=5.802  
## 3 Decatur M Train 2 Decatur M Train 2 T=0 Decatur M Train 2.T=0  
## 4 Decatur M Train 2 Decatur M Train 2 T=5.802 Decatur M Train 2.T=5.802  
## 5 Decatur M Train 3 Decatur M Train 3 T=0 Decatur M Train 3.T=0  
## 6 Decatur M Train 3 Decatur M Train 3 T=5.802 Decatur M Train 3.T=5.802  
## 7 Decatur M Train 4 Decatur M Train 4 T=0 Decatur M Train 4.T=0  
## 8 Decatur M Train 4 Decatur M Train 4 T=5.802 Decatur M Train 4.T=5.802  
## 9 Decatur M Train 5 Decatur M Train 5 T=0 Decatur M Train 5.T=0  
## 10 Decatur M Train 5 Decatur M Train 5 T=5.802 Decatur M Train 5.T=5.802  
## 11 Decatur M Train 6 Decatur M Train 6 T=0 Decatur M Train 6.T=0  
## 12 Decatur M Train 6 Decatur M Train 6 T=5.802 Decatur M Train 6.T=5.802  
## 13 Decatur M Train 7 Decatur M Train 7 T=0 Decatur M Train 7.T=0  
## 14 Decatur M Train 7 Decatur M Train 7 T=5.802 Decatur M Train 7.T=5.802  
## 15 Decatur M Train 8 Decatur M Train 8 T=0 Decatur M Train 8.T=0  
## 16 Decatur M Train 8 Decatur M Train 8 T=5.802 Decatur M Train 8.T=5.802  
## 17 Decatur M Train 9 Decatur M Train 9 T=0 Decatur M Train 9.T=0  
## 18 Decatur M Train 9 Decatur M Train 9 T=5.802 Decatur M Train 9.T=5.802  
## 19 Decatur F Train 10 Decatur F Train 10 T=0 Decatur F Train 10.T=0  
## 20 Decatur F Train 10 Decatur F Train 10 T=5.802 Decatur F Train 10.T=5.802  
## 21 Decatur F Train 11 Decatur F Train 11 T=0 Decatur F Train 11.T=0  
## 22 Decatur F Train 11 Decatur F Train 11 T=5.802 Decatur F Train 11.T=5.802  
## 23 Decatur F Train 12 Decatur F Train 12 T=0 Decatur F Train 12.T=0  
## 24 Decatur F Train 12 Decatur F Train 12 T=5.802 Decatur F Train 12.T=5.802  
## 25 Decatur F Train 13 Decatur F Train 13 T=0 Decatur F Train 13.T=0  
## 26 Decatur F Train 13 Decatur F Train 13 T=5.802 Decatur F Train 13.T=5.802  
## 27 Decatur F Train 14 Decatur F Train 14 T=0 Decatur F Train 14.T=0  
## 28 Decatur F Train 14 Decatur F Train 14 T=5.802 Decatur F Train 14.T=5.802  
## 29 Decatur F Train 15 Decatur F Train 15 T=0 Decatur F Train 15.T=0  
## 30 Decatur F Train 15 Decatur F Train 15 T=5.802 Decatur F Train 15.T=5.802  
## 31 Decatur F Train 16 Decatur F Train 16 T=0 Decatur F Train 16.T=0  
## 32 Decatur F Train 16 Decatur F Train 16 T=5.802 Decatur F Train 16.T=5.802  
## 33 Decatur F Train 17 Decatur F Train 17 T=0 Decatur F Train 17.T=0  
## 34 Decatur F Train 17 Decatur F Train 17 T=5.802 Decatur F Train 17.T=5.802  
## 35 Decatur F Train 18 Decatur F Train 18 T=0 Decatur F Train 18.T=0  
## 36 Decatur F Train 18 Decatur F Train 18 T=5.802 Decatur F Train 18.T=5.802  
## 37 Decatur M Test 19 Decatur M Test 19 T=0 Decatur M Test 19.T=0  
## 38 Decatur M Test 19 Decatur M Test 19 T=5.802 Decatur M Test 19.T=5.802  
## 39 Decatur M Test 20 Decatur M Test 20 T=0 Decatur M Test 20.T=0  
## 40 Decatur M Test 20 Decatur M Test 20 T=5.802 Decatur M Test 20.T=5.802  
## 41 Decatur M Test 21 Decatur M Test 21 T=0 Decatur M Test 21.T=0  
## 42 Decatur M Test 21 Decatur M Test 21 T=5.802 Decatur M Test 21.T=5.802  
## 43 Decatur M Test 22 Decatur M Test 22 T=0 Decatur M Test 22.T=0  
## 44 Decatur M Test 22 Decatur M Test 22 T=5.802 Decatur M Test 22.T=5.802  
## 45 Decatur M Test 23 Decatur M Test 23 T=0 Decatur M Test 23.T=0  
## 46 Decatur M Test 23 Decatur M Test 23 T=5.802 Decatur M Test 23.T=5.802  
## 47 Decatur M Test 24 Decatur M Test 24 T=0 Decatur M Test 24.T=0  
## 48 Decatur M Test 24 Decatur M Test 24 T=5.802 Decatur M Test 24.T=5.802  
## 49 Decatur M Test 25 Decatur M Test 25 T=0 Decatur M Test 25.T=0  
## 50 Decatur M Test 25 Decatur M Test 25 T=5.802 Decatur M Test 25.T=5.802  
## 51 Decatur M Test 26 Decatur M Test 26 T=0 Decatur M Test 26.T=0  
## 52 Decatur M Test 26 Decatur M Test 26 T=5.802 Decatur M Test 26.T=5.802  
## 53 Decatur M Test 27 Decatur M Test 27 T=0 Decatur M Test 27.T=0  
## 54 Decatur M Test 27 Decatur M Test 27 T=5.802 Decatur M Test 27.T=5.802  
## 55 Decatur F Test 28 Decatur F Test 28 T=0 Decatur F Test 28.T=0  
## 56 Decatur F Test 28 Decatur F Test 28 T=5.802 Decatur F Test 28.T=5.802  
## 57 Decatur F Test 29 Decatur F Test 29 T=0 Decatur F Test 29.T=0  
## 58 Decatur F Test 29 Decatur F Test 29 T=5.802 Decatur F Test 29.T=5.802  
## 59 Decatur F Test 30 Decatur F Test 30 T=0 Decatur F Test 30.T=0  
## 60 Decatur F Test 30 Decatur F Test 30 T=5.802 Decatur F Test 30.T=5.802  
## 61 Decatur F Test 31 Decatur F Test 31 T=0 Decatur F Test 31.T=0  
## 62 Decatur F Test 31 Decatur F Test 31 T=5.802 Decatur F Test 31.T=5.802  
## 63 Decatur F Test 32 Decatur F Test 32 T=0 Decatur F Test 32.T=0  
## 64 Decatur F Test 32 Decatur F Test 32 T=5.802 Decatur F Test 32.T=5.802  
## 65 Decatur F Test 33 Decatur F Test 33 T=0 Decatur F Test 33.T=0  
## 66 Decatur F Test 33 Decatur F Test 33 T=5.802 Decatur F Test 33.T=5.802  
## 67 Decatur F Test 34 Decatur F Test 34 T=0 Decatur F Test 34.T=0  
## 68 Decatur F Test 34 Decatur F Test 34 T=5.802 Decatur F Test 34.T=5.802  
## 69 Decatur F Test 35 Decatur F Test 35 T=0 Decatur F Test 35.T=0  
## 70 Decatur F Test 35 Decatur F Test 35 T=5.802 Decatur F Test 35.T=5.802  
## 71 Decatur F Test 36 Decatur F Test 36 T=0 Decatur F Test 36.T=0  
## 72 Decatur F Test 36 Decatur F Test 36 T=5.802 Decatur F Test 36.T=5.802  
## 73 Decatur F Test 37 Decatur F Test 37 T=0 Decatur F Test 37.T=0  
## 74 Decatur F Test 37 Decatur F Test 37 T=5.802 Decatur F Test 37.T=5.802  
## 75 Paulsboro-Train 38 Paulsboro-Train 38 T=2.2 Paulsboro-Train 38.T=2.2  
## 76 Horsham-Train 39 Horsham-Train 39 T=2 Horsham-Train 39.T=2  
## 77 Warminster-Test 40 Warminster-Test 40 T=2 Warminster-Test 40.T=2  
## 78 Warrington-Train 41 Warrington-Train 41 T=2 Warrington-Train 41.T=2

#Multicheck plot  
  
# Split Steady State Group into different populations for boxplot grouping  
#df\_check[df\_check$Time.desc == "SteadyState" & grepl("Lubeck",df\_check$Dataset),]$Time.desc <- "Lubeck"  
#df\_check[df\_check$Time.desc == "SteadyState" & grepl("Little Hocking",df\_check$Dataset),]$Time.desc <- "Little Hocking"

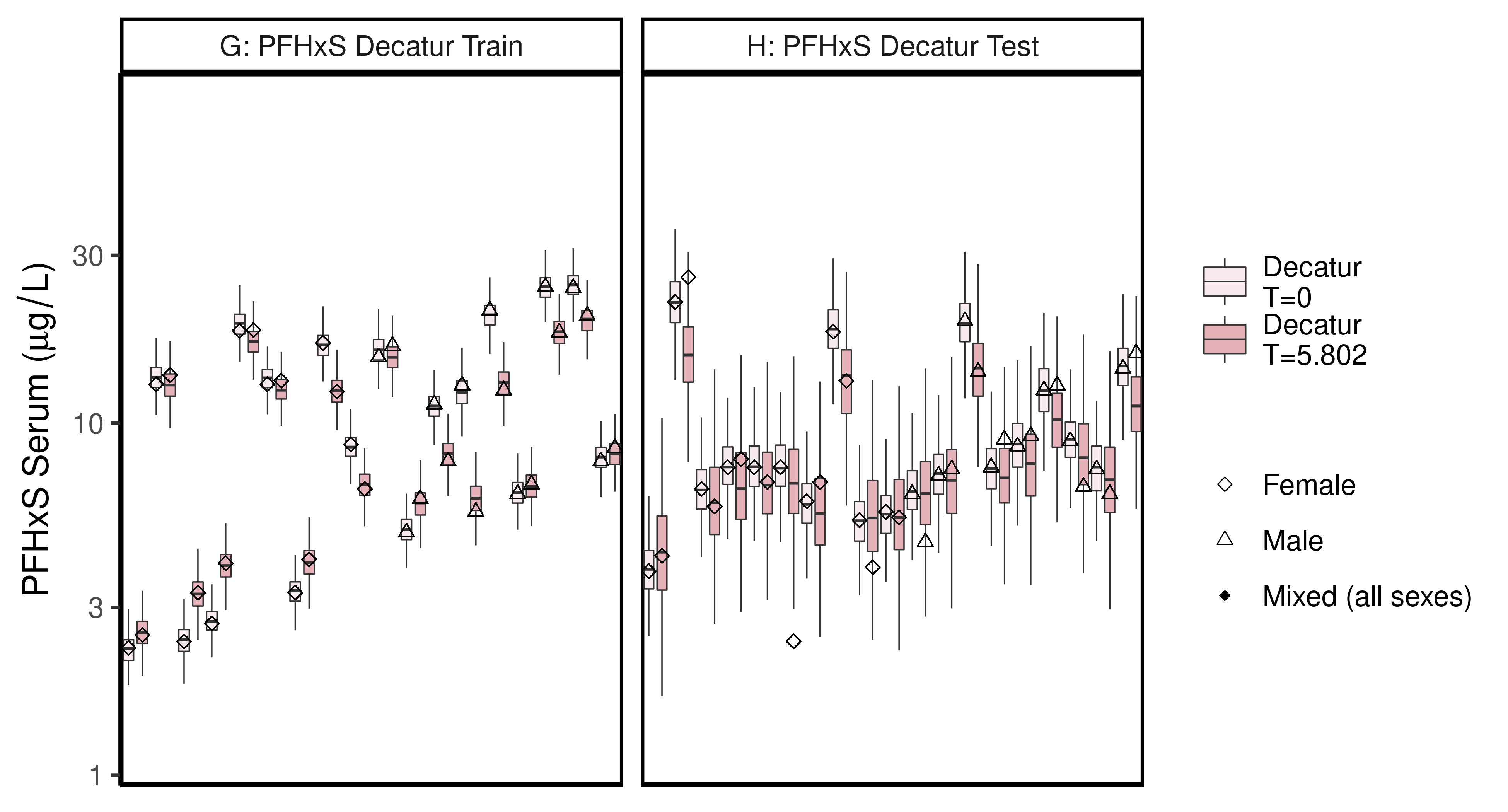
Modify aesthetics lookup table for boxplots

## additional source aesthetic lookup table for grey-scale time (years); merged legends save space on plotting output  
times <- df\_check%>% select(Time.desc, Time) %>% unique () %>%   
 mutate(rank = rank(Time) , grey = grey.colors(start=1,end=0.4, n = n()),  
 alpha = (rank)/8) %>%   
 select(-Time)  
   
df\_check <- df\_check %>% mutate (legend\_label = (paste0(City, "\n", Time.desc ) )) # add legend-labels  
aes\_lut <- df\_check %>%   
 select(City, Train\_Test, datatype,Time, Time.desc, legend\_label) %>% unique () %>%  
 left\_join(aes\_lut[, c("City", "cols")], by = "City") %>% ungroup () %>% unique ()%>%  
 left\_join (times, by = "Time.desc") %>%   
 arrange(datatype, City, Train\_Test, Time) %>%   
 mutate(alpha = if\_else(City == "Horsham", alpha/2, alpha)) %>% # otherwise too dark with this color  
 mutate\_if(is.factor, as.character)

## Decatur boxplots

Changed grey start to 1 instead of 0.8, end at 0.6 instead of 0.4. Changed shape of symbols so they are filled.

#GH  
# Decatur   
  
df\_decat <- df\_check %>%   
 filter(City == "Decatur" & Train\_Test %in% c ("Train", "Test")) %>%   
 mutate(panel = ordered (Train\_Test, levels = c ("Train", "Test"),   
 labels = c("G: PFHxS Decatur Train", "H: PFHxS Decatur Test") ))  
  
aes\_lut\_df\_df\_decat <- aes\_lut %>%   
 filter(City == "Decatur" & Train\_Test %in% c ("Train", "Test")) %>%   
 mutate\_if(is.factor, as.character)   
  
source( paste0(gsub(basename(here()), 'shared\_functions', here()), '/plot\_sum\_boxplot.r'))  
  
  
plt\_train <- plot\_sum\_boxplot (dframe = df\_decat, aes\_lut= aes\_lut\_df\_df\_decat, facets = TRUE , pfas\_nom = pfas\_name )   
print(plt\_train)



ggsave(here ("output-plots",paste0( sa,"DecaturTrainTestboxplot",pfas\_name,".pdf")),plt\_train,dpi=600)

## Saving 6.5 x 3.5 in image

ggsave(here ("output-plots",paste0( sa,"DecaturTrainTestboxplot",pfas\_name,".png")),plt\_train,dpi=600)

## Saving 6.5 x 3.5 in image

## All boxplots

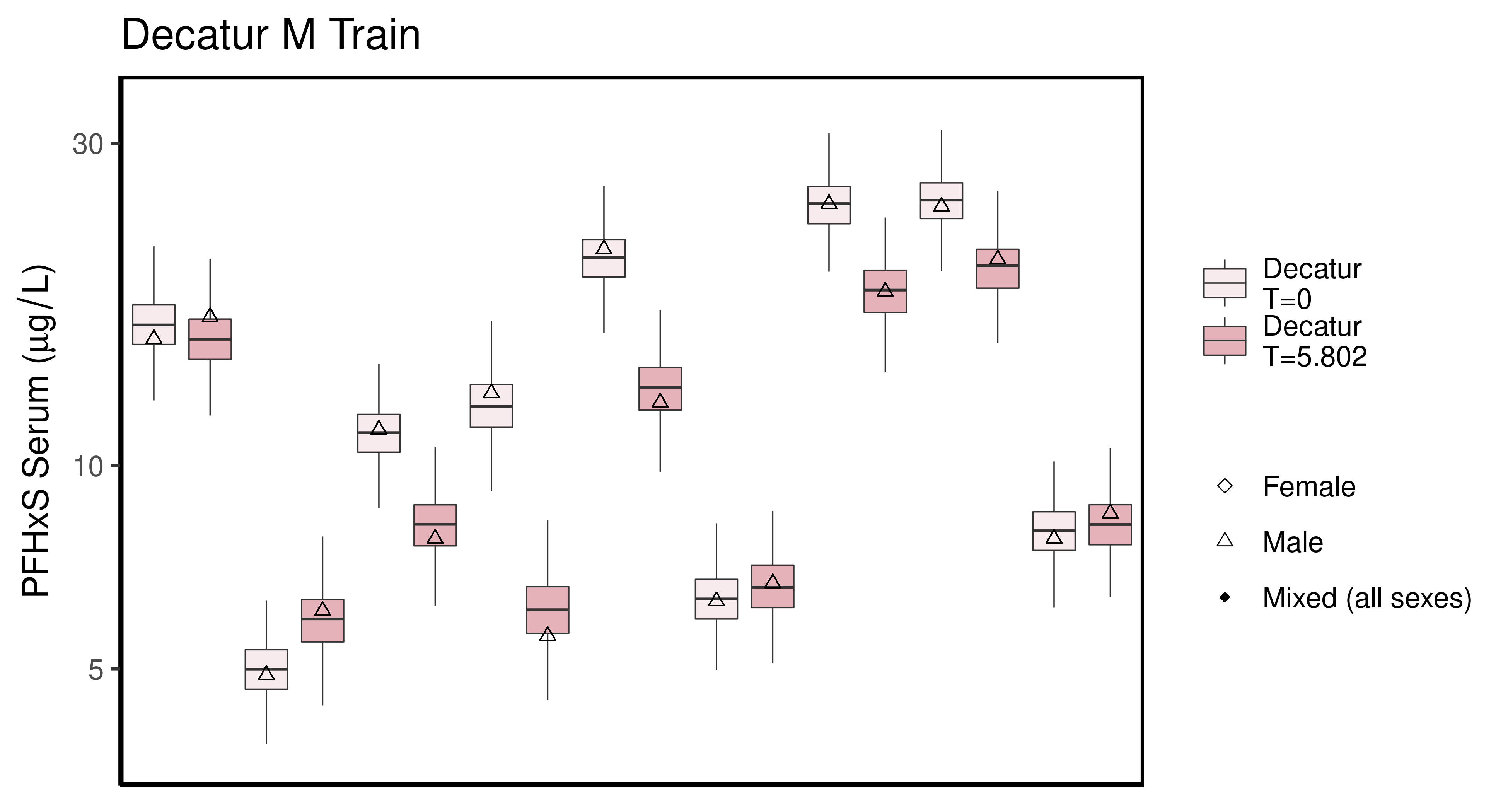
Changed grey start to 1 instead of 0.8, end at 0.6 instead of 0.4. Added shapes and fills to data points.

lets <- LETTERS;  
names(lets)[1:(length(unique(df\_check$dataset))-4)]<-as.character(unique(df\_check$dataset))[5:length(unique(df\_check$dataset))]  
  
for (d in unique(df\_check$dataset)) { # d = unique(df\_check$dataset)[11]  
 ddset <- df\_check %>%   
 filter(dataset == d)   
   
 aes\_lut\_ddset <- ddset %>% select(legend\_label, City,Train\_Test,datatype, Time.desc ) %>% unique () %>% inner\_join(aes\_lut)  
   
 gt <- ifelse(is.na(lets[d]),d,paste0(lets[d],": ", d))  
 plt <- plot\_sum\_boxplot(dframe = ddset, aes\_lut= aes\_lut\_ddset, gtitle= gt, facets = FALSE, pfas\_nom = pfas\_name)  
   
 print(plt)  
 ggsave(here ("output-plots",  
 paste0( sa, d,"-boxplot-",   
 pfas\_name,".pdf")) ,  
 plt,dpi=600)  
   
 ggsave(here ("output-plots",  
 paste0( sa, d,"-boxplot-",   
 pfas\_name,".png")) ,  
 plt,dpi=600)  
  
}

## Joining, by = c("legend\_label", "City", "Train\_Test", "datatype", "Time.desc")

## Saving 6.5 x 3.5 in image  
## Saving 6.5 x 3.5 in image

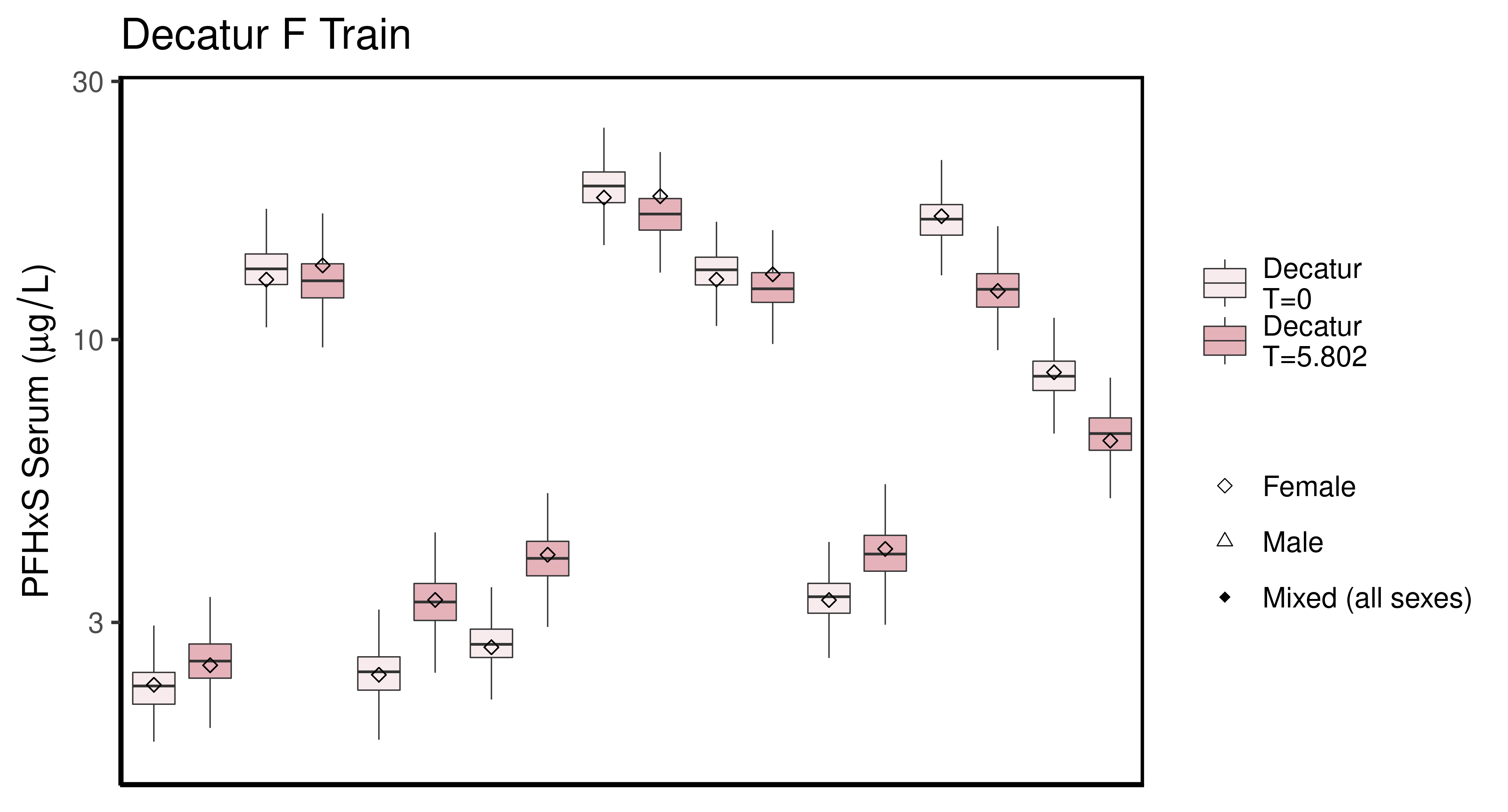
## Joining, by = c("legend\_label", "City", "Train\_Test", "datatype", "Time.desc")



## Saving 6.5 x 3.5 in image

## Saving 6.5 x 3.5 in image

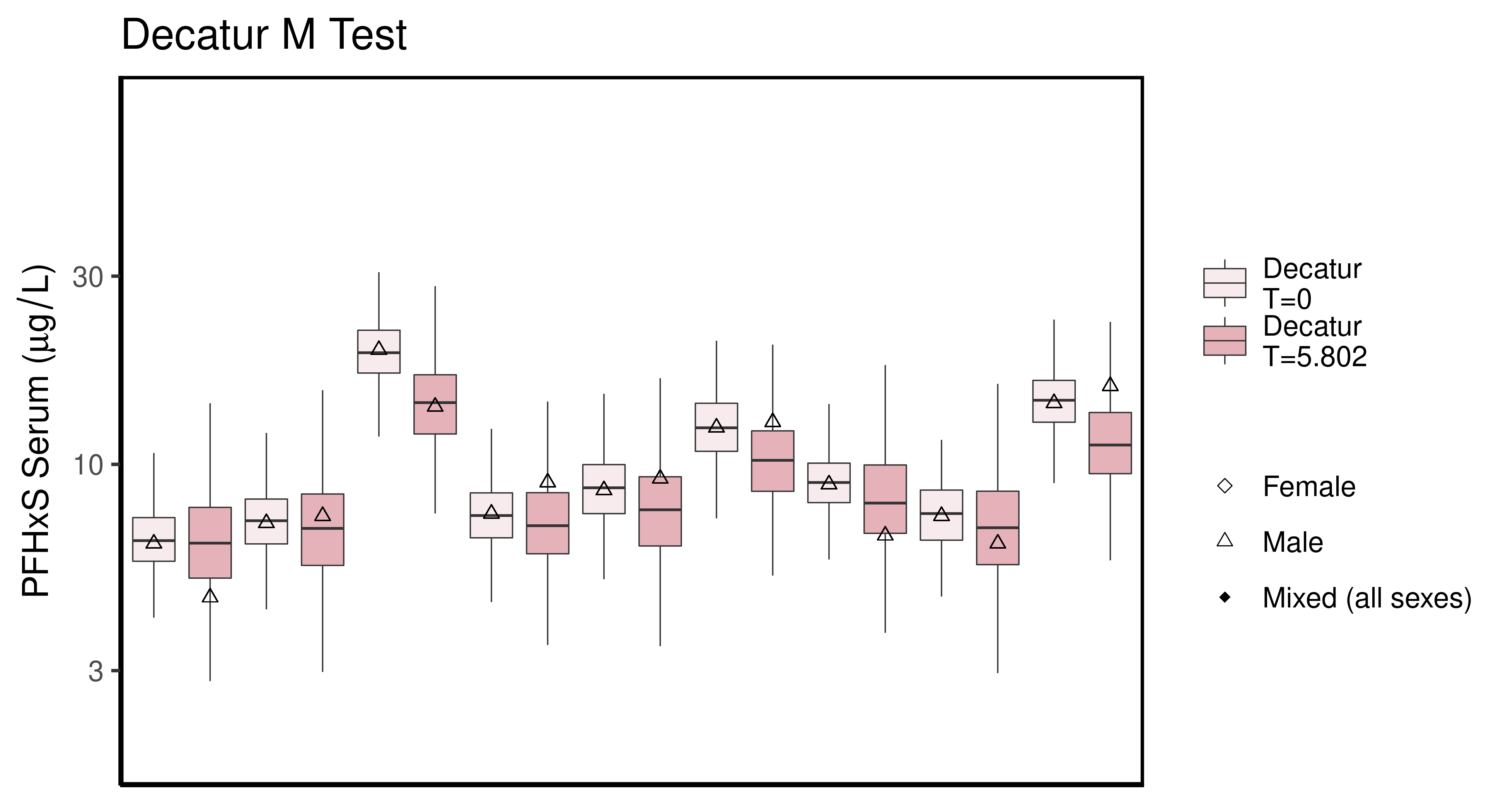
## Joining, by = c("legend\_label", "City", "Train\_Test", "datatype", "Time.desc")



## Saving 6.5 x 3.5 in image

## Saving 6.5 x 3.5 in image

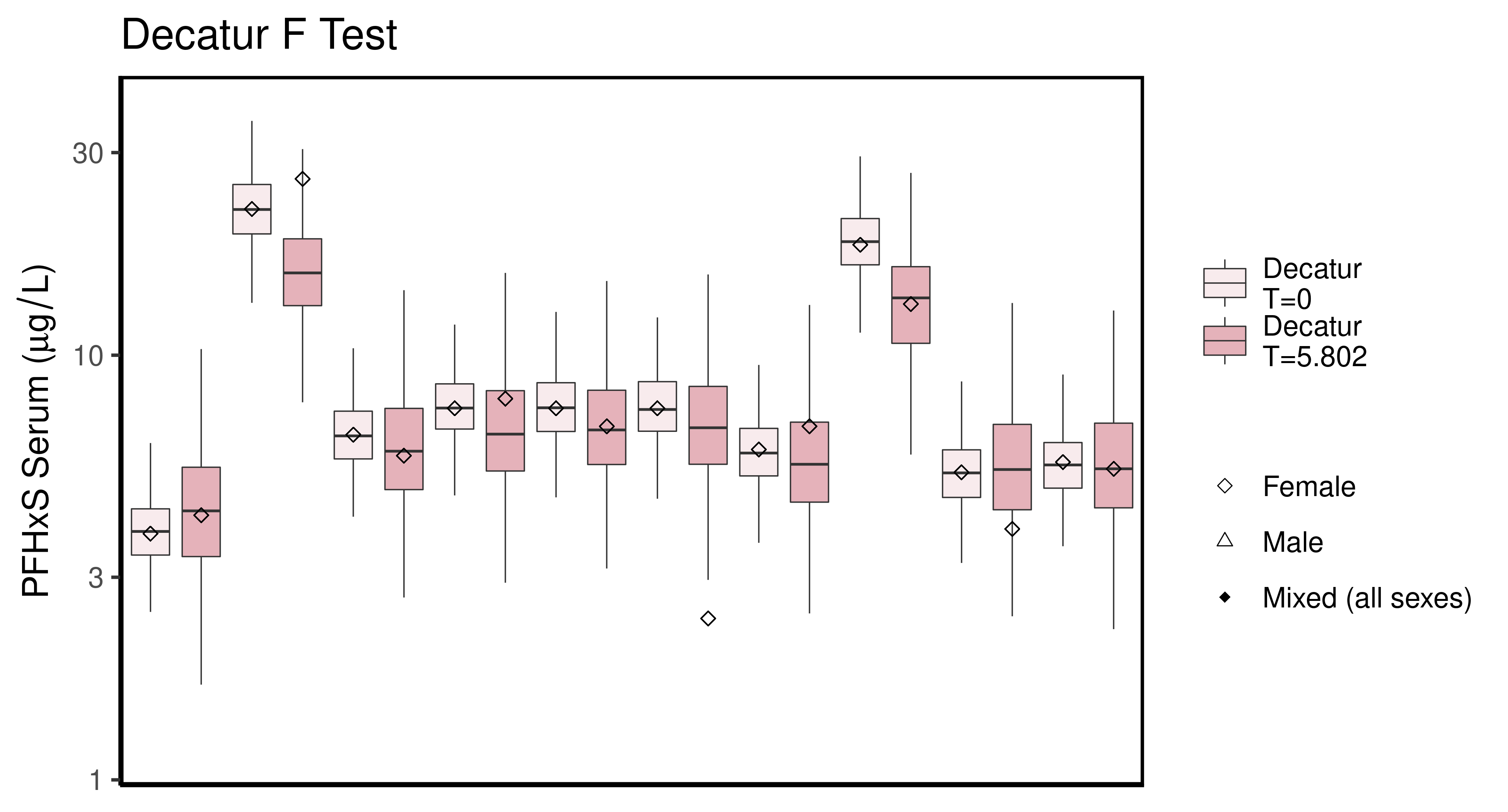
## Joining, by = c("legend\_label", "City", "Train\_Test", "datatype", "Time.desc")



## Saving 6.5 x 3.5 in image

## Saving 6.5 x 3.5 in image

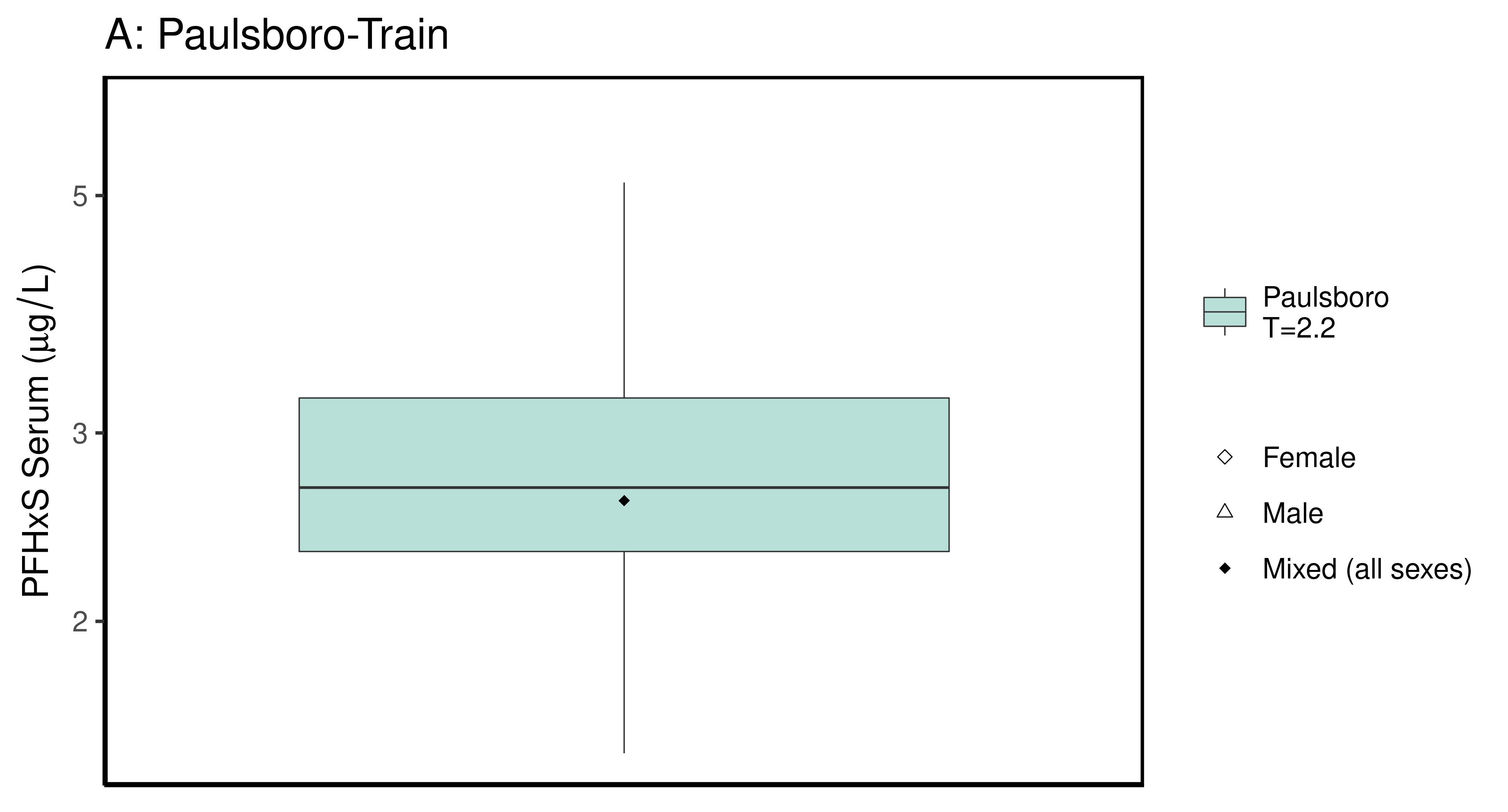
## Joining, by = c("legend\_label", "City", "Train\_Test", "datatype", "Time.desc")



## Saving 6.5 x 3.5 in image

## Saving 6.5 x 3.5 in image

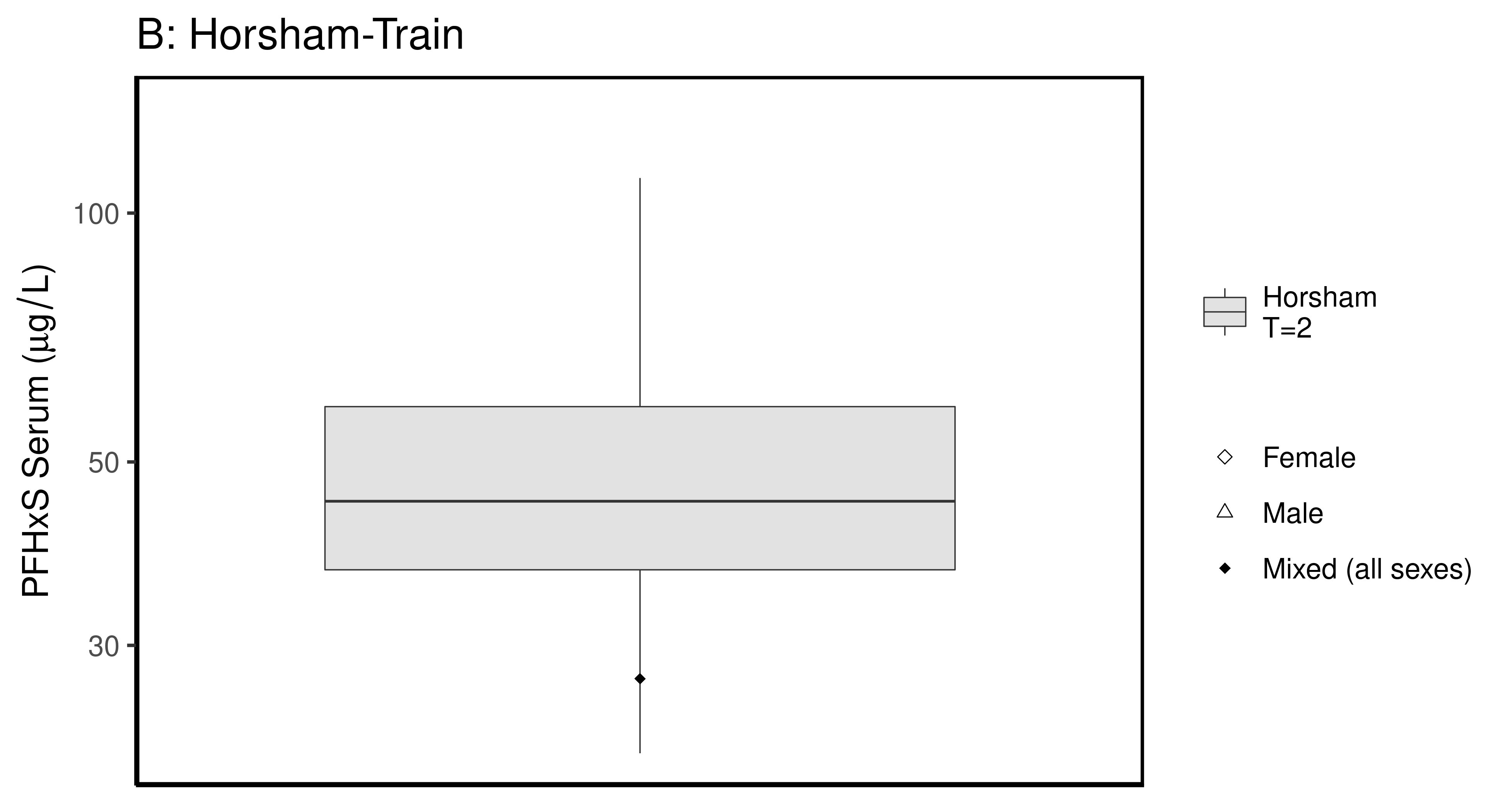
## Joining, by = c("legend\_label", "City", "Train\_Test", "datatype", "Time.desc")



## Saving 6.5 x 3.5 in image

## Saving 6.5 x 3.5 in image

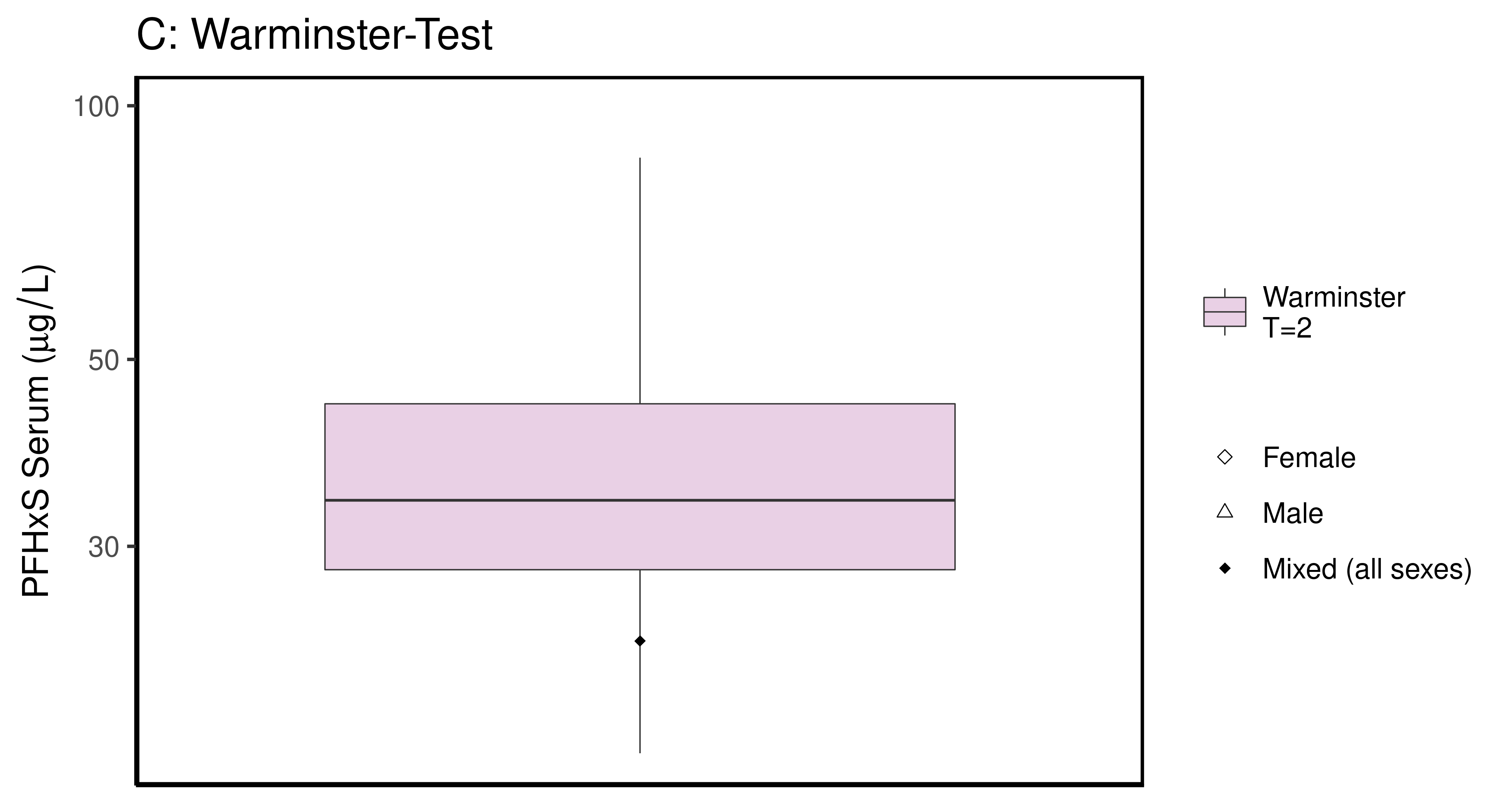
## Joining, by = c("legend\_label", "City", "Train\_Test", "datatype", "Time.desc")



## Saving 6.5 x 3.5 in image

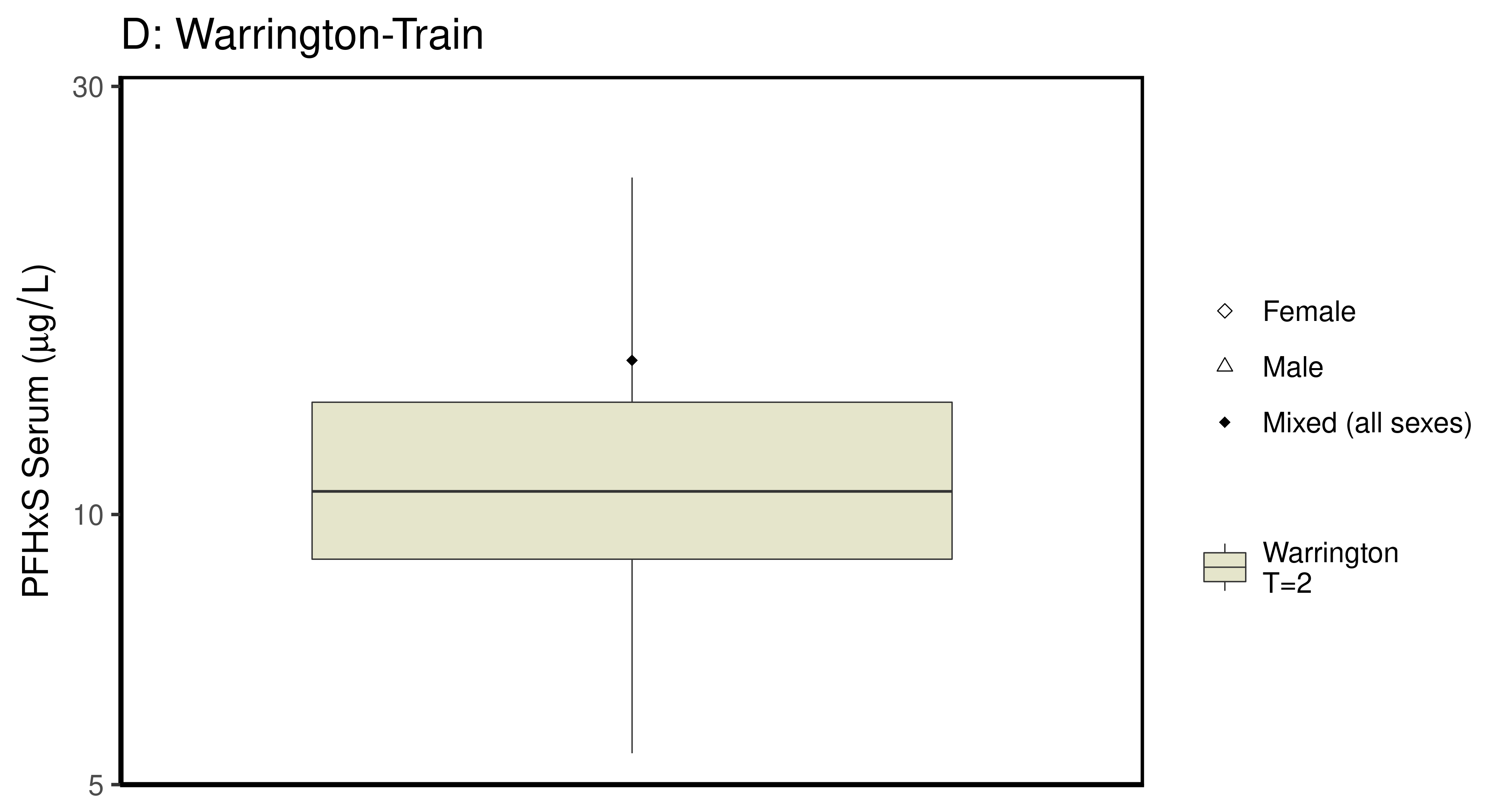
## Saving 6.5 x 3.5 in image

## Joining, by = c("legend\_label", "City", "Train\_Test", "datatype", "Time.desc")



## Saving 6.5 x 3.5 in image

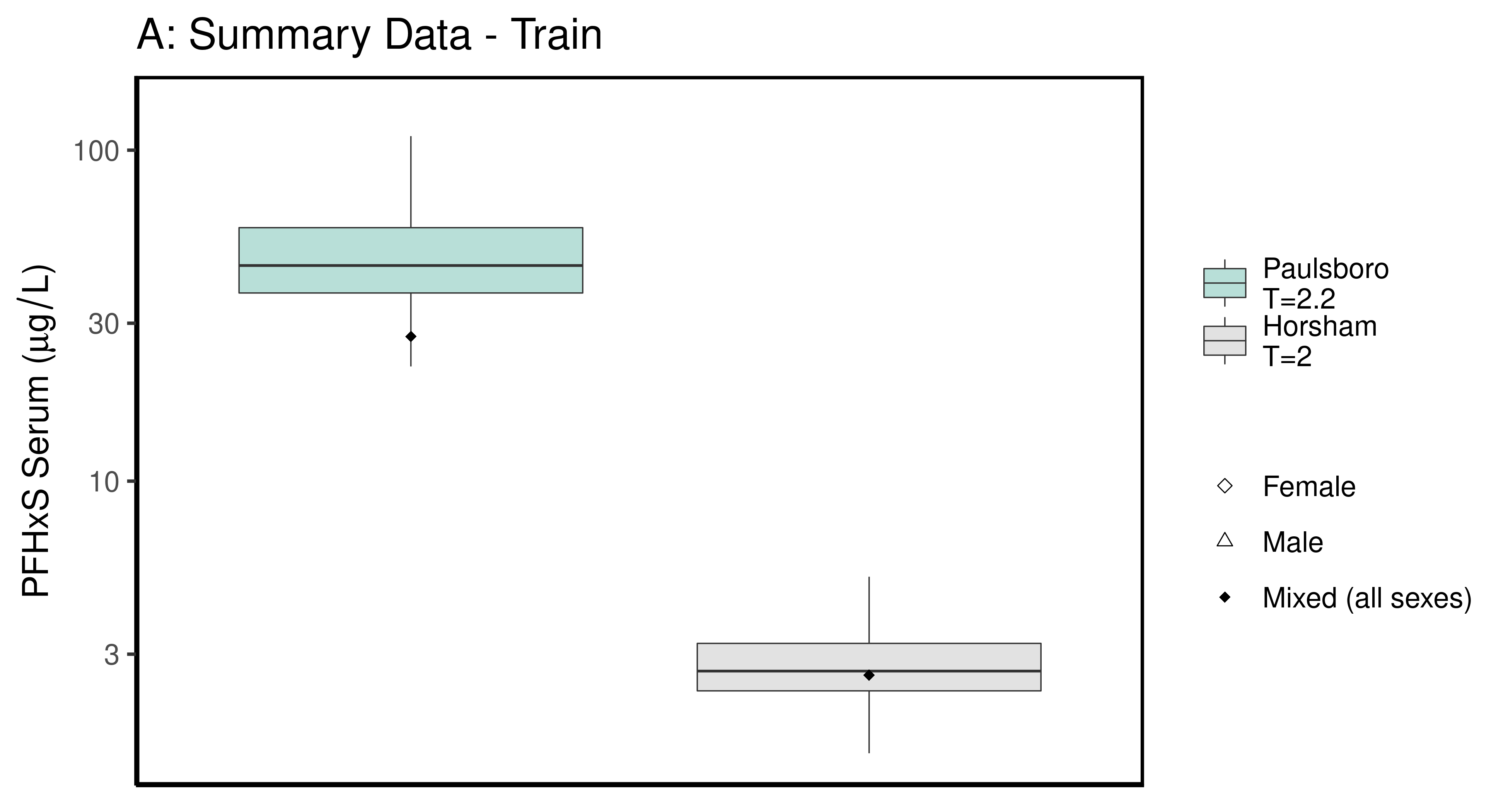
## Saving 6.5 x 3.5 in image



### make Training plot   
  
df\_d\_trt <- df\_check %>%   
 filter( (Train\_Test == "Train") & ((Output\_Var == "M\_Cbgd\_Css") | (Output\_Var == "M\_Cserum"))) %>%  
 mutate\_if(is.factor, as.character) %>% # drop factor levels unused  
 mutate(Dataset.Time = factor(Dataset.Time))   
   
  
 aes\_lut\_df\_d\_trt <- df\_d\_trt %>% select(City, datatype,Time, Time.desc, legend\_label) %>%   
 inner\_join(aes\_lut ) %>%   
 select(-Train\_Test) %>% ungroup () %>% unique ()

## Joining, by = c("City", "datatype", "Time", "Time.desc", "legend\_label")

plt\_train <- plot\_sum\_boxplot(dframe = df\_d\_trt, aes\_lut= aes\_lut\_df\_d\_trt,   
 gtitle="A: Summary Data - Train" , facets = FALSE,   
 pfas\_nom = pfas\_name )  
 print(plt\_train)



ggsave(here ("output-plots", paste0( sa, "SummaryTrainDataboxplot", pfas\_name,".pdf")), plt\_train,dpi=600)

## Saving 6.5 x 3.5 in image

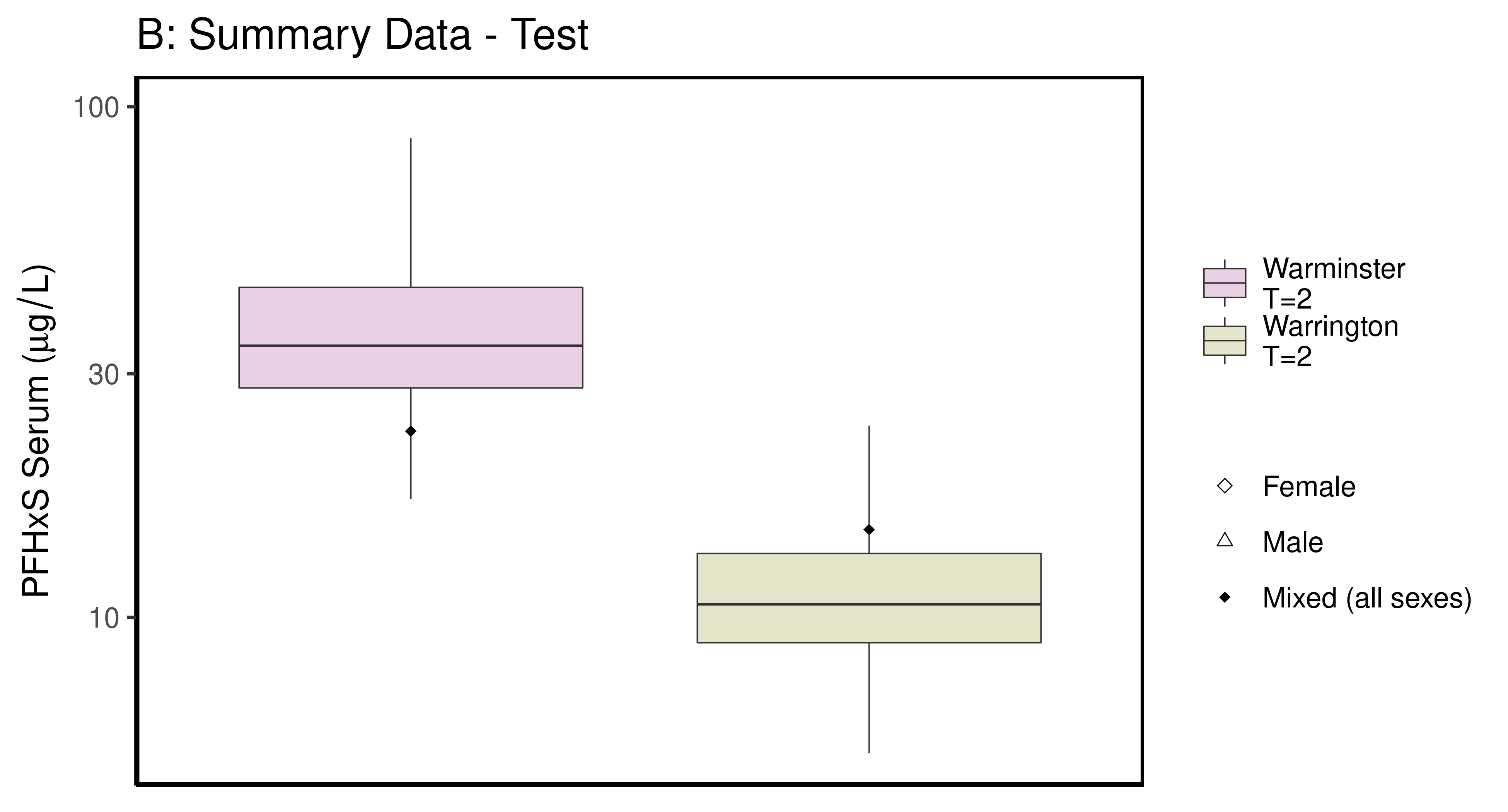
ggsave(here ("output-plots", paste0( sa, "SummaryTrainDataboxplot", pfas\_name,".png")), plt\_train,dpi=600)

## Saving 6.5 x 3.5 in image

### make Test plot  
df\_d\_test <- df\_check %>%   
 filter((Train\_Test == "Test") &   
 ((Output\_Var == "M\_Cbgd\_Css") | (Output\_Var == "M\_Cserum"))) %>%  
 mutate\_if(is.factor, as.character) %>% # drop factor levels unused  
 mutate(Dataset.Time = factor(Dataset.Time))   
  
aes\_lut\_df\_d\_test <- df\_d\_test %>% select(City, datatype,Time, Time.desc, legend\_label) %>%   
 inner\_join(aes\_lut ) %>%   
 select(-Train\_Test) %>% ungroup () %>% unique ()

## Joining, by = c("City", "datatype", "Time", "Time.desc", "legend\_label")

plt\_test <- plot\_sum\_boxplot(dframe = df\_d\_test, aes\_lut= aes\_lut\_df\_d\_test,   
 gtitle="B: Summary Data - Test", facets = FALSE, pfas\_nom = pfas\_name)  
 print(plt\_test)



ggsave(here ("output-plots",paste0( sa, "SummaryTestDataboxplot",pfas\_name,".pdf")), plt\_test,dpi=600)

## Saving 6.5 x 3.5 in image

ggsave(here ("output-plots",paste0( sa, "SummaryTestDataboxplot",pfas\_name,".png")), plt\_test,dpi=600)

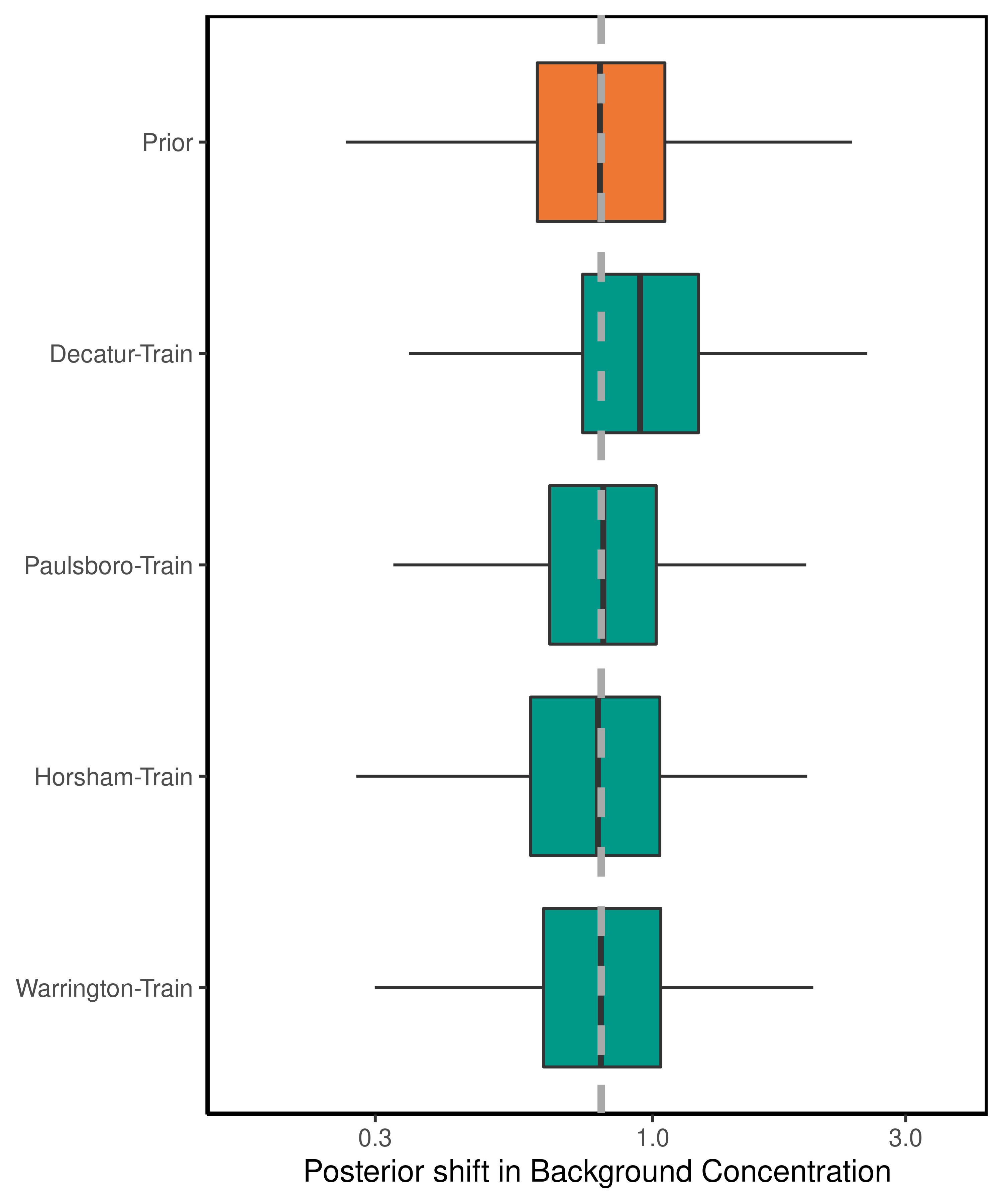
## Saving 6.5 x 3.5 in image

## PFHxS

### Background posteriors

Shows shift in background estimate.

gmscale<-0.8  
  
dat <- multicheck$parms.samp[,grep("M\_ln\_Cbgd",names(multicheck$parms.samp))]  
datasetnames <- as.character(unique(calibdata$dataset))  
datasetnames <- gsub(" Train","-Train",datasetnames)  
datasetnames <- gsub(" Test","-Train",datasetnames)  
datasetnames <- gsub(" M","",datasetnames)  
datasetnames <- gsub(" F","",datasetnames)  
datasetnames<-datasetnames[!duplicated(datasetnames)]  
datasetnames <- datasetnames[grep("Train",datasetnames)]  
names(dat) <- datasetnames  
dat <- dat[,grep("Train",names(dat))]  
dat.df <- pivot\_longer(dat,1:ncol(dat))  
dat.df <- rbind(dat.df,  
 data.frame(name="Prior",value=rnorm(5000,m=log(gmscale),sd=0.4055)))  
dat.df$name <- factor(dat.df$name,levels=rev(  
 c("Prior",datasetnames[grep("Train",datasetnames)])))  
dat.df$value <- exp(dat.df$value)  
  
p<-ggplot(dat.df)+  
 # geom\_violin(aes(x=name,y=value,fill=name=="Prior"))+  
 geom\_boxplot(aes(x=name,y=value,fill=name=="Prior"),outlier.shape=NA)+  
 scale\_y\_log10()+coord\_flip()+  
 scale\_fill\_manual(name=NULL,   
 values=c("#009988", "#EE7733" )) +  
 theme\_classic() +   
 theme(axis.title.y.left = element\_blank())+  
 geom\_hline(yintercept = gmscale, color="darkgrey", linetype = 2, size = 1.25)+  
 theme(legend.position="none",  
 panel.background = element\_rect(color="black",size=1))+  
 ylab("Posterior shift in Background Concentration")  
  
print(p)



ggsave(here ("output-plots",paste0( sa, "PFHxS\_GM\_Cbgd.pdf")) ,p,dpi=600)

## Saving 5 x 6 in image

ggsave(here ("output-plots",paste0( sa, "PFHxS\_GM\_Cbgd.png")) ,p,dpi=600)

## Saving 5 x 6 in image

### Half-life

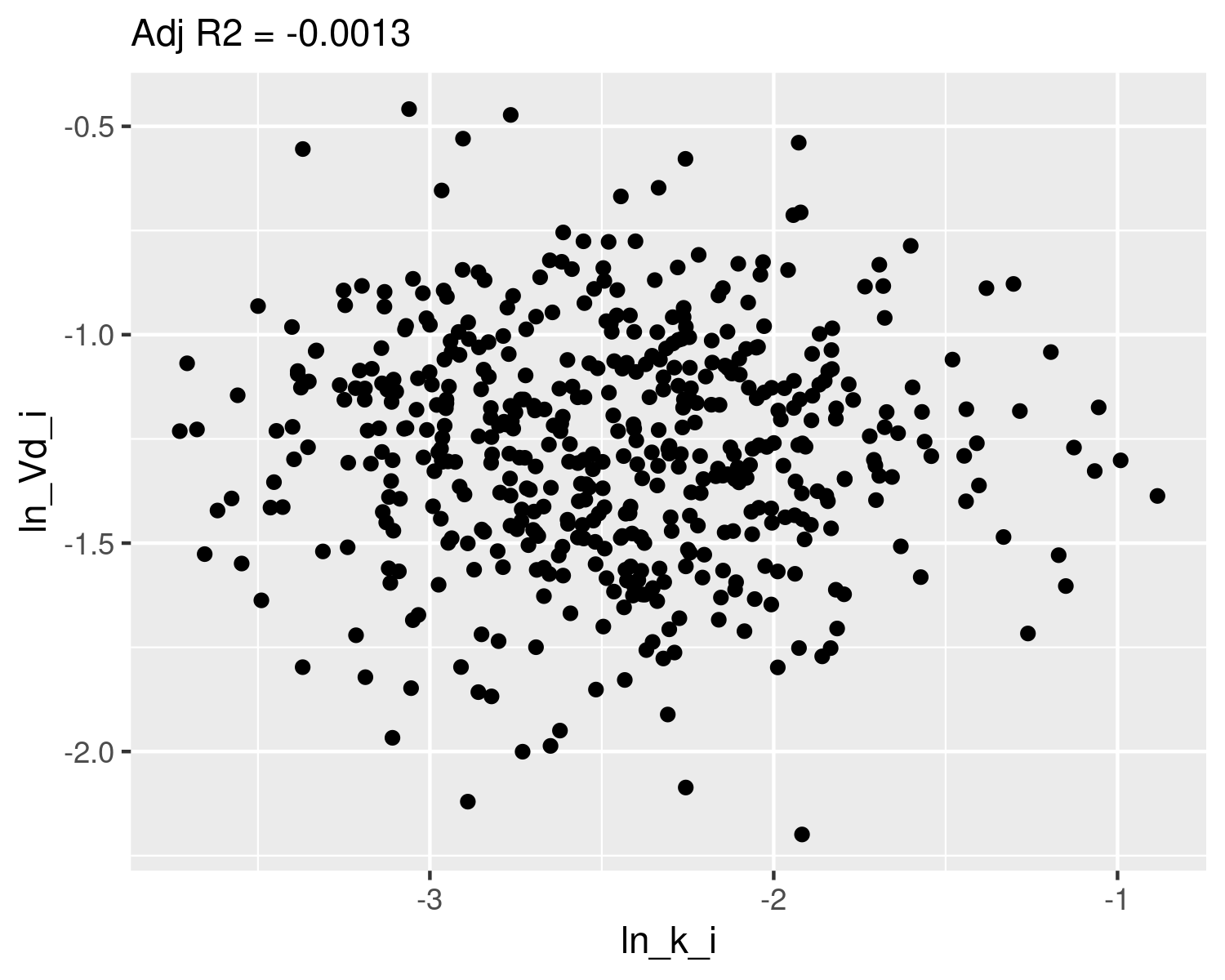
For PFHxS, the population GM of the half-life has a posterior distribution that is narrower than the prior, with a posterior median (95% CI) estimate of 3.06 (2.16-4.37) years. The population GSD posterior is larger than the prior at 1.47(1.44-1.75).

dat <- multicheck$parms.samp[,c("M\_ln\_k.1.","V\_ln\_k.1.", "M\_ln\_Vd.1.", "SD\_ln\_Vd.1.")]  
names(dat) <- c("M\_ln\_k(1)","V\_ln\_k(1)", "M\_ln\_Vd(1)", "SD\_ln\_Vd(1)")  
   
set.seed(3.14159)  
dat$z\_ln\_k <- rnorm(nrow(dat))  
dat$z\_ln\_Vd <- rnorm(nrow(dat))  
dat %>% rename\_()

## Warning: `rename\_()` was deprecated in dplyr 0.7.0.  
## Please use `rename()` instead.  
## This warning is displayed once every 8 hours.  
## Call `lifecycle::last\_lifecycle\_warnings()` to see where this warning was generated.

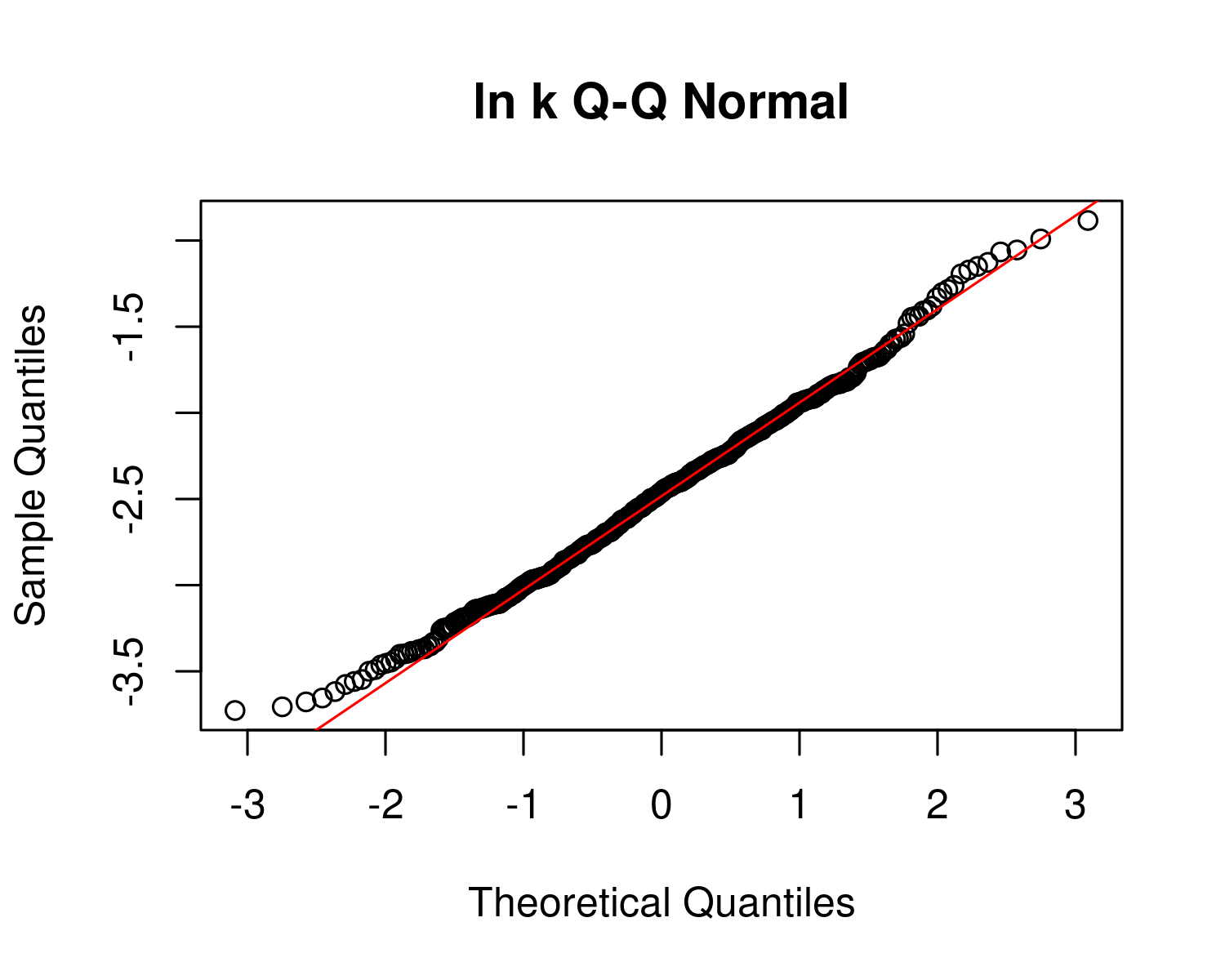
## M\_ln\_k(1) V\_ln\_k(1) M\_ln\_Vd(1) SD\_ln\_Vd(1) z\_ln\_k z\_ln\_Vd  
## 12503 -2.75291 0.2041830 -1.412470 0.191222000 -0.961933416 -2.139841913  
## 15901 -2.72905 0.3388370 -1.262040 0.095982100 -0.292525723 -1.263479244  
## 17543 -2.67035 0.2659090 -1.390400 0.286419000 0.258788216 0.083307969  
## 18721 -2.31531 0.1939420 -1.200790 0.132528000 -1.152131886 0.188325130  
## 11662 -2.51794 0.1579330 -1.097060 0.082538200 0.195782826 0.189814193  
## 13403 -2.83577 0.1720120 -1.244460 0.207939000 0.030123945 0.217615352  
## 13192 -2.36731 0.2929010 -1.090200 0.063831600 0.085417732 -0.648814142  
## 15663 -2.35551 0.1869580 -1.423570 0.188954000 1.116610213 0.254774383  
## 1246 -2.48740 0.2047250 -1.167350 0.026815800 -1.218857416 -0.468830015  
## 16132 -2.28657 0.2310820 -1.212640 0.007859580 1.267368722 -1.192510054  
## 10602 -2.76543 0.2356490 -1.451400 0.008926270 -0.744781596 0.121228235  
## 10563 -2.46342 0.1863690 -1.042400 0.109301000 -1.131218571 -1.035932658  
## 15932 -2.70822 0.2264000 -0.840291 0.133664000 -0.716358490 -0.191888295  
## 1114 -2.74916 0.1426340 -1.200700 0.064772000 0.252652370 -0.970546336  
## 1699 -2.69943 0.2717080 -1.233420 0.034752200 0.152045707 0.057771938  
## 1141 -2.74801 0.2122340 -1.339750 0.076209900 -0.307656430 -2.110692675  
## 13383 -2.96621 0.1474260 -1.060710 0.021755100 -0.953017331 0.978176274  
## 12832 -2.69611 0.2346190 -1.002450 0.113141000 -0.648242811 0.371021737  
## 19552 -2.51489 0.3228540 -1.223570 0.030571700 1.224313624 0.720910931  
## 14423 -2.40016 0.1580630 -1.223940 0.064784600 0.199811608 1.879594880  
## 12372 -2.65261 0.1251740 -1.018450 0.030116200 -0.578483722 -0.396316112  
## 12042 -2.29333 0.1725240 -1.497460 0.131016000 -0.942300733 0.110322947  
## 10473 -2.32319 0.1890660 -1.383060 0.158345000 -0.203728180 -0.593140881  
## 1509 -2.87891 0.2592500 -1.324400 0.221668000 -1.666474840 0.419621770  
## 1552 -2.34423 0.2684330 -1.264770 0.072917700 -0.484455109 -0.545738743  
## 19513 -2.04339 0.1517460 -1.563720 0.002395420 -0.741072661 1.160921499  
## 17471 -2.44949 0.1714430 -1.134580 0.009712250 1.160615779 0.639817834  
## 18092 -2.92293 0.2649380 -0.749883 0.212432000 1.012067125 -0.122020443  
## 1613 -2.62062 0.1960020 -1.583610 0.051115900 -0.072078474 0.184645026  
## 14383 -2.07859 0.1494270 -1.473980 0.147827000 -1.136782298 -0.517806023  
## 14263 -2.35721 0.1679390 -1.817640 0.286360000 0.900624729 0.067988352  
## 18291 -2.16099 0.1645120 -1.694750 0.054638300 0.851770447 -0.184797156  
## 1945 -2.41898 0.2266590 -1.056890 0.050055700 0.727715174 -1.403691615  
## 19613 -2.12554 0.2178070 -1.204940 0.374359000 0.736502146 0.229740706  
## 1842 -2.26703 0.2282840 -1.489630 0.185089000 -0.352129617 -0.889081301  
## 1873 -2.30463 0.1012290 -1.017190 0.105298000 0.705515513 -0.160401177  
## 19423 -2.29065 0.1188390 -1.376410 0.095319000 1.300357989 -0.242136794  
## 19352 -2.83644 0.1681330 -1.866550 0.041875600 0.038252014 -0.028837196  
## 11812 -2.36280 0.1702990 -1.223170 0.385055000 -0.979283770 -0.316215474  
## 14982 -2.19641 0.1326170 -1.220060 0.117067000 0.793761231 -0.416160087  
## 11263 -2.66246 0.1561150 -1.449460 0.154783000 0.786506872 -1.023895957  
## 1028 -3.11690 0.4711690 -1.141750 0.094573500 -0.310463131 1.099495275  
## 1051 -2.36686 0.2249400 -1.372110 0.141351000 1.698884846 0.817712470  
## 17222 -2.58380 0.2625230 -1.444480 0.194389000 -0.794593709 0.168875510  
## 1747 -2.39220 0.2981630 -1.534210 0.165598000 0.348437716 0.038665440  
## 1284 -2.63369 0.2122760 -1.286310 0.054794600 -2.265401074 1.078174892  
## 1564 -2.38815 0.2185380 -1.107500 0.116141000 -0.162205279 0.379427297  
## 18632 -2.39139 0.1617920 -1.278670 0.067979300 1.130864991 -1.078174559  
## 19142 -2.35800 0.1688640 -1.359770 0.006053020 -0.455545976 0.188763468  
## 15793 -2.33111 0.1659800 -1.205790 0.022403200 -0.899166316 1.583133508  
## 1126 -2.30517 0.1680520 -1.373240 0.174320000 0.726838902 -0.250112295  
## 16502 -2.46539 0.1421920 -1.093550 0.137344000 -0.809440902 0.342994219  
## 15103 -2.22348 0.1332940 -1.259620 0.085270400 0.267085116 -0.124701883  
## 11251 -2.29728 0.1875460 -1.587920 0.097563800 -1.737263711 -0.993314763  
## 11732 -2.68818 0.1392170 -1.631840 0.087352100 -1.411425136 -1.021439085  
## 18382 -2.90847 0.1900410 -1.513860 0.051897200 -0.453551227 0.840642928  
## 13433 -2.94800 0.1661000 -1.828890 0.036808300 -1.035491275 0.849573118  
## 16403 -2.48109 0.1899540 -1.169690 0.049453500 1.362142893 0.469618942  
## 11871 -2.81063 0.2940520 -1.020030 0.010292700 0.917456737 -1.269655261  
## 1988 -2.30920 0.2368420 -1.316840 0.144035000 -0.785142161 -1.105120219  
## 1589 -2.55007 0.1595040 -1.389440 0.204108000 0.573518173 -1.897945965  
## 1449 -2.69991 0.2258280 -1.252910 0.158285000 0.918196208 0.491787265  
## 1232 -2.52956 0.1945000 -1.265820 0.207718000 0.256287273 -0.704722317  
## 10171 -2.43987 0.2402250 -1.119880 0.061807300 0.351966556 1.777576130  
## 1259 -2.04854 0.2679840 -1.376630 0.094045200 1.174337357 -0.245389029  
## 18503 -2.58795 0.1304050 -1.068760 0.065600100 -0.480846375 -2.111252318  
## 10801 -2.81228 0.2198140 -1.138300 0.154019000 -0.418829722 -0.585314536  
## 15263 -2.88365 0.1925590 -1.408220 0.401838000 0.955112803 -0.517903726  
## 1042 -2.44489 0.1455260 -1.534480 0.090103500 -1.289006611 0.513919145  
## 18363 -2.47945 0.2458030 -1.447640 0.061136700 0.186197433 -0.626426919  
## 19072 -2.24737 0.2172030 -1.037940 0.532891000 -0.031325502 0.192142595  
## 16632 -2.54620 0.1326950 -1.155580 0.241190000 0.467097310 -1.427277992  
## 1401 -2.39196 0.1566060 -1.226810 0.111998000 1.024197674 0.400991028  
## 14691 -1.96276 0.2302790 -1.829950 0.067186500 0.267358452 1.161564956  
## 17041 -2.44979 0.2460070 -0.655498 0.024611200 0.231826103 0.326346805  
## 19822 -2.61613 0.1584750 -1.507100 0.085059700 0.747592465 -1.014982273  
## 13613 -2.53506 0.2380780 -1.132490 0.215915000 1.217068511 0.101258362  
## 11013 -2.29436 0.2116990 -1.540040 0.069525200 0.383358345 0.991893279  
## 15791 -2.79579 0.1583310 -1.293580 0.149252000 -0.988052822 1.102650759  
## 16531 -2.52777 0.2901210 -0.861600 0.381616000 -0.156852910 0.280375017  
## 1970 -1.93855 0.2062580 -1.557290 0.112599000 1.735535216 -0.406177410  
## 16331 -2.46581 0.1775700 -1.576200 0.052572400 -0.352298306 1.286412690  
## 1242 -2.96545 0.2232860 -1.168150 0.195051000 0.688640044 -0.254030579  
## 15251 -2.25203 0.2001350 -1.137520 0.153892000 1.224406096 -1.144448704  
## 1653 -2.47856 0.2267720 -1.097500 0.150603000 0.794296303 0.266410910  
## 1140 -3.10974 0.2057520 -1.474390 0.193588000 -0.006402398 0.636855748  
## 17771 -2.71548 0.1695250 -1.357270 0.122985000 0.219150635 -1.402115093  
## 18012 -2.20468 0.2212810 -1.484250 0.307921000 -0.886463751 -1.511824827  
## 14062 -2.61130 0.1973600 -1.555340 0.001521500 0.439760291 -0.301192859  
## 1964 -2.32689 0.1492140 -1.415530 0.006501350 -0.886389751 0.491361149  
## 18413 -2.67592 0.2644960 -1.418900 0.196740000 -0.853818454 -0.898156496  
## 13791 -2.22297 0.1853230 -1.412380 0.364550000 -0.989994331 -1.575054814  
## 15832 -2.10205 0.2009170 -1.533270 0.110261000 -0.650877737 -0.503256667  
## 1060 -2.76987 0.2081140 -1.040110 0.062290300 1.053946660 -0.622908441  
## 12073 -2.38674 0.2187970 -1.414980 0.104553000 -0.390878033 -0.684829990  
## 15683 -2.37177 0.1980320 -1.554080 0.281385000 -0.070586394 -0.221516764  
## 1693 -2.33933 0.3009600 -1.277810 0.039864200 -0.462050809 0.388050833  
## 17531 -2.12322 0.1466000 -1.261170 0.225183000 0.540908267 -0.806316456  
## 10311 -2.37840 0.2331660 -1.258570 0.011139100 0.931634971 -0.547213054  
## 18041 -2.36913 0.1680350 -1.293030 0.329758000 -0.209274345 1.213494144  
## 19672 -2.57831 0.2934200 -1.392010 0.100753000 0.617350048 -0.417013230  
## 18351 -2.07196 0.2859340 -1.773640 0.067483100 -0.405077513 0.164593185  
## 13741 -2.44427 0.2304690 -1.270080 0.103919000 1.053103763 -1.569627007  
## 11892 -2.54864 0.2530710 -1.011650 0.003788400 0.602284246 1.466675890  
## 18762 -2.80078 0.1509990 -1.590590 0.009532450 1.017461177 -1.496085449  
## 1808 -2.30820 0.1345940 -1.650170 0.053128200 0.608167318 -1.150289019  
## 15861 -2.53035 0.2334230 -1.232060 0.097601000 0.206735995 -2.023009144  
## 12312 -2.52164 0.2281500 -1.353320 0.088427100 -1.897727292 -0.681652395  
## 13473 -2.60209 0.2255960 -1.309040 0.007423090 -0.682582832 0.507010414  
## 1040 -2.65433 0.2798510 -1.355010 0.135711000 0.481338415 0.747111894  
## 17573 -2.61984 0.2490640 -1.188020 0.323572000 -0.463031038 0.175989014  
## 19872 -2.25348 0.1875930 -1.510300 0.259134000 -0.279741696 -0.437523579  
## 1953 -2.77582 0.2330970 -1.298360 0.017528100 -0.413690145 0.942726154  
## 15883 -2.73255 0.2387910 -1.224400 0.054648900 1.618766521 0.891044960  
## 1663 -2.36710 0.2305610 -1.283490 0.130845000 -0.721055711 -1.691785958  
## 18913 -2.18828 0.2151700 -1.508750 0.156973000 -0.453093157 -0.712147393  
## 1005 -2.56690 0.1906970 -1.293830 0.191563000 0.014257161 -0.332268892  
## 1878 -2.30489 0.1841830 -1.424570 0.045400100 0.215764621 0.983033453  
## 15553 -2.78380 0.2394740 -1.551280 0.200838000 0.188870195 -0.988041784  
## 19433 -2.36880 0.3129100 -1.342660 0.120168000 -0.050148494 0.265546446  
## 15642 -2.97237 0.2083640 -1.304700 0.304446000 -1.495419632 -0.728186155  
## 13873 -2.31278 0.2007360 -1.217850 0.239692000 0.367837753 1.375405013  
## 17882 -2.27076 0.1736600 -1.564410 0.116566000 0.517144022 -0.599264333  
## 1857 -2.33889 0.1434540 -0.866820 0.085612400 -0.484335465 -0.272173202  
## 1103 -2.92392 0.2305130 -1.091080 0.075357800 0.674855616 0.405324773  
## 12581 -2.40805 0.1438660 -1.444150 0.023356100 -0.762448605 0.846734123  
## 1974 -2.19802 0.2013320 -1.594180 0.075405100 0.386073791 0.518125984  
## 13822 -2.31772 0.2787760 -1.576910 0.011320200 -0.664003345 1.579413771  
## 1101 -2.80045 0.2758540 -1.065320 0.029147400 -1.724344195 -0.112951698  
## 1198 -2.35923 0.1808900 -1.090600 0.060685800 1.156319079 -0.484652932  
## 19762 -2.48823 0.1958400 -1.003800 0.069038700 0.693506590 -2.377853826  
## 18271 -2.23329 0.2633250 -1.577230 0.167681000 0.143156356 -0.636048007  
## 18911 -2.52300 0.2552750 -1.158860 0.062347400 1.492813560 0.035663770  
## 19111 -2.42786 0.1817920 -1.124380 0.067765000 -1.632153479 -0.109166877  
## 1374 -2.56186 0.1729880 -1.211040 0.116528000 0.127846023 -1.871305162  
## 1920 -2.19961 0.2138380 -1.496300 0.297740000 -2.403663727 -0.079333090  
## 1504 -2.41772 0.1536940 -1.039570 0.223998000 1.443928256 -0.318596856  
## 16771 -2.31342 0.2179900 -1.309680 0.121366000 -0.878893050 0.118181119  
## 19693 -2.42895 0.1916690 -1.137700 0.089808500 -1.306438327 0.532403631  
## 13331 -2.46457 0.2337390 -0.922659 0.241005000 -0.877198993 -0.197683940  
## 18422 -2.37022 0.1760710 -1.118770 0.161377000 -1.164380459 1.664379027  
## 11663 -2.52820 0.1939240 -0.985310 0.012060500 -1.982347680 0.314495086  
## 13082 -2.46787 0.2529500 -0.760383 0.176931000 -0.989944235 0.601591759  
## 14941 -2.49005 0.2774060 -1.096670 0.037465000 -0.151684601 -1.429557536  
## 1073 -2.72484 0.1586700 -1.069150 0.170916000 0.912506791 -0.472999146  
## 1781 -2.52177 0.1699500 -0.953653 0.198558000 0.407669808 -0.487888829  
## 15091 -2.38368 0.2209770 -1.339460 0.064258800 -1.242184375 1.027388649  
## 13413 -2.69624 0.1876340 -1.519910 0.072977900 -0.642694412 -1.095763686  
## 12463 -2.49994 0.1202310 -1.081360 0.001063220 1.930243689 -0.916445608  
## 11261 -2.23463 0.1709890 -1.460740 0.034025800 0.410199364 1.054094779  
## 10231 -2.35073 0.2178260 -1.089260 0.060503000 -1.291349321 -1.439834035  
## 16911 -2.58632 0.2792290 -1.113420 0.134728000 2.635045375 0.532564109  
## 12011 -2.29810 0.3252870 -1.232520 0.035037200 0.487072288 -1.046928774  
## 14342 -2.84087 0.1372380 -1.452200 0.071899500 0.853892318 0.087385881  
## 1170 -2.33214 0.1450430 -2.173590 0.146048000 1.088442705 -0.174111691  
## 17372 -2.86449 0.3391110 -1.374310 0.084368400 0.226013957 -0.533826331  
## 1839 -2.49799 0.2125070 -1.201020 0.074374400 0.068198838 0.099317747  
## 11531 -2.42986 0.1638310 -1.003030 0.095948800 -0.984815533 -1.020107099  
## 16811 -2.40424 0.1833770 -1.374040 0.170499000 -1.310854422 0.430041832  
## 1825 -2.88936 0.2196440 -1.095250 0.140126000 2.464055335 1.502847759  
## 12443 -2.61625 0.2365260 -0.998579 0.062891500 -0.665428070 -0.270760795  
## 12893 -2.90385 0.2400490 -1.276770 0.370750000 0.912862648 0.871555415  
## 1694 -2.41987 0.2208380 -1.338730 0.081638700 0.964664226 -1.216293810  
## 11843 -2.40482 0.2881710 -1.140490 0.079717400 1.608002937 -1.892738669  
## 15731 -2.61519 0.2731750 -1.179770 0.196440000 1.835399525 -0.822382571  
## 15783 -2.33111 0.1680370 -1.282040 0.022403200 0.702462733 0.751782015  
## 11601 -2.20933 0.1909580 -1.093280 0.172722000 1.217854079 0.773574518  
## 17572 -2.67874 0.2066160 -1.144130 0.006625640 -1.123653829 -1.787532258  
## 1062 -2.63788 0.1429220 -1.588510 0.248611000 0.668330116 0.088817469  
## 12182 -2.58992 0.2829290 -0.679101 0.188839000 1.216411488 -0.180052904  
## 18073 -2.26493 0.1925540 -1.239750 0.055304500 0.234575354 -1.471714902  
## 1773 -2.73778 0.2612380 -1.174970 0.021498200 -0.418696588 0.490094947  
## 12412 -3.08925 0.2760230 -1.286600 0.045970100 0.238220090 -0.397833206  
## 14813 -2.36180 0.2256620 -1.095400 0.200376000 -0.550588192 -0.169679442  
## 19503 -2.04339 0.1800320 -1.563720 0.235967000 -0.500602769 -2.214982820  
## 1029 -2.61957 0.1955960 -1.344070 0.025136000 1.163897492 1.008145236  
## 1473 -2.26200 0.1863250 -1.579330 0.179233000 2.155536954 0.524260695  
## 1430 -2.18280 0.1285830 -1.392400 0.051597700 -1.709157009 0.264820635  
## 12303 -2.46105 0.1804300 -1.000730 0.081049900 -1.600822615 -0.388256354  
## 1738 -2.57684 0.1273710 -1.264670 0.098407900 -1.038553413 -0.399451920  
## 11051 -2.50241 0.2564160 -1.644170 0.022408200 0.323094221 0.227146190  
## 18882 -2.58510 0.1654160 -1.335280 0.268573000 -0.888847154 -0.611156980  
## 14953 -2.68345 0.2493870 -1.049420 0.151422000 0.393678982 0.540650478  
## 17223 -2.65136 0.1719400 -0.976722 0.220725000 0.236541468 0.909955002  
## 10543 -2.36751 0.1754600 -1.512710 0.083097200 -0.430496813 1.406622276  
## 19871 -2.79660 0.2455330 -1.190730 0.043775900 -0.547933126 -0.755587121  
## 15571 -2.84325 0.1674940 -1.092950 0.060062900 -1.322251659 -0.020592429  
## 19333 -2.77915 0.2003730 -0.995358 0.160267000 0.682126749 0.137858516  
## 1357 -2.45709 0.1673250 -1.374130 0.225119000 2.162789482 -0.921540392  
## 12813 -2.50382 0.2022030 -0.997649 0.071132000 -0.416669649 0.576523431  
## 1222 -2.48637 0.1958910 -1.425170 0.105178000 -1.357317897 0.298893720  
## 19032 -2.52333 0.1301280 -1.163360 0.258227000 -0.671226462 -0.862301037  
## 1190 -2.72165 0.1996390 -1.271090 0.361538000 0.649918243 -0.810138091  
## 17662 -2.32062 0.1957360 -1.275350 0.152257000 0.771291175 0.472555898  
## 1570 -2.38758 0.2479160 -1.256770 0.144942000 2.676631927 0.568499025  
## 12933 -2.40723 0.2095010 -1.214140 0.077553300 -1.370871421 1.413098269  
## 17162 -2.51899 0.1386810 -1.319660 0.249399000 0.057759145 0.058817380  
## 18321 -2.25596 0.1757300 -1.334810 0.046047300 -0.197067508 -0.583447921  
## 13532 -2.45655 0.3188720 -1.176650 0.351083000 -1.261517626 0.270123085  
## 14012 -2.69239 0.1631220 -0.940604 0.043263600 -0.662442562 1.078046441  
## 19571 -2.48342 0.1428350 -1.319680 0.053334200 -1.332352324 -0.146579622  
## 12991 -2.03512 0.1534680 -1.639280 0.177007000 0.277323621 -0.635253591  
## 17063 -2.19833 0.2167940 -1.368050 0.056133500 1.085533849 0.524084655  
## 19203 -2.54319 0.1844310 -1.230140 0.066937700 -1.642718489 1.105146081  
## 14513 -2.61139 0.2934240 -1.698790 0.316797000 -0.457180258 -0.501011903  
## 18052 -2.54457 0.2153740 -1.002320 0.009872310 1.461171845 0.424192601  
## 1957 -2.69954 0.2289580 -0.857036 0.068936900 -1.672532602 -1.077357662  
## 1965 -2.07979 0.1845780 -1.252350 0.012800500 1.561095996 -0.625837491  
## 13273 -2.80150 0.2126280 -1.659280 0.026146200 -1.493370648 0.840120571  
## 1789 -3.07443 0.1679580 -1.123210 0.372005000 -1.182449141 -0.059466609  
## 13492 -2.24859 0.1408350 -1.336770 0.101983000 -0.356586881 -0.077217546  
## 12513 -2.69843 0.2244530 -0.957387 0.059339500 -0.915640125 0.416709627  
## 14672 -2.55776 0.2750330 -1.546270 0.047430700 0.849485485 -1.379914328  
## 18373 -2.48439 0.2522700 -1.703710 0.200188000 -0.489607238 -1.482149933  
## 11793 -2.59881 0.1317320 -1.231870 0.025842500 0.727983210 0.122071411  
## 1361 -2.33281 0.2099320 -1.041780 0.181210000 -0.619313219 1.195887446  
## 15123 -2.43398 0.1522670 -1.362070 0.298861000 0.343735543 -0.254062405  
## 18791 -2.27984 0.2337370 -1.296400 0.189478000 1.813625401 -0.344407316  
## 10681 -2.53237 0.2147390 -1.758890 0.007090130 1.453226830 -1.809799151  
## 11143 -2.05439 0.2528240 -1.181710 0.125576000 0.412429443 -1.630895018  
## 1194 -3.07044 0.2800410 -1.227560 0.072222700 0.201975959 -0.271965872  
## 1330 -2.48884 0.2145210 -1.224220 0.042406900 1.684346898 -1.790901346  
## 18142 -2.79597 0.2179330 -0.864544 0.283637000 -0.964824478 -0.228499227  
## 1934 -2.32740 0.1900660 -1.124510 0.150563000 0.662155194 1.784240948  
## 1622 -2.76439 0.2363920 -1.654290 0.108306000 -0.553343994 -0.166313575  
## 14913 -2.80855 0.3796350 -0.978326 0.123809000 2.444205231 0.809087394  
## 14073 -2.53267 0.1983580 -1.504490 0.126207000 -0.304629836 -0.972329346  
## 19561 -2.56215 0.1836790 -1.472510 0.053992600 -0.117369869 -1.952502354  
## 1231 -2.66040 0.1734920 -0.747964 0.095120300 0.969428787 1.786422520  
## 1849 -2.29940 0.1771320 -1.404900 0.284743000 0.590939630 1.312987748  
## 15711 -2.47559 0.2341020 -0.888482 0.387690000 -1.208958402 1.108674735  
## 13711 -2.24576 0.2144270 -1.349430 0.137888000 -0.301928423 -0.993200725  
## 17031 -2.49576 0.1395050 -0.870143 0.002173620 0.401260475 0.526099803  
## 16543 -2.32161 0.2591240 -1.214730 0.019918400 -1.688368241 -0.768071012  
## 17671 -2.48518 0.2333450 -1.118780 0.191145000 -0.712992471 0.529000051  
## 19841 -2.65009 0.2510610 -1.218210 0.018979100 -0.254300337 0.142537992  
## 19063 -2.52320 0.1705120 -1.011170 0.019330300 -1.001566828 -1.540493313  
## 17851 -3.01154 0.2392500 -1.210270 0.263259000 -0.924100354 -0.776809099  
## 1478 -2.29989 0.2088490 -1.056580 0.012835500 0.440245569 -3.053299791  
## 18823 -2.39892 0.1768400 -1.156130 0.100688000 -0.326816611 0.871474290  
## 10932 -2.36588 0.2231800 -1.713160 0.067718600 -0.274475875 0.190962149  
## 12382 -2.79370 0.1604890 -0.873608 0.041233900 1.284142059 0.849572629  
## 11613 -2.38011 0.2012120 -1.112140 0.074383700 0.304211777 -0.563481733  
## 12701 -2.67122 0.1989490 -0.943063 0.094049400 1.273682656 1.204416928  
## 14801 -2.39327 0.2387860 -1.105520 0.034982000 1.033447932 1.704492687  
## 13023 -2.46962 0.2681240 -1.064080 0.258586000 0.665384156 -0.058008606  
## 12123 -2.65180 0.1536720 -0.957012 0.028476100 -0.313328109 0.770876717  
## 10272 -2.34482 0.1445510 -1.339490 0.003875490 0.464076685 -0.027690324  
## 18622 -2.37285 0.1617920 -1.148460 0.103156000 0.478605860 1.303131096  
## 15523 -2.40395 0.1445440 -1.168240 0.014854300 2.536235795 -0.711585886  
## 15201 -2.22287 0.1650790 -1.457750 0.032670600 -0.535063085 -0.771192049  
## 14391 -2.72062 0.2865870 -1.494570 0.054853500 1.368106222 -1.338907825  
## 1658 -2.46578 0.1671910 -1.217310 0.005706140 0.141844290 0.408160652  
## 17003 -2.42203 0.2178880 -1.564510 0.032914800 -0.782814969 0.206537589  
## 1316 -2.31656 0.3011380 -1.141570 0.185141000 1.881518733 -0.224476304  
## 15442 -2.16743 0.1943220 -1.149900 0.003636570 0.552922865 -1.504028164  
## 1008 -2.56075 0.1993120 -1.095590 0.034944400 -1.844856583 0.238798379  
## 18082 -2.69110 0.2121310 -1.056010 0.155655000 -0.550307914 -0.440094481  
## 1931 -2.40877 0.1590850 -1.430710 0.002067050 -0.026014759 1.085948334  
## 1192 -2.55122 0.2439360 -1.475330 0.186027000 -2.159250362 0.287937278  
## 16292 -2.61694 0.2164450 -1.328060 0.031774700 1.770600080 -0.592582238  
## 12013 -2.53680 0.1754140 -0.757388 0.049503200 0.760473482 -1.030482341  
## 1999 -2.64652 0.1748310 -1.545560 0.108084000 -0.484542846 0.722738858  
## 15621 -2.92264 0.2101960 -1.250310 0.038002000 -0.472763899 -0.814832560  
## 17171 -2.43860 0.1648300 -1.423750 0.334111000 0.331672075 0.414635154  
## 19781 -2.19796 0.1706480 -1.443240 0.003208060 -0.972300798 -0.192636529  
## 11202 -2.66117 0.2185130 -1.753420 0.107003000 -0.531244209 -0.410108658  
## 16253 -2.76171 0.1794870 -1.186740 0.152557000 0.062036740 0.206361537  
## 16722 -2.74387 0.2279050 -1.196210 0.050700000 0.103529988 0.284630186  
## 18962 -2.35693 0.1417370 -1.133050 0.211020000 1.169799329 -0.604884836  
## 1076 -2.34083 0.2741940 -1.077470 0.145279000 0.394189201 0.583220593  
## 1501 -2.43717 0.1698760 -1.516910 0.368256000 -0.134102361 0.280145010  
## 10951 -2.77197 0.2392180 -1.310750 0.190830000 -0.379875566 1.315610342  
## 15121 -2.44239 0.2278510 -0.936212 0.069268700 1.278666582 -1.452676821  
## 1006 -2.43982 0.1924730 -1.350980 0.099959500 -0.699332309 -1.158009146  
## 16093 -2.59834 0.1631580 -1.751650 0.105190000 0.564250526 -0.047178285  
## 18592 -2.29741 0.2941710 -1.093490 0.080981300 0.302941956 0.162912346  
## 16783 -2.36561 0.1679740 -1.159770 0.039779500 0.215441157 0.940933129  
## 17292 -2.56554 0.3367450 -0.812416 0.033911100 0.921002999 -0.392834083  
## 11801 -2.36054 0.1937110 -1.126520 0.106244000 -0.887551892 -0.562332042  
## 16312 -2.42576 0.1370540 -1.693180 0.068768900 -0.002893014 1.502073402  
## 1104 -2.89792 0.2433030 -1.085230 0.072115100 -0.195173624 -0.477261637  
## 13902 -2.02691 0.2392080 -1.456340 0.138713000 -2.234678901 -0.750163568  
## 1347 -2.45531 0.2235150 -0.942613 0.263366000 0.478322985 -1.018058650  
## 1769 -2.78120 0.2054710 -1.005110 0.022875900 1.622692015 -1.054689491  
## 11621 -2.19739 0.1538330 -1.173330 0.009318130 -1.196074086 -0.641345254  
## 18953 -2.29140 0.1755190 -1.402240 0.196519000 1.659509162 1.404537373  
## 10771 -2.66313 0.1235040 -1.648980 0.295477000 0.204545434 -0.065751336  
## 1117 -2.40776 0.2822140 -1.183470 0.040711800 1.385975853 -0.049392617  
## 10782 -2.51912 0.2145630 -1.040310 0.091891000 -0.857811795 0.510827457  
## 16763 -2.58130 0.2178100 -1.179090 0.083672400 0.685862076 0.285062593  
## 12292 -2.89422 0.2275310 -1.305650 0.084636100 -0.581662540 -0.044179742  
## 14191 -2.78723 0.2241930 -1.343880 0.084503900 -0.684552910 2.160619408  
## 15113 -2.24135 0.1100910 -1.300100 0.051129300 0.378472922 0.251437855  
## 19862 -2.36400 0.1875930 -1.467650 0.204736000 1.312957410 -0.757802743  
## 16053 -2.48198 0.1706310 -1.834090 0.047798600 0.314021521 2.029114324  
## 11031 -2.55610 0.2363130 -1.678340 0.074828800 -0.502185174 -0.759754169  
## 10963 -2.60872 0.1995160 -1.254700 0.189131000 1.252114604 0.540998808  
## 1755 -2.73716 0.1597100 -0.757581 0.214426000 -0.303155378 -2.266765003  
## 16961 -2.46277 0.1897360 -1.219790 0.273506000 0.811231864 -1.367218227  
## 1616 -2.49306 0.1989030 -2.046630 0.189875000 -1.259378794 1.046089176  
## 12221 -2.29862 0.2441410 -0.706881 0.001040700 0.762987289 0.345531150  
## 15873 -2.69980 0.2455930 -1.453880 0.271365000 -0.803453540 1.169703621  
## 1387 -2.31298 0.2241580 -1.040070 0.104266000 -0.379383917 1.622790746  
## 18883 -2.48564 0.2348300 -1.218640 0.021007100 -1.366494285 -0.277961194  
## 12152 -2.60267 0.1759000 -1.258920 0.010163900 3.519299065 -1.162967695  
## 1874 -2.30463 0.2447790 -0.957766 0.129564000 -1.213148497 0.875151132  
## 11321 -2.56314 0.2241780 -1.002240 0.038032600 0.194357317 0.244195278  
## 17601 -2.55465 0.2032450 -1.080330 0.007865450 0.094365637 -0.025469417  
## 16001 -2.53314 0.1878430 -1.041170 0.095584100 -0.878491419 -0.077051330  
## 15491 -2.61521 0.1422880 -1.793440 0.028257400 0.260435859 -2.050564466  
## 15013 -2.47917 0.2342920 -0.922222 0.268203000 0.657707063 0.061337130  
## 15821 -2.35096 0.2016750 -1.581820 0.132238000 0.765796530 -0.494998449  
## 15921 -2.28917 0.1799840 -1.330450 0.048681800 -2.235360626 0.479269669  
## 1852 -2.40673 0.2085140 -1.298470 0.015120300 -0.902826117 0.751390595  
## 10712 -2.38357 0.1722860 -1.238350 0.040024400 -0.166383835 0.172589245  
## 1394 -2.37273 0.1796020 -1.083980 0.087185500 -0.065499417 -0.060847194  
## 13771 -2.20821 0.1881340 -1.183620 0.182519000 -0.505638021 0.636966353  
## 11862 -2.41525 0.1492530 -0.834007 0.312228000 0.520286715 -1.463983320  
## 1966 -2.07809 0.1882600 -1.394100 0.099571600 0.035701483 -0.849069592  
## 11292 -2.18864 0.1597110 -1.537110 0.094656600 -0.910689564 0.508940075  
## 10831 -3.16043 0.2339010 -1.476150 0.170323000 0.914343124 0.632460951  
## 17903 -2.48192 0.2305350 -0.978394 0.027430600 -0.336709121 1.166244744  
## 1644 -2.40491 0.2828200 -1.390580 0.277275000 0.094762857 0.390579179  
## 1565 -2.57352 0.1260470 -1.160660 0.116141000 2.127220716 -0.136916556  
## 13241 -2.33673 0.2228740 -1.327240 0.178999000 0.196584478 -1.093804247  
## 16871 -2.58865 0.1974760 -0.994211 0.179774000 -0.967604740 -1.669799789  
## 1229 -2.36540 0.2142330 -1.477760 0.096631100 -0.284897245 1.131946555  
## 10372 -2.51429 0.1799650 -0.761858 0.164488000 0.164445997 0.569647807  
## 16003 -2.48131 0.1230270 -1.558900 0.007786990 -1.111870811 -0.629400068  
## 11721 -2.33397 0.2259040 -0.892438 0.001468270 -1.929364819 -0.917405418  
## 14731 -1.99460 0.1864290 -1.245440 0.145641000 -0.470260857 0.995658022  
## 14522 -2.35137 0.1524680 -1.330330 0.162020000 0.534387834 -0.889488176  
## 1525 -2.46434 0.1320820 -1.571680 0.266630000 -0.921049943 1.325976748  
## 16992 -2.57681 0.2009260 -1.326740 0.118381000 0.684374353 0.340950621  
## 16511 -2.69094 0.1897340 -1.168690 0.149909000 1.522078770 1.262076498  
## 14453 -2.34086 0.2649400 -1.221640 0.042497800 0.194613305 2.189657376  
## 11523 -2.69688 0.2056190 -0.874307 0.179480000 1.697224718 1.866736427  
## 13622 -2.81078 0.1909380 -1.606930 0.058103100 0.278068186 0.735091166  
## 15022 -2.22775 0.2134100 -1.324540 0.239999000 -0.571010463 -0.786304399  
## 1807 -2.36812 0.1696650 -1.788130 0.045823600 -0.157879440 -0.876974257  
## 13841 -2.11170 0.2028680 -1.123240 0.145967000 2.488339353 -1.220868128  
## 13602 -2.61921 0.2158160 -1.457980 0.225712000 1.071369094 1.616651857  
## 12461 -2.40362 0.1872570 -1.567720 0.023843000 0.342219077 0.498043756  
## 1135 -2.85813 0.2876250 -1.328960 0.021253400 0.221327427 1.602559271  
## 19683 -2.42895 0.1159610 -1.103940 0.174346000 0.068362742 0.635575578  
## 1320 -2.41499 0.2364620 -1.207360 0.014168100 0.307421602 -1.068431798  
## 12102 -2.25358 0.1706820 -0.979677 0.309230000 1.508633864 -1.707926964  
## 1567 -2.46527 0.2497050 -1.010630 0.154422000 0.615842239 -1.021218419  
## 19002 -2.24075 0.1400850 -1.528250 0.239416000 0.089812018 -0.227106122  
## 10362 -2.41991 0.1567680 -0.928403 0.300910000 -2.398452730 1.242464352  
## 15102 -2.20069 0.2118960 -1.044030 0.039451400 0.047807291 -0.578908578  
## 11942 -2.50907 0.1657300 -1.189890 0.115271000 0.336143355 1.032077158  
## 12732 -2.19947 0.2114250 -1.253010 0.035217200 0.434845736 -0.191477506  
## 1255 -2.39132 0.1608470 -1.245500 0.066009200 0.211023629 -0.424186098  
## 12453 -2.59509 0.1547980 -1.299020 0.000405925 0.110793489 -2.391613110  
## 13211 -2.50536 0.1992630 -1.489620 0.115853000 -1.993322911 1.646065343  
## 10972 -2.58263 0.1165580 -1.196730 0.151024000 1.505015705 -0.767060351  
## 16851 -2.64225 0.4001730 -0.990932 0.049278600 0.354157119 0.756270977  
## 1031 -2.77477 0.1867380 -1.312000 0.314459000 0.434837780 1.492709667  
## 10612 -2.80936 0.2356490 -1.305670 0.022095100 0.768243915 0.655936058  
## 16583 -2.31601 0.1819290 -1.181950 0.218057000 0.409688369 0.494717169  
## 14233 -1.95139 0.2065900 -1.703160 0.062042000 -1.976651601 -0.253101875  
## 10792 -2.60180 0.2145630 -1.003060 0.157798000 -1.087579363 -0.662941615  
## 1698 -2.34905 0.2717080 -1.385170 0.070941200 1.064863970 0.562125564  
## 1626 -2.92354 0.2073190 -0.690894 0.197559000 2.120202537 -0.779297892  
## 14002 -2.54335 0.1242340 -1.105440 0.009030700 -0.505837533 0.858962271  
## 1228 -2.24837 0.2537870 -1.476540 0.065627900 0.406987081 0.934961145  
## 1155 -2.28256 0.2509890 -1.706590 0.012908900 2.040025754 -0.791952504  
## 19331 -2.64976 0.3526000 -0.757965 0.209861000 -0.193935372 1.359486952  
## 1281 -2.76050 0.2090630 -1.337770 0.152799000 1.370222895 0.019649377  
## 17963 -2.53452 0.2009760 -1.082850 0.074052700 -0.515357348 -1.185940292  
## 17163 -3.10240 0.2293590 -1.300380 0.202736000 -0.718232806 0.343632123  
## 15473 -2.50039 0.2297830 -1.285410 0.274989000 -0.177242285 0.585487847  
## 16942 -2.40721 0.2265330 -1.318880 0.094239200 -0.401576463 -1.430200670  
## 13951 -2.59756 0.1340700 -1.209130 0.246509000 0.829883568 1.019029395  
## 17981 -2.40594 0.2319840 -1.252990 0.138020000 -0.785806614 0.320298932  
## 17043 -2.14834 0.1989730 -1.322350 0.154075000 -0.938501217 -0.501576176  
## 19892 -2.31831 0.1676150 -1.540090 0.182249000 0.417806768 -0.141376028  
## 16761 -2.51601 0.2601310 -1.175690 0.080375400 2.098097844 -1.430818217  
## 14713 -2.22716 0.3577880 -0.997161 0.267315000 -0.448862342 0.587857654  
## 12702 -2.28792 0.2376540 -1.437580 0.149641000 -1.095288092 0.868963574  
## 10551 -2.35730 0.4425240 -1.397530 0.055647700 -0.528972750 0.456654992  
## 10322 -2.20720 0.1739310 -1.334410 0.214915000 -0.484793938 -1.355959524  
## 18782 -2.93458 0.1399770 -1.234240 0.347723000 0.127757042 0.643807790  
## 10061 -2.51340 0.2468520 -1.164160 0.206347000 -0.074673107 0.070101188  
## 1083 -2.43802 0.1344960 -1.308430 0.042682700 -0.359162136 0.010391561  
## 14773 -2.50380 0.2149330 -1.382700 0.040187700 -2.048480559 0.721199978  
## 13932 -2.36046 0.2037610 -1.499560 0.332726000 -0.751180021 0.093814902  
## 14413 -2.13457 0.1785980 -1.470840 0.085842800 -1.581383000 -0.564257032  
## 15182 -2.85752 0.1460580 -1.441500 0.000644230 -0.289776058 0.007023682  
## 1371 -2.76031 0.2963820 -1.269930 0.276776000 -0.691954528 -0.561537486  
## 11582 -2.21017 0.1476000 -1.392470 0.043026600 -0.171987946 1.755959697  
## 11852 -2.37051 0.2427800 -0.844069 0.392518000 -0.222564010 0.170426604  
## 11001 -2.53218 0.1747070 -1.339830 0.001559260 0.919241279 1.065195204  
## 1943 -2.33617 0.1906190 -1.417810 0.114529000 -1.110065480 1.504156131  
## 15613 -3.04282 0.2528050 -1.197360 0.106698000 -0.660888997 0.659959697  
## 1793 -2.31278 0.2724480 -1.145820 0.166364000 -0.873774912 -0.838201076  
## 11502 -2.20824 0.2476800 -1.706390 0.066294500 -0.132809701 0.394689060  
## 16553 -2.15016 0.2254690 -1.394890 0.148632000 0.305489104 -0.379625994  
## 11162 -2.82267 0.3850410 -1.230960 0.082858500 2.163842485 2.066779896  
## 13412 -2.27822 0.1926330 -1.339660 0.113112000 0.880378505 -1.029749769  
## 14722 -2.71809 0.1551660 -1.484170 0.103863000 0.505116253 -0.123095504  
## 12323 -2.28551 0.1412030 -1.122580 0.249512000 0.110590205 0.173957707  
## 1722 -2.70577 0.2166930 -1.536540 0.115085000 -0.292858583 0.553190600  
## 1256 -2.64000 0.2900880 -1.110870 0.069901300 0.772319992 -0.759068678  
## 13092 -2.35258 0.2081970 -0.891715 0.093318500 2.128794229 0.035829594  
## 14762 -2.62003 0.1879280 -1.622810 0.152014000 0.547967970 -0.003495063  
## 18661 -2.26181 0.2046780 -1.143220 0.190714000 1.532129229 -0.220190416  
## 1032 -2.85814 0.1333870 -1.395130 0.077248900 1.666430854 -1.557680246  
## 12772 -2.31467 0.1955830 -1.312430 0.043286400 -0.478000049 -0.238045956  
## 1346 -2.49382 0.1819620 -0.907279 0.201956000 -0.368681625 0.424982482  
## 14222 -2.50812 0.2342960 -0.781122 0.281474000 -0.516166127 -0.446779715  
## 1432 -2.47556 0.1779880 -1.877770 0.023859300 0.396937689 -1.401854780  
## 1420 -2.77163 0.2028120 -1.319920 0.081282400 -0.314871760 -0.547414257  
## 15641 -2.89992 0.1890900 -0.977808 0.382889000 -0.684689801 0.247941769  
## 1591 -2.58881 0.3391210 -1.111740 0.092118700 1.632247789 -1.351702539  
## 10181 -2.64255 0.1454860 -0.789090 0.201865000 -0.684533761 1.285374670  
## 15213 -2.54255 0.2295360 -1.387110 0.033866100 -0.313076970 2.096632994  
## 18383 -2.48439 0.1518220 -1.125890 0.298323000 -0.640032652 -1.075716671  
## 17121 -2.43973 0.2029480 -1.646500 0.024709600 0.635304358 0.643818555  
## 11961 -2.26462 0.1342380 -1.217030 0.290942000 0.024848366 0.811107086  
## 1550 -2.71503 0.0866708 -1.274440 0.248071000 0.227622212 -0.373369162  
## 1021 -2.41049 0.2205590 -1.259630 0.227042000 -0.246068246 -0.117755702  
## 13783 -2.69021 0.2032320 -0.976812 0.122012000 0.024223829 0.939509301  
## 1472 -2.31697 0.1253420 -1.224330 0.004328210 -1.244582466 -0.223544486  
## 17673 -2.49725 0.2248920 -0.953237 0.021532700 -1.205974540 -1.188865752  
## 14023 -2.98113 0.2638230 -0.908168 0.065088700 0.273093950 0.599961163  
## 15453 -2.90986 0.2309110 -1.239930 0.119861000 1.729414034 -0.863306018  
## 16151 -2.57108 0.1473410 -1.136320 0.048124600 -1.064950869 -0.669136112  
## 14392 -2.73386 0.2153710 -1.123620 0.158941000 -0.113736688 0.758080913  
## 1606 -2.71698 0.3049020 -1.202840 0.220314000 -0.643751471 0.977212323  
## 14853 -2.13772 0.2354160 -1.212720 0.221584000 0.940648418 1.486960427  
## 16371 -2.54830 0.1443340 -1.651420 0.048851200 1.918810176 0.809182023  
## 14261 -2.53918 0.3321380 -1.416020 0.025390200 -1.006045550 1.049591298  
## 13171 -2.52621 0.1663700 -0.893134 0.271118000 0.547082630 -1.378317315  
## 10481 -2.41713 0.1124750 -1.124400 0.086692100 0.033847059 -1.166379732  
## 14142 -2.71942 0.1545870 -1.102770 0.042130400 -0.315828929 0.450531663  
## 15372 -2.47209 0.2391330 -1.101620 0.075715300 0.288544593 0.553986739  
## 13811 -2.18047 0.2045900 -1.370160 0.324137000 -0.344392202 0.171533785  
## 1300 -2.67440 0.2350430 -1.102010 0.137870000 0.116631435 -0.806881470  
## 1855 -2.40777 0.2950850 -1.357810 0.011541000 0.542965968 0.978852878  
## 18943 -2.12616 0.1371430 -1.402240 0.004606600 1.143947544 1.171035498  
## 18831 -2.59519 0.1793470 -1.102120 0.057315400 -1.284548845 -0.254750999  
## 1629 -2.83572 0.1805340 -1.441630 0.176029000 0.919811790 -0.261779474  
## 11953 -2.61364 0.2311260 -1.086680 0.137345000 -0.846090161 1.357008537  
## 13562 -2.68224 0.2243340 -1.458800 0.180173000 -1.419304273 1.048195428  
## 18421 -1.92579 0.1908020 -1.629080 0.027536500 -2.665697589 2.228898486  
## 20003 -2.34036 0.1948530 -1.190090 0.084370600 0.835106141 -1.470518812  
## 11401 -2.25904 0.2516890 -1.238840 0.085555000 2.377446549 -1.031250041  
## 1048 -2.56151 0.1372240 -1.522600 0.401672000 0.013969708 0.164813527  
## 12143 -2.50822 0.1877110 -0.970648 0.126705000 -1.633423995 -1.243838156  
## 1075 -2.46098 0.2040470 -1.388510 0.088963900 0.488932473 0.116308174  
## 18021 -2.28623 0.2211310 -1.524870 0.055665500 -1.019179409 1.197547370  
## 17623 -2.79484 0.2085800 -1.296040 0.250570000 1.090553483 -0.698130988  
## 1499 -2.62833 0.1774690 -1.430950 0.442119000 -1.139469805 0.293604948  
## 18673 -2.47243 0.1904210 -1.219440 0.183877000 -0.015725635 0.437297744  
## 1093 -3.22546 0.2556450 -1.293500 0.107878000 0.297491282 0.627762573  
## 13553 -2.20810 0.1711950 -1.279340 0.092635200 3.200590040 -1.160543514  
## 12592 -2.04985 0.2424870 -1.122830 0.219212000 0.089244244 -0.020788418  
## 12912 -2.11565 0.2306120 -1.168590 0.077838100 0.570972338 1.044724089  
## 1785 -2.83093 0.2811680 -1.193550 0.205387000 0.528685630 1.311594585  
## 14041 -2.74332 0.2207640 -0.899848 0.112913000 -0.440904798 -0.082439028  
## 1624 -3.04566 0.2764900 -1.327870 0.116234000 -0.672793368 0.919224762  
## 1841 -2.26703 0.2587430 -1.489040 0.164183000 2.154313365 -0.243269921  
## 19553 -2.19122 0.2239030 -1.503950 0.036649500 0.593852716 0.351094377  
## 12722 -2.03384 0.2361680 -1.159480 0.222747000 -0.384063768 -1.340384172  
## 15122 -2.41409 0.2228960 -0.964277 0.097466900 0.719783276 0.425846094  
## 1348 -2.62948 0.2830370 -1.066280 0.217605000 1.708173402 -0.814353733  
## 15183 -2.83780 0.2155680 -1.139210 0.209466000 1.075215667 0.693374081  
## 14432 -2.58703 0.1585020 -1.086690 0.105126000 0.777420367 0.705544556  
## 1468 -2.63761 0.2137270 -1.200060 0.012087000 0.052200047 0.278350539  
## 17953 -2.69536 0.2201650 -0.944860 0.117315000 -1.083698485 -1.202692142  
## 1557 -2.28829 0.2691060 -1.096310 0.117185000 -0.008856207 0.642761817  
## 19642 -2.45008 0.1746040 -0.757825 0.047770600 2.030643112 -0.605973945  
## 16191 -2.07468 0.1367260 -1.307650 0.200690000 -1.113328933 -1.377257613  
## 17011 -2.54155 0.1779580 -1.088670 0.200049000 1.138271925 -0.925992687  
## 1820 -2.67781 0.2744160 -1.130930 0.272739000 -0.614892642 0.567243857  
## 10851 -2.61594 0.1918340 -1.604430 0.081496100 -2.193536615 2.595481260  
## 13483 -2.97149 0.2244890 -1.520540 0.118417000 -1.215686372 -0.239984547  
## 13183 -2.69848 0.2226620 -1.059710 0.105705000 -0.045326362 0.687536401  
## 18462 -2.44366 0.1458040 -1.460300 0.028477800 1.598812624 -0.151788986  
## 15542 -2.37954 0.2782630 -1.224820 0.084755500 0.927542200 0.227544260  
## 18752 -2.64053 0.1648670 -1.638830 0.101685000 0.829284552 -0.668648038  
## 12081 -2.39692 0.2183460 -1.381470 0.028250100 1.024603876 0.030829067  
## 1014 -2.75811 0.1743640 -1.221350 0.103274000 -0.476238068 0.028429558  
## 15612 -2.76002 0.1996990 -1.089480 0.134250000 1.638868393 -0.365455074  
## 14833 -1.95189 0.1598310 -1.265820 0.036476900 -0.632050958 -2.208012195  
## 1639 -2.73364 0.2009320 -1.162480 0.169149000 -1.379618576 0.297039427  
## 10353 -2.15850 0.1615210 -1.368610 0.192932000 -0.257455748 2.129700659  
## 15011 -2.55299 0.2621500 -0.913079 0.061140100 1.679972711 1.325041143  
## 12253 -2.36845 0.1232820 -1.094110 0.230603000 -2.545858038 -0.116171447  
## 10793 -1.94338 0.2035550 -1.535410 0.026017300 0.012078784 -1.470146221  
## 10103 -2.87468 0.1556830 -1.345700 0.022665200 1.960925111 -0.379271835  
## 1903 -2.71520 0.2042490 -1.459870 0.450800000 -0.385905481 -1.465005931  
## 1679 -2.30976 0.2779830 -1.179630 0.181344000 0.910757548 1.075148263  
## 1795 -2.62165 0.1818670 -1.292130 0.177647000 -1.448130794 -1.226124877  
## 1489 -2.65237 0.1656490 -1.472160 0.161936000 -1.121614596 -3.056328234  
## 1881 -2.69172 0.2042330 -1.243580 0.238797000 -0.973617906 1.450657775  
## 15061 -2.69719 0.1883980 -1.343580 0.262119000 -0.067186922 0.717976859

dat$ln\_k\_i <- dat$`M\_ln\_k(1)` + sqrt(dat$`V\_ln\_k(1)`)\*dat$z\_ln\_k  
dat$ln\_Vd\_i <- dat$`M\_ln\_Vd(1)`+ dat$`SD\_ln\_Vd(1)`\*dat$z\_ln\_Vd  
linmod <- lm(ln\_Vd\_i ~ ln\_k\_i,data=dat)  
ggplot(dat) + geom\_point(aes(ln\_k\_i,ln\_Vd\_i)) +   
 labs(subtitle=paste("Adj R2 =",signif(summary(linmod)$adj.r.squared,2)))

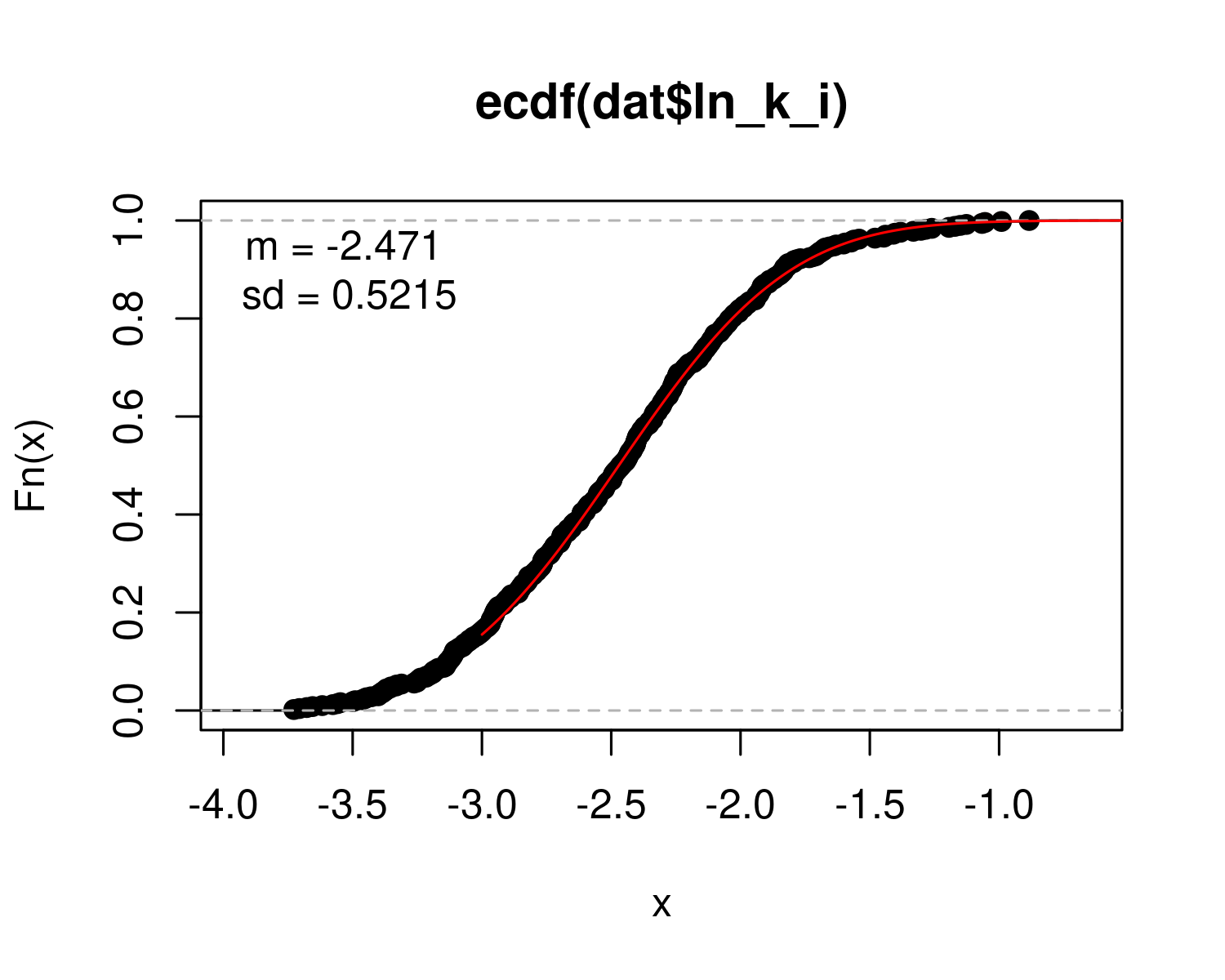


## Check normality

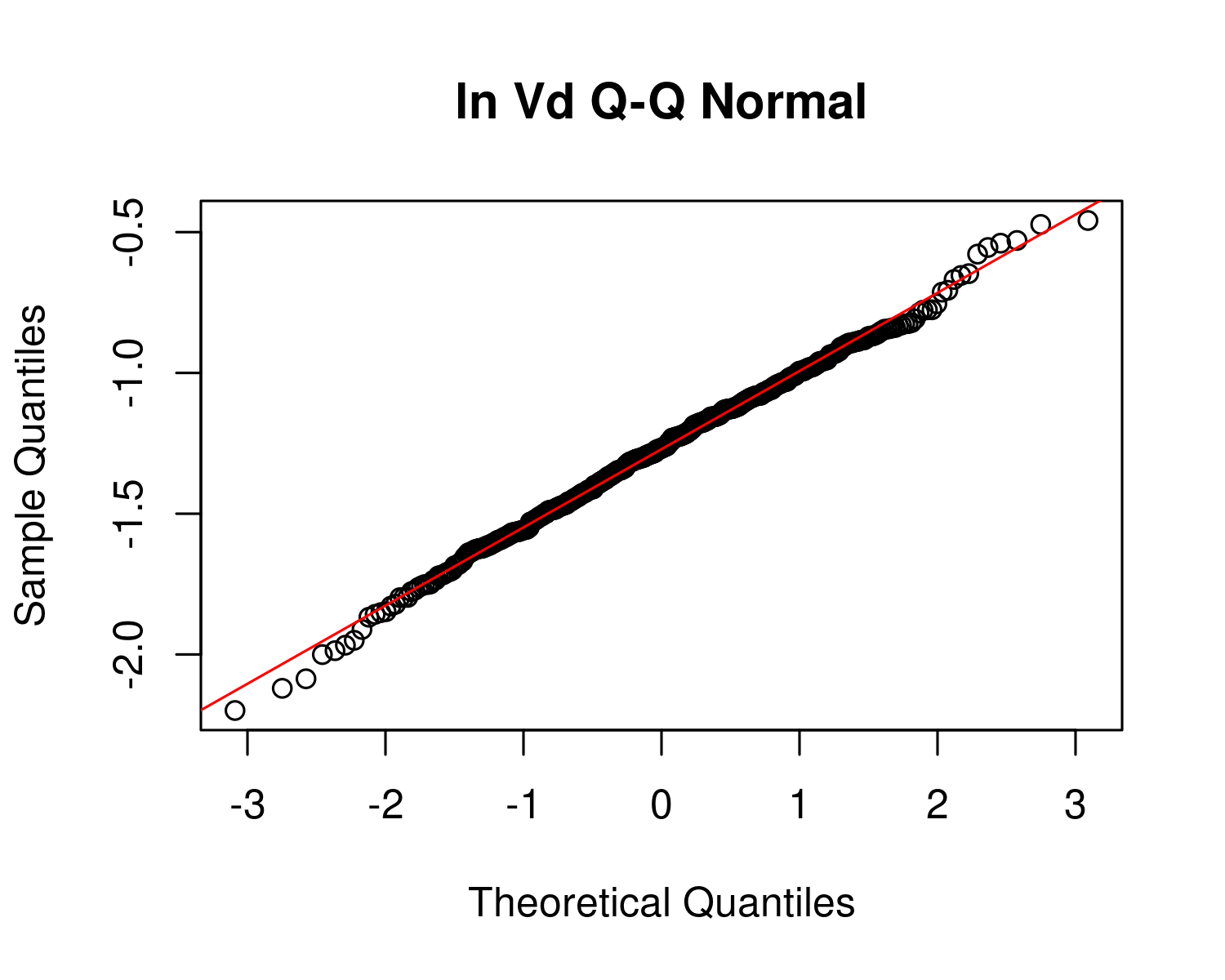
qqnorm(dat$ln\_k\_i,main="ln k Q-Q Normal")  
qqline(dat$ln\_k\_i,col="red")



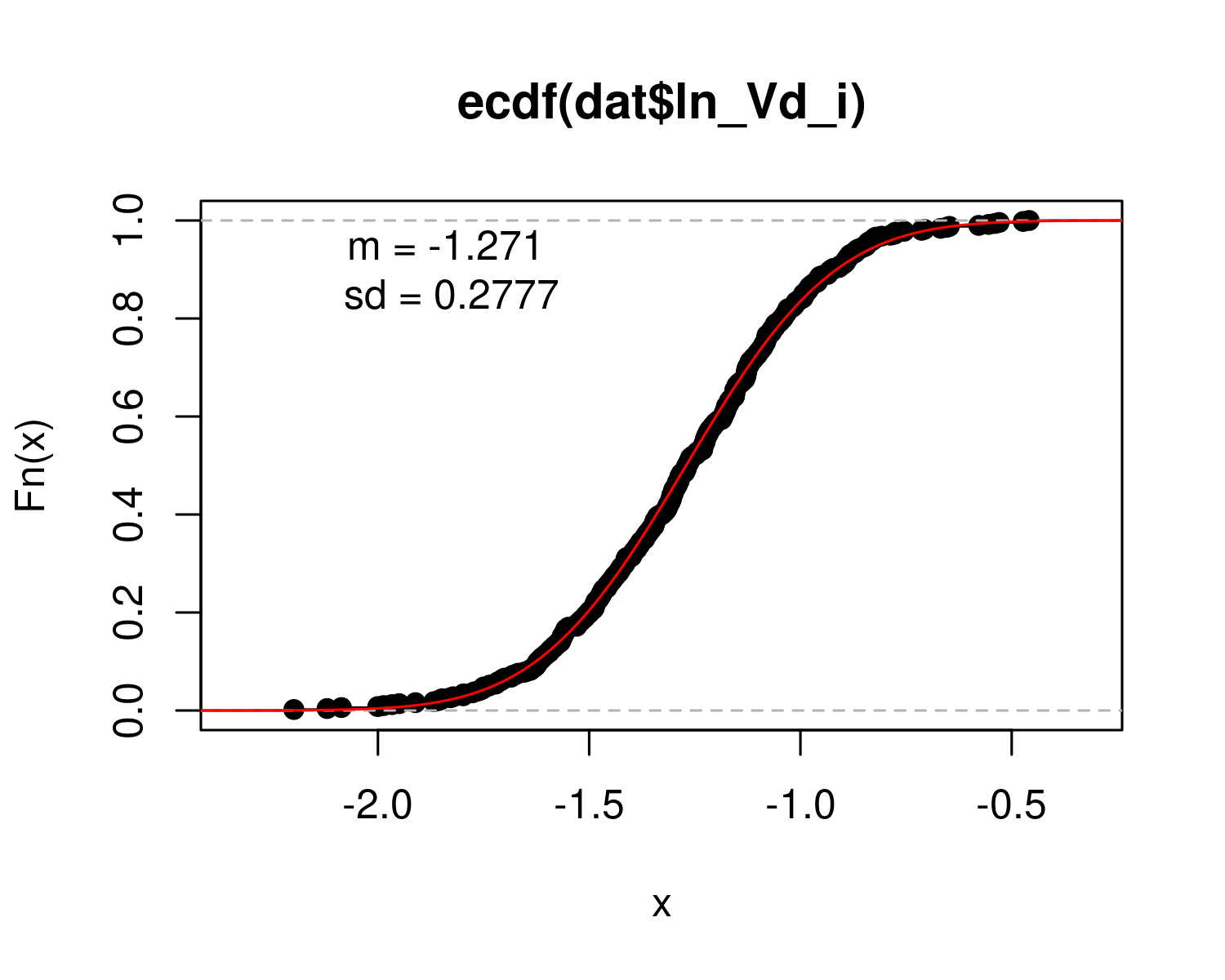
plot(ecdf(dat$ln\_k\_i))  
x <- seq(-3,1,0.01)  
m\_ln\_k\_i <- mean(dat$ln\_k\_i)  
sd\_ln\_k\_i <- sd(dat$ln\_k\_i)  
lines(x,pnorm(x,mean=m\_ln\_k\_i,sd=sd\_ln\_k\_i),col="red")  
text(m\_ln\_k\_i-2\*sd\_ln\_k\_i,0.9,paste("m =",signif(m\_ln\_k\_i,4),"\nsd =",signif(sd\_ln\_k\_i,4)))



qqnorm(dat$ln\_Vd\_i,main="ln Vd Q-Q Normal")  
qqline(dat$ln\_Vd\_i,col="red")



plot(ecdf(dat$ln\_Vd\_i))  
x <- seq(-3,1,0.01)  
m\_ln\_Vd\_i <- mean(dat$ln\_Vd\_i)  
sd\_ln\_Vd\_i <- sd(dat$ln\_Vd\_i)  
  
lines(x,pnorm(x,mean=m\_ln\_Vd\_i,sd=sd\_ln\_Vd\_i),col="red")  
text(m\_ln\_Vd\_i-2\*sd\_ln\_Vd\_i,0.9,paste("m =",signif(m\_ln\_Vd\_i,4),"\nsd =",signif(sd\_ln\_Vd\_i,4)))



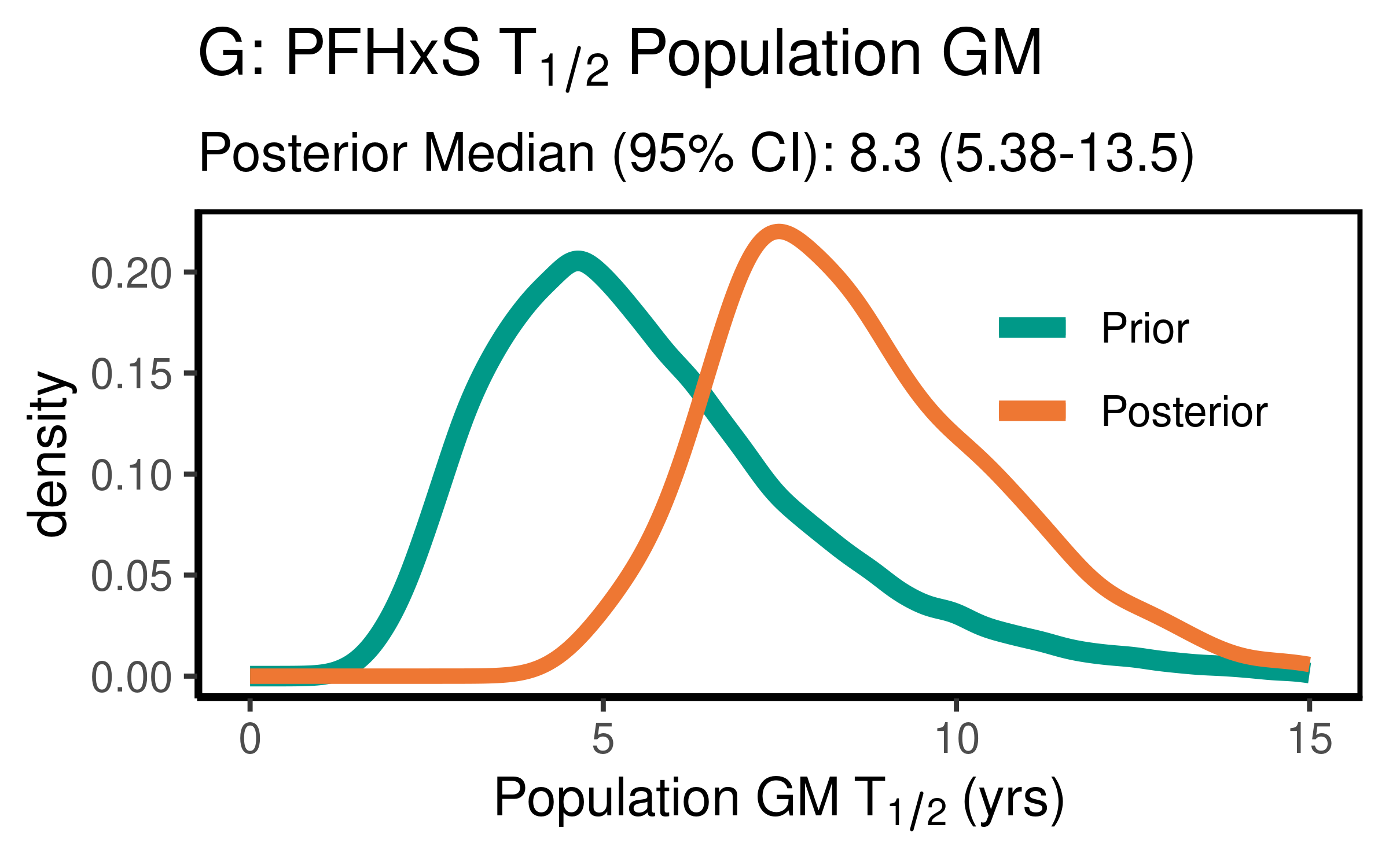
## Calculate table values for individual-level

hl\_i <- log(2)/ exp(dat$ln\_k\_i) # individual half-life   
med\_hl\_i <- paste(signif (median (hl\_i), 3)) # median of individual half-life  
ci\_med\_hl\_i <- paste(signif (quantile(hl\_i, prob=c(0.025,0.975)), 3),collapse="-") # 95ci med individual halflife  
ci98\_med\_hl\_i <- paste(signif (quantile(hl\_i, prob=c(0.01,0.99)), 3),collapse="-") # 98ci med individual halflife  
gm\_hl\_i <- paste(signif (exp(mean(log(hl\_i))), 3)) # gm (which should be really close)  
gsd\_hl\_i <- paste(signif (exp(sd(log(hl\_i))), 3)) # gsd individual  
  
med\_Vd\_i <- paste(signif (median(exp(dat$ln\_Vd\_i)), 3)) # median individual Vd  
ci\_med\_Vd\_i <-paste(signif (quantile(exp(dat$ln\_Vd\_i), prob=c(0.025,0.975)), 3),collapse="-") # 95ci med individual Vd  
ci98\_med\_Vd\_i <-paste(signif (quantile(exp(dat$ln\_Vd\_i), prob=c(0.01,0.99)), 3),collapse="-") # 98ci med individual Vd  
gm\_vd\_i <- paste(signif (exp(mean(dat$ln\_Vd\_i)), 3)) # gm (which should be really close)  
gsd\_vd\_i<- paste(signif (exp(sd(dat$ln\_Vd\_i)), 3)) # gsd indiv  
  
med\_CL\_i <- paste(signif (median(exp(dat$ln\_Vd\_i+dat$ln\_k\_i)), 3)) # median individual CL  
ci\_med\_CL\_i <-paste(signif (quantile(exp(dat$ln\_Vd\_i+dat$ln\_k\_i), prob=c(0.025,0.975)), 3),collapse="-") # 95ci med individual CL  
ci98\_med\_CL\_i <-paste(signif (quantile(exp(dat$ln\_Vd\_i+dat$ln\_k\_i), prob=c(0.01,0.99)), 3),collapse="-") # 98ci med individual CL  
gm\_CL\_i <- paste(signif (exp(mean(dat$ln\_Vd\_i+dat$ln\_k\_i)), 3)) # gm (which should be really close)  
gsd\_CL\_i<- paste(signif (exp(sd(dat$ln\_Vd\_i+dat$ln\_k\_i)), 3)) # gsd indiv

PFHxS\_priors <- data.frame(  
 halflife\_GM= log(2)/rlnorm(50000,  
 meanlog=-2.03422,sdlog=0.4055))  
M\_k <- exp(as.numeric(dat$`M\_ln\_k(1)`))  
PFHxS\_halflife\_GM <- log(2)/M\_k  
  
PFHxS\_hlgm\_pr\_med <- signif(median(PFHxS\_priors$halflife\_GM,3))  
PFHxS\_hlgm\_pr\_med\_95ci <-paste(signif(quantile(PFHxS\_priors$halflife\_GM,  
 prob=c(0.025,0.975)),  
 3),  
 collapse="-")  
  
PFHxS\_hl\_median\_gm <- signif(median(PFHxS\_halflife\_GM),3)  
PFHxS\_hl\_median\_gm\_95ci <- paste(signif(quantile(PFHxS\_halflife\_GM,  
 prob=c(0.025,0.975)),3),collapse="-")  
  
p<-ggplot()+  
 stat\_density(aes(halflife\_GM, color = "Prior"),data=PFHxS\_priors,geom="line",size=2)+  
 stat\_density(aes(PFHxS\_halflife\_GM, stat(density),color="Posterior"),geom="line",size=1.5 )+  
 xlim(0,15)+  
 labs(title = bquote("G: PFHxS"~T[1/2]~"Population GM") ,  
 subtitle=paste("Posterior Median (95% CI): ",  
 PFHxS\_hl\_median\_gm," (",  
 PFHxS\_hl\_median\_gm\_95ci,  
 ")",sep=""))+  
 xlab(bquote("Population GM"~T[1/2]~"(yrs)")) +  
 scale\_color\_manual(name=NULL,#  
 values=c(Prior="#009988", Posterior="#EE7733" )) +   
 theme\_classic() +   
 theme(legend.title = element\_blank(),legend.position=c(0.8,0.7),  
 panel.background = element\_rect(color="black",size=1),  
 legend.background = element\_rect(fill="transparent", color=NA))  
print(p)

## Warning: Removed 250 rows containing non-finite values (stat\_density).

## Warning: Removed 6 rows containing non-finite values (stat\_density).



ggsave(here ("output-plots",paste0( sa, "PFHxS\_hl\_gm.pdf")),p,dpi=600)

## Saving 4 x 2.5 in image

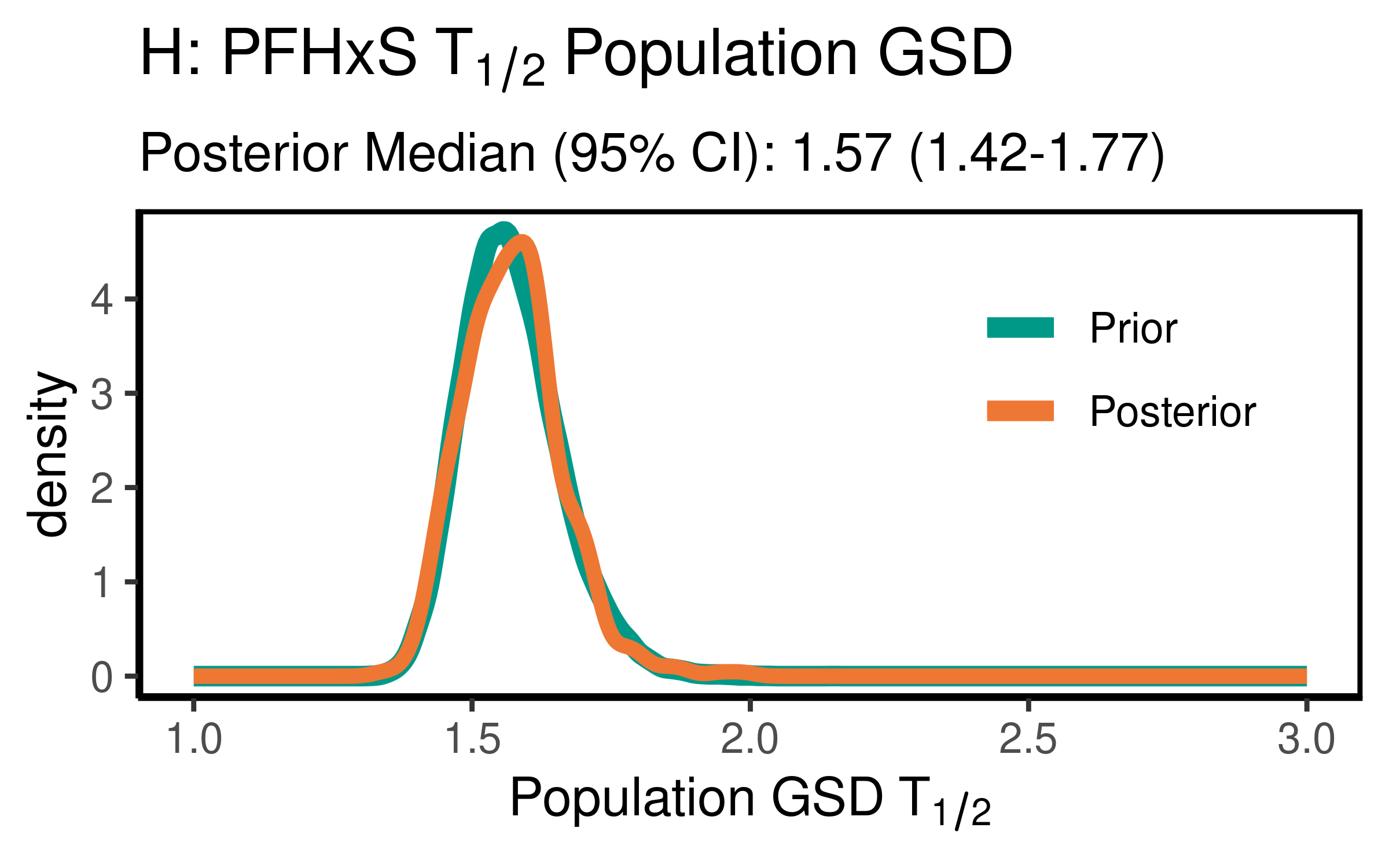
## Warning: Removed 250 rows containing non-finite values (stat\_density).  
  
## Warning: Removed 6 rows containing non-finite values (stat\_density).

ggsave(here ("output-plots",paste0( sa, "PFHxS\_hl\_gm.png")),p,dpi=600)

## Saving 4 x 2.5 in image

## Warning: Removed 250 rows containing non-finite values (stat\_density).  
  
## Warning: Removed 6 rows containing non-finite values (stat\_density).

PFHxS\_priors$halflife\_GSD = exp(sqrt(exp(rnorm(50000,m=log(0.2),sd=log(1.275)))))   
PFHxS\_halflife\_GSD <- exp(sqrt(dat$`V\_ln\_k(1)`))  
  
PFHxS\_hlgsd\_pr\_med <- signif(median(PFHxS\_priors$halflife\_GSD,3))  
PFHxS\_hlgsd\_pr\_med\_95ci <-paste(signif(quantile(PFHxS\_priors$halflife\_GSD,  
 prob=c(0.025,0.975)),  
 3),  
 collapse="-")  
PFHxS\_hl\_gsd\_med <- signif(median(PFHxS\_halflife\_GSD),3)  
PFHxS\_hl\_gsd\_med\_95ci <- paste(signif(quantile(PFHxS\_halflife\_GSD,  
 prob=c(0.025,0.975)),3),collapse="-")  
p<-ggplot()+  
 stat\_density(aes(halflife\_GSD, color = "Prior"),data=PFHxS\_priors,geom="line", size=2)+  
 stat\_density(aes(PFHxS\_halflife\_GSD,stat(density), color = "Posterior"),geom="line",size=1.5)+  
 xlim(1,3)+  
 labs(title = bquote("H: PFHxS"~T[1/2]~"Population GSD"),   
 subtitle=paste("Posterior Median (95% CI): ",  
 PFHxS\_hl\_gsd\_med," (",  
 PFHxS\_hl\_gsd\_med\_95ci,  
 ")",sep=""))+  
 xlab(bquote("Population GSD"~T[1/2]))+  
 scale\_color\_manual(name=NULL,#  
 values=c(Prior="#009988", Posterior="#EE7733" )) +   
 theme\_classic() +   
 theme(legend.title = element\_blank(),legend.position=c(0.8,0.7),  
 panel.background = element\_rect(color="black",size=1),  
 legend.background = element\_rect(fill="transparent", color=NA))  
print(p)

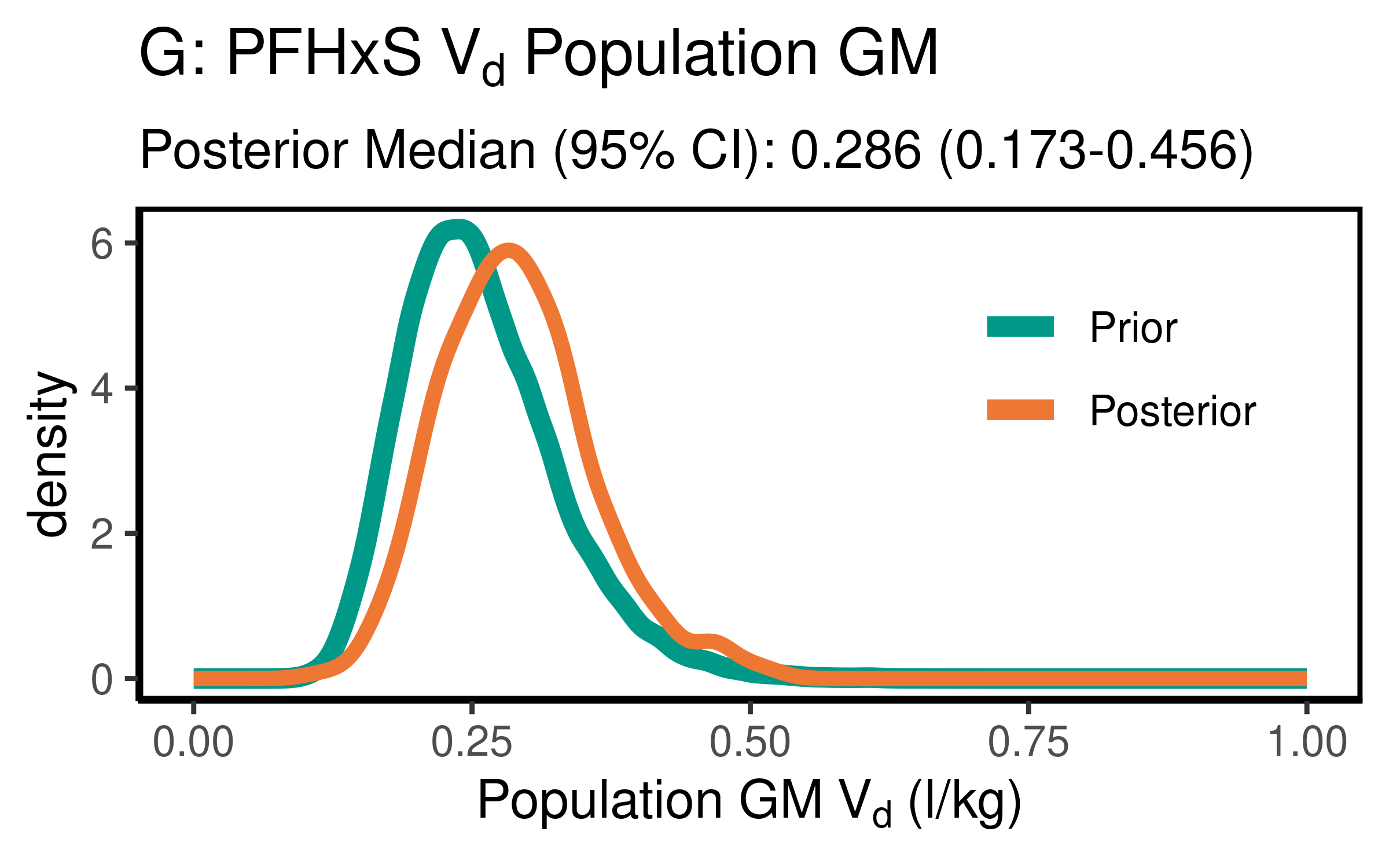


ggsave(here ("output-plots",paste0( sa, "PFHxS\_hl\_gsd.pdf")) ,p,dpi=600)  
ggsave(here ("output-plots",paste0( sa, "PFHxS\_hl\_gsd.png")) ,p,dpi=600)

### Volume of distribution

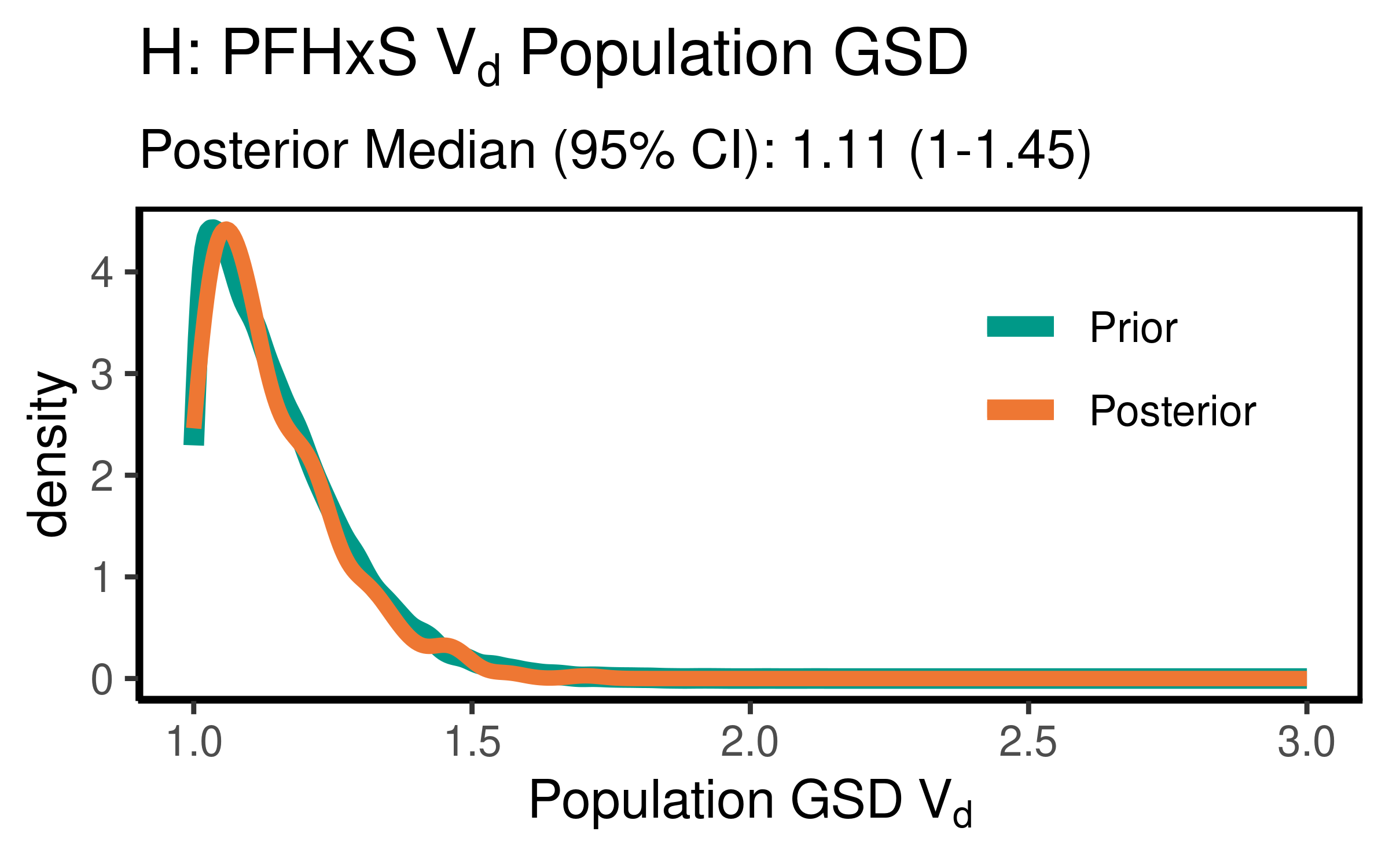
For PFHxS, the data were not particularly informative, but slightly increased the estimate of the median to 0.308(0.223-0.548) slightly compared to the case where Cbgd was considered to be entirely from non-drinking water sources. They were not informative as to the population GSD, with the posterior distributions essentially unchanged from the priors.

PFHxS\_priors$Vd\_GM <- rlnorm(50000,  
 meanlog=-1.38629,  
 sdlog=0.2624)  
PFHxS\_Vd\_GM <- exp(dat$`M\_ln\_Vd(1)`)  
  
   
  
PFHxS\_vd\_gm\_pr\_med <- signif(median(PFHxS\_priors$Vd\_GM,3))  
PFHxS\_vd\_gm\_pr\_med\_95ci <- paste(signif(quantile(PFHxS\_priors$Vd\_GM,  
 prob=c(0.025,0.975)), 3), collapse="-")  
PFHxS\_vd\_gm\_med <- signif(median(PFHxS\_Vd\_GM),3)  
PFHxS\_vd\_gm\_med\_95ci <- paste(signif(quantile(PFHxS\_Vd\_GM,  
 prob=c(0.025,0.975)),3),collapse="-")  
  
p<-ggplot()+  
 stat\_density(aes(Vd\_GM, color = "Prior"),data=PFHxS\_priors,geom="line",size=2)+  
 stat\_density(aes(PFHxS\_Vd\_GM,stat(density), color = "Posterior"),geom="line",size=1.5)+  
 xlim(0,1)+labs(title = bquote("G: PFHxS"~V[d]~"Population GM"),  
 subtitle=paste("Posterior Median (95% CI): ",  
 PFHxS\_vd\_gm\_med," (",   
 PFHxS\_vd\_gm\_med\_95ci,")",sep=""))+  
 xlab(bquote("Population GM"~V[d]~"(l/kg)"))+  
 scale\_color\_manual(name=NULL,#  
 values=c(Prior="#009988", Posterior="#EE7733" )) +   
 theme\_classic() +   
 theme(legend.title = element\_blank(),legend.position=c(0.8,0.7),  
 panel.background = element\_rect(color="black",size=1),  
 legend.background = element\_rect(fill="transparent", color=NA))  
print(p)



ggsave(here ("output-plots",paste0( sa, "PFHxS\_vd\_gm.pdf")) ,p,dpi=600)  
ggsave(here ("output-plots",paste0( sa, "PFHxS\_vd\_gm.png")) ,p,dpi=600)

PFHxS\_priors$Vd\_GSD = exp(abs(rnorm(50000,sd=0.17)))  
PFHxS\_Vd\_GSD <- exp(dat$`SD\_ln\_Vd(1)`)  
  
PFHxS\_vd\_gsd\_pr\_med <- signif(median(PFHxS\_priors$Vd\_GSD,3))  
PFHxS\_vd\_gsd\_pr\_med\_95ci <- paste(signif(quantile(PFHxS\_priors$Vd\_GSD,  
 prob=c(0.025,0.975)), 3), collapse="-")  
  
PFHxS\_vd\_gsd\_med <- signif(median(PFHxS\_Vd\_GSD),3)  
PFHxS\_vd\_gsd\_med\_95ci <- paste(signif(quantile(PFHxS\_Vd\_GSD,  
 prob=c(0.025,0.975)),3),collapse="-")  
  
p<-ggplot()+  
 stat\_density(aes(Vd\_GSD, color = "Prior"),data=PFHxS\_priors,geom="line",size=2)+  
 stat\_density(aes(PFHxS\_Vd\_GSD,stat(density), color = "Posterior"),geom="line",size=1.5)+  
 xlim(1,3)+  
 labs(title = bquote("H: PFHxS"~V[d]~"Population GSD "),  
 subtitle=paste("Posterior Median (95% CI): ",  
 PFHxS\_vd\_gsd\_med," (",  
 PFHxS\_vd\_gsd\_med\_95ci,  
 ")",sep=""))+  
 xlab(bquote("Population GSD"~V[d]))+  
 scale\_color\_manual(name=NULL,   
 values=c(Prior="#009988", Posterior="#EE7733" )) +   
 theme\_classic() +   
 theme(legend.title = element\_blank(),legend.position=c(0.8,0.7),  
 panel.background = element\_rect(color="black",size=1),  
 legend.background = element\_rect(fill="transparent", color=NA))  
print(p)

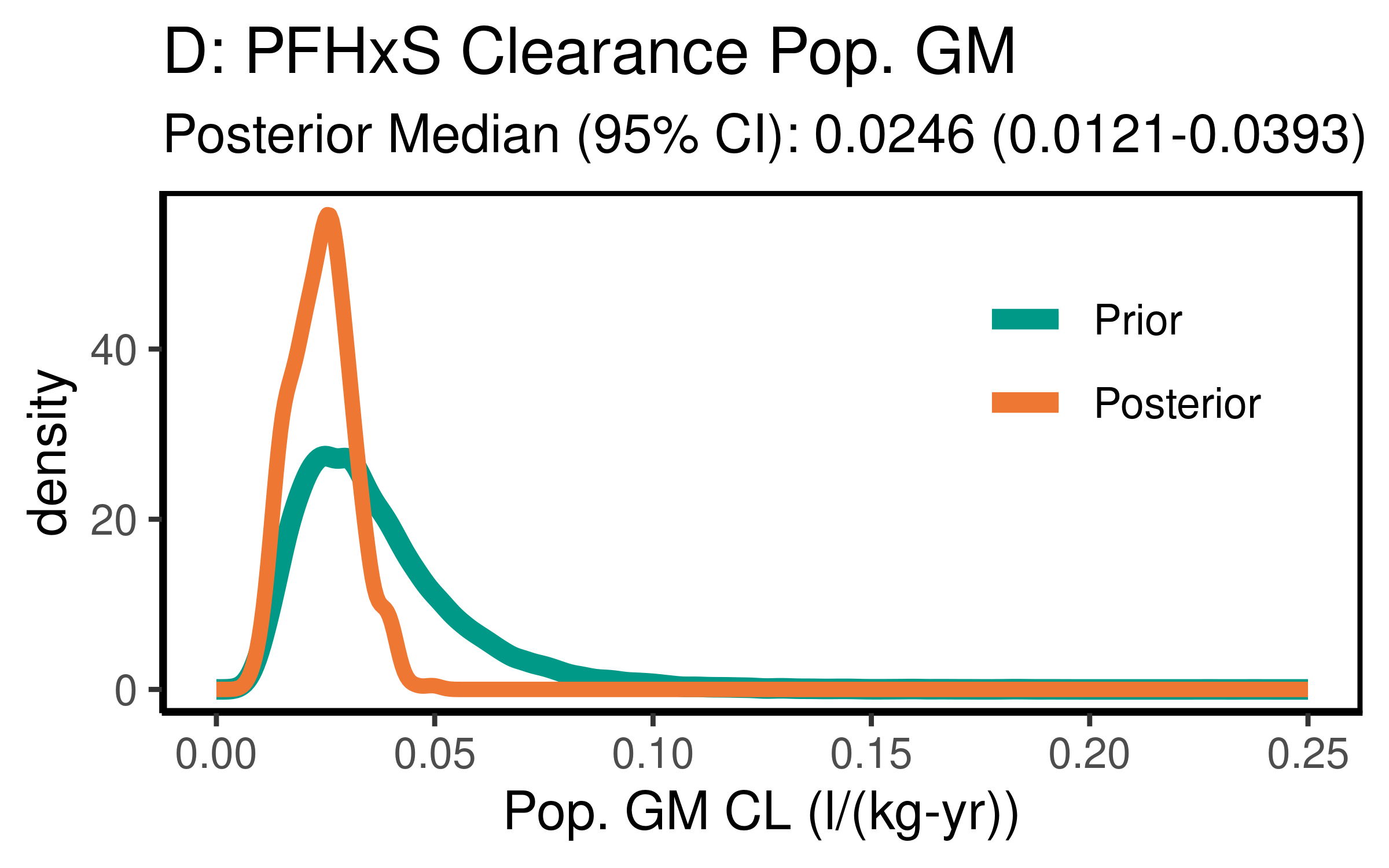


ggsave(here ("output-plots",paste0( sa,"PFHxS\_vd\_gsd.pdf")) ,p,dpi=600)  
ggsave(here ("output-plots",paste0( sa,"PFHxS\_vd\_gsd.png")) ,p,dpi=600)

### Clearance

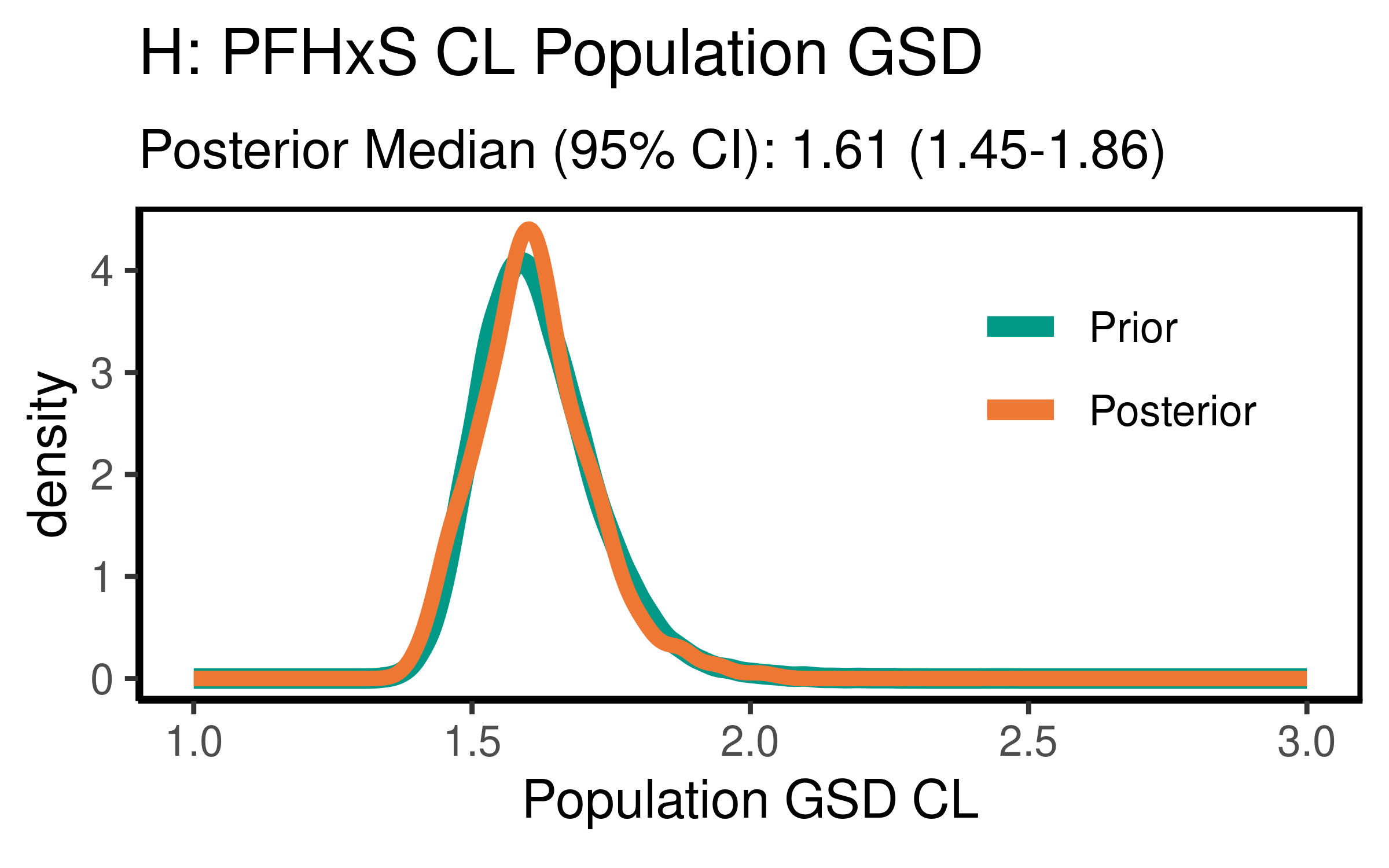
Cl is k \* Vd

PFHxS\_priors$CL\_GM <- PFHxS\_priors$Vd\_GM \* (log(2)/PFHxS\_priors$halflife\_GM)  
PFHxS\_CL\_GM <- exp(dat$`M\_ln\_Vd(1)` + dat$`M\_ln\_k(1)`)  
  
PFHxS\_cl\_gm\_pr\_med <- signif(median(PFHxS\_priors$CL\_GM,3))  
PFHxS\_cl\_gm\_pr\_med\_95ci <- paste(signif(quantile(PFHxS\_priors$CL\_GM,  
 prob=c(0.025,0.975)), 3), collapse="-")  
PFHxS\_cl\_gm\_med <- signif(median(PFHxS\_CL\_GM),3)  
PFHxS\_cl\_gm\_med\_95ci <- paste(signif(quantile(PFHxS\_CL\_GM,  
 prob=c(0.025,0.975)),3),collapse="-")  
  
p<-ggplot()+  
 stat\_density(aes(CL\_GM, color = "Prior"),data=PFHxS\_priors,geom="line",size=2)+  
 stat\_density(aes(PFHxS\_CL\_GM,stat(density), color = "Posterior"),geom="line",size=1.5)+  
 xlim(0,0.25)+labs(title = "D: PFHxS Clearance Pop. GM ",subtitle=paste("Posterior Median (95% CI): ",  
 PFHxS\_cl\_gm\_med," (",  
 PFHxS\_cl\_gm\_med\_95ci,  
 ")",sep=""))+  
 xlab("Pop. GM CL (l/(kg-yr))")+  
 scale\_color\_manual(name=NULL,#  
 values=c(Prior="#009988", Posterior="#EE7733" )) +   
 theme\_classic() +   
 theme(legend.title = element\_blank(),legend.position=c(0.8,0.7),  
 panel.background = element\_rect(color="black",size=1),  
 legend.background = element\_rect(fill="transparent", color=NA))  
print(p)



ggsave(here ("output-plots",paste0( sa, "PFHxS\_CL\_gm.pdf")), p,dpi=600)  
ggsave(here ("output-plots",paste0( sa, "PFHxS\_CL\_gm.png")), p,dpi=600)

PFHxS\_priors$CL\_GSD = exp(sqrt(log(PFHxS\_priors$Vd\_GSD)^2 +   
 log(PFHxS\_priors$halflife\_GSD)^2))  
PFHxS\_CL\_GSD <- exp(sqrt(log(PFHxS\_Vd\_GSD)^2 +   
 log(PFHxS\_halflife\_GSD)^2))  
  
PFHxS\_CL\_gsd\_pr\_med <- signif(median(PFHxS\_priors$CL\_GSD,3))  
PFHxS\_CL\_gsd\_pr\_med\_95ci <- paste(signif(quantile(PFHxS\_priors$CL\_GSD,  
 prob=c(0.025,0.975)), 3), collapse="-")  
  
PFHxS\_CL\_gsd\_med <- signif(median(PFHxS\_CL\_GSD),3)  
PFHxS\_CL\_gsd\_med\_95ci <- paste(signif(quantile(PFHxS\_CL\_GSD,  
 prob=c(0.025,0.975)),3),collapse="-")  
  
p<-ggplot()+  
 stat\_density(aes(CL\_GSD, color = "Prior"),data=PFHxS\_priors,geom="line",size=2)+  
 stat\_density(aes(PFHxS\_CL\_GSD,stat(density), color = "Posterior"),geom="line",size=1.5)+  
 xlim(1,3)+  
 labs(title = bquote("H: PFHxS"~CL~"Population GSD "),  
 subtitle=paste("Posterior Median (95% CI): ",  
 PFHxS\_CL\_gsd\_med," (",  
 PFHxS\_CL\_gsd\_med\_95ci,  
 ")",sep=""))+  
 xlab(bquote("Population GSD"~CL))+  
 scale\_color\_manual(name=NULL,   
 values=c(Prior="#009988", Posterior="#EE7733" )) +   
 theme\_classic() +   
 theme(legend.title = element\_blank(),legend.position=c(0.8,0.7),  
 panel.background = element\_rect(color="black",size=1),  
 legend.background = element\_rect(fill="transparent", color=NA))  
print(p)



ggsave(here ("output-plots",paste0( sa,"PFHxS\_CL\_gsd.pdf")) ,p,dpi=600)  
ggsave(here ("output-plots",paste0( sa,"PFHxS\_CL\_gsd.png")) ,p,dpi=600)

## Table significant digit values

PFHxS\_hlgm\_pr\_med <- paste(signif(PFHxS\_hlgm\_pr\_med, 3))  
PFHxS\_hl\_median\_gm<- paste(signif(PFHxS\_hl\_median\_gm, 3))  
PFHxS\_hlgsd\_pr\_med<- paste(signif(PFHxS\_hlgsd\_pr\_med, 3))  
PFHxS\_hl\_gsd\_med<- paste(signif(PFHxS\_hl\_gsd\_med, 3))  
PFHxS\_vd\_gm\_pr\_med<- paste(signif(PFHxS\_vd\_gm\_pr\_med, 3))  
PFHxS\_vd\_gm\_med<- paste(signif(PFHxS\_vd\_gm\_med, 3))  
PFHxS\_vd\_gsd\_pr\_med<- paste(signif(PFHxS\_vd\_gsd\_pr\_med, 3))  
PFHxS\_vd\_gsd\_med<- paste(signif(PFHxS\_vd\_gsd\_med, 3))  
PFHxS\_cl\_gm\_pr\_med<- paste(signif(PFHxS\_cl\_gm\_pr\_med, 3))  
PFHxS\_cl\_gm\_med<- paste(signif(PFHxS\_cl\_gm\_med, 3))

### Population median estimates [95% CI]

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Parameter | Prior GM | Posterior GM | Prior GSD | Posterior GSD |
| Half-life (years) | 5.29 | 8.3 | 1.56 | 1.57 |
| HL [95% CI] | [2.39-11.8] | [5.38-13.5] | [1.42-1.77] | [1.42-1.77] |
| Volume of distribution | 0.25 | 0.286 | 1.12 | 1.11 |
| [95% CI] | [0.149-0.418] | [0.173-0.456] | [1.01-1.46] | [1-1.45] |
| Clearance | 0.0326 | 0.0246 |  |  |
| [95% CI] | [0.0126-0.0847] | [0.0121-0.0393] | [] | [] |

### Individual Posterior estimates

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | median GM [95% CI] [[98% CI]] | GM calculator input | GSD individual |
| Half-life (years) | 8.12 [ 2.79-21.6 ] [[ 2.19-24.8 ]] | 8.2 | 1.68 |
| Volume of distribution | 0.281 [ 0.161-0.46 ] [[ 0.14-0.561 ]] | 0.281 | 1.32 |
| Clearance (L/kg-yr) | 0.0224 [ 0.00751-0.078 ] [[ 0.00648-0.101 ]] | 0.0237 | 1.79 |

### Session information

## ─ Session info ───────────────────────────────────────────────────────────────  
## setting value   
## version R version 4.0.5 (2021-03-31)   
## os Red Hat Enterprise Linux Server 7.9 (Maipo)  
## system x86\_64, linux-gnu   
## ui X11   
## language (EN)   
## collate en\_US.UTF-8   
## ctype en\_US.UTF-8   
## tz America/New\_York   
## date 2022-01-23   
##   
## ─ Packages ───────────────────────────────────────────────────────────────────  
## package \* version date lib source   
## assertthat 0.2.1 2019-03-21 [2] CRAN (R 4.0.5)  
## backports 1.4.0 2021-11-23 [1] CRAN (R 4.0.5)  
## bayesplot \* 1.8.0 2021-01-10 [2] CRAN (R 4.0.5)  
## broom 0.7.10 2021-10-31 [1] CRAN (R 4.0.5)  
## cachem 1.0.4 2021-02-13 [2] CRAN (R 4.0.5)  
## callr 3.7.0 2021-04-20 [1] CRAN (R 4.0.5)  
## cellranger 1.1.0 2016-07-27 [2] CRAN (R 4.0.5)  
## cli 3.1.0 2021-10-27 [1] CRAN (R 4.0.5)  
## coda \* 0.19-4 2020-09-30 [2] CRAN (R 4.0.5)  
## codetools 0.2-18 2020-11-04 [2] CRAN (R 4.0.5)  
## colorspace 2.0-2 2021-06-24 [1] CRAN (R 4.0.5)  
## crayon 1.4.2 2021-10-29 [1] CRAN (R 4.0.5)  
## DBI 1.1.1 2021-01-15 [2] CRAN (R 4.0.5)  
## dbplyr 2.1.1 2021-04-06 [1] CRAN (R 4.0.5)  
## desc 1.3.0 2021-03-05 [2] CRAN (R 4.0.5)  
## devtools 2.3.2 2020-09-18 [2] CRAN (R 4.0.5)  
## digest 0.6.29 2021-12-01 [1] CRAN (R 4.0.5)  
## dplyr \* 1.0.7 2021-06-18 [1] CRAN (R 4.0.5)  
## ellipsis 0.3.2 2021-04-29 [1] CRAN (R 4.0.5)  
## evaluate 0.14 2019-05-28 [2] CRAN (R 4.0.5)  
## fansi 0.5.0 2021-05-25 [1] CRAN (R 4.0.5)  
## farver 2.1.0 2021-02-28 [2] CRAN (R 4.0.5)  
## fastmap 1.1.0 2021-01-25 [2] CRAN (R 4.0.5)  
## forcats \* 0.5.1 2021-01-27 [2] CRAN (R 4.0.5)  
## fs 1.5.1 2021-11-30 [1] CRAN (R 4.0.5)  
## generics 0.1.1 2021-10-25 [1] CRAN (R 4.0.5)  
## ggplot2 \* 3.3.5 2021-06-25 [1] CRAN (R 4.0.5)  
## ggridges 0.5.3 2021-01-08 [2] CRAN (R 4.0.5)  
## ggsci \* 2.9 2018-05-14 [2] CRAN (R 4.0.5)  
## glue 1.5.1 2021-11-30 [1] CRAN (R 4.0.5)  
## gtable 0.3.0 2019-03-25 [2] CRAN (R 4.0.5)  
## haven 2.4.3 2021-08-04 [1] CRAN (R 4.0.5)  
## here \* 1.0.1 2020-12-13 [2] CRAN (R 4.0.5)  
## highr 0.9 2021-04-16 [1] CRAN (R 4.0.5)  
## hms 1.1.1 2021-09-26 [1] CRAN (R 4.0.5)  
## htmltools 0.5.2 2021-08-25 [1] CRAN (R 4.0.5)  
## httr 1.4.2 2020-07-20 [2] CRAN (R 4.0.5)  
## jsonlite 1.7.2 2020-12-09 [2] CRAN (R 4.0.5)  
## khroma \* 1.5.0 2021-04-23 [1] CRAN (R 4.0.5)  
## knitr 1.36 2021-09-29 [1] CRAN (R 4.0.5)  
## labeling 0.4.2 2020-10-20 [2] CRAN (R 4.0.5)  
## lattice 0.20-41 2020-04-02 [2] CRAN (R 4.0.5)  
## lifecycle 1.0.1 2021-09-24 [1] CRAN (R 4.0.5)  
## lubridate 1.8.0 2021-10-07 [1] CRAN (R 4.0.5)  
## magrittr 2.0.1 2020-11-17 [2] CRAN (R 4.0.5)  
## memoise 2.0.0 2021-01-26 [2] CRAN (R 4.0.5)  
## modelr 0.1.8 2020-05-19 [2] CRAN (R 4.0.5)  
## munsell 0.5.0 2018-06-12 [2] CRAN (R 4.0.5)  
## pillar 1.6.4 2021-10-18 [1] CRAN (R 4.0.5)  
## pkgbuild 1.2.0 2020-12-15 [2] CRAN (R 4.0.5)  
## pkgconfig 2.0.3 2019-09-22 [2] CRAN (R 4.0.5)  
## pkgload 1.2.0 2021-02-23 [2] CRAN (R 4.0.5)  
## plyr 1.8.6 2020-03-03 [2] CRAN (R 4.0.5)  
## prettyunits 1.1.1 2020-01-24 [2] CRAN (R 4.0.5)  
## pROC 1.18.0 2021-09-03 [1] CRAN (R 4.0.5)  
## processx 3.5.2 2021-04-30 [1] CRAN (R 4.0.5)  
## ps 1.6.0 2021-02-28 [2] CRAN (R 4.0.5)  
## purrr \* 0.3.4 2020-04-17 [2] CRAN (R 4.0.5)  
## R6 2.5.1 2021-08-19 [1] CRAN (R 4.0.5)  
## ragg 1.1.2 2021-03-17 [2] CRAN (R 4.0.5)  
## Rcpp 1.0.7 2021-07-07 [1] CRAN (R 4.0.5)  
## readr \* 2.1.1 2021-11-30 [1] CRAN (R 4.0.5)  
## readxl 1.3.1 2019-03-13 [2] CRAN (R 4.0.5)  
## remotes 2.3.0 2021-04-01 [2] CRAN (R 4.0.5)  
## reprex 2.0.1 2021-08-05 [1] CRAN (R 4.0.5)  
## reshape2 \* 1.4.4 2020-04-09 [2] CRAN (R 4.0.5)  
## rlang 0.4.12 2021-10-18 [1] CRAN (R 4.0.5)  
## rmarkdown 2.11 2021-09-14 [1] CRAN (R 4.0.5)  
## rprojroot 2.0.2 2020-11-15 [2] CRAN (R 4.0.5)  
## rstudioapi 0.13 2020-11-12 [2] CRAN (R 4.0.5)  
## rvest 1.0.2 2021-10-16 [1] CRAN (R 4.0.5)  
## scales \* 1.1.1 2020-05-11 [2] CRAN (R 4.0.5)  
## sessioninfo 1.1.1 2018-11-05 [2] CRAN (R 4.0.5)  
## stringi 1.7.6 2021-11-29 [1] CRAN (R 4.0.5)  
## stringr \* 1.4.0 2019-02-10 [2] CRAN (R 4.0.5)  
## systemfonts 1.0.3 2021-10-13 [1] CRAN (R 4.0.5)  
## testthat 3.0.2 2021-02-14 [2] CRAN (R 4.0.5)  
## textshaping 0.3.3 2021-03-16 [2] CRAN (R 4.0.5)  
## tibble \* 3.1.6 2021-11-07 [1] CRAN (R 4.0.5)  
## tidyr \* 1.1.4 2021-09-27 [1] CRAN (R 4.0.5)  
## tidyselect 1.1.1 2021-04-30 [1] CRAN (R 4.0.5)  
## tidyverse \* 1.3.1 2021-04-15 [1] CRAN (R 4.0.5)  
## tzdb 0.2.0 2021-10-27 [1] CRAN (R 4.0.5)  
## usethis 2.0.1 2021-02-10 [2] CRAN (R 4.0.5)  
## utf8 1.2.2 2021-07-24 [1] CRAN (R 4.0.5)  
## vctrs 0.3.8 2021-04-29 [1] CRAN (R 4.0.5)  
## withr 2.4.3 2021-11-30 [1] CRAN (R 4.0.5)  
## xfun 0.28 2021-11-04 [1] CRAN (R 4.0.5)  
## xml2 1.3.3 2021-11-30 [1] CRAN (R 4.0.5)  
## yaml 2.2.1 2020-02-01 [2] CRAN (R 4.0.5)  
## yardstick \* 0.0.9 2021-11-22 [1] CRAN (R 4.0.5)  
##   
## [1] /home/ad.abt.local/layc/R/x86\_64-pc-linux-gnu-library/4.0  
## [2] /opt/R/4.0.5/lib64/R/library