```
library(vLiverPBPK)

library(gdata)

setwd("\\\\Aa/ord/RTP/Data/Lab/NCCT_vLiver/projects/v-Liver-Sim/vLiverPBPK/R/Unneeded")

chem.prop <- read.xls("vLiver-Chem-Prop-100212.xlsx",stringsAsFactors=FALSE,skip=2)

setwd("\\\\Aa/ord/RTP/Data/Lab/NCCT_vLiver/projects/v-Liver-Chems")
setwd("/Volumes/NCCT_vLiver/projects/v-Liver-Chems")
load("\\\\Aa\\ord\\RTP\\DATA\\Lab\\NCCT_vLiver\\projects\\v-Liver-Sim\\vLiverPBPK\\R\\sysdata.rda")

in.vivo.data <- read.xls("HT-PBPK compounds-102212.xlsx",stringsAsFactors=FALSE,sheet=3)
#in.vivo.data[is.na(in.vivo.data)] <- ""
in.vivo.data[in.vivo.data==" "] <- ""
for (this.col in (1:length(colnames(in.vivo.data))))
  if (is.na(in.vivo.data[1,this.col]))
  {
    this.row <- 1
    while (is.na(in.vivo.data[this.row,this.col]))
    {
      in.vivo.data[this.row,this.col] <- ""
      this.row <- this.row + 1
    }
    this.row <- dim(in.vivo.data)[1]
    while (is.na(in.vivo.data[this.row,this.col]))
    {
      in.vivo.data[this.row,this.col] <- ""
      this.row <- this.row - 1
    }
  }

PK.data.in.vivo <- NULL
this.col <- 1
this.chem <- NA
for (this.col in (1:length(colnames(in.vivo.data)))[in.vivo.data[1,]!=""])
{
  this.CAS <- NA
  this.species <- NA
  this.reference <- NA
  this.weight <- NA
  this.weight.units <- NA
  this.time <- NA
  this.time.units <- NA
  this.media <- NA
  this.media.units <- NA
  this.dose <- NA
  this.dose.units <- NA
  this.value <- NA
  print(paste(this.col))
  if (in.vivo.data[1,this.col]!="")
  {
    if (regexpr("\\.",colnames(in.vivo.data)[this.col])==-1) this.chem <- colnames(in.vivo.data)[this.col]
    if (any(chem.prop[,1]==this.chem)) this.CAS <- chem.prop[chem.prop[,1]==this.chem,2]
    if (in.vivo.data[2,this.col]!="") this.species <- in.vivo.data[2,this.col]
    if (in.vivo.data[1,this.col]!="") this.reference <- in.vivo.data[1,this.col]
```

```
    if (in.vivo.data[3,this.col]!="") this.weight <- in.vivo.data[3,this.col]
    if (in.vivo.data[4,this.col]!="") this.weight.units <- in.vivo.data[4,this.col]
    if (in.vivo.data[6,this.col]!="") this.time.units <- in.vivo.data[6,this.col]
    this.first.col <- this.col
    this.col <- this.col + 1
    while (in.vivo.data[1,this.col]=="")
    {
      this.row.index <- 8
      while(is.na(in.vivo.data[this.row.index,this.col]))
      {
        this.row.index <- this.row.index + 1
        if (this.row.index > dim(in.vivo.data)[1]) break()
      }
      if (this.row.index > dim(in.vivo.data)[1]) break()
      if (in.vivo.data[3,this.col]!="") this.dose <- in.vivo.data[3,this.col]
      if (in.vivo.data[4,this.col]!="") this.dose.units <- in.vivo.data[4,this.col]
      if (in.vivo.data[5,this.col]!="") this.media <- in.vivo.data[5,this.col]
      if (in.vivo.data[6,this.col]!="") this.media.units <- in.vivo.data[6,this.col]
      this.row.stub <- data.frame(this.chem,stringsAsFactors=FALSE)
      this.row.stub <- cbind(this.row.stub,data.frame(this.CAS,stringsAsFactors=FALSE))
      this.row.stub <- cbind(this.row.stub,data.frame(this.reference,stringsAsFactors=FALSE))
      this.row.stub <- cbind(this.row.stub,data.frame(this.species,stringsAsFactors=FALSE))
      this.row.stub <- cbind(this.row.stub,data.frame(as.numeric(this.weight),stringsAsFactors=FALSE))
      this.row.stub <- cbind(this.row.stub,data.frame(this.weight.units,stringsAsFactors=FALSE))
      this.row.stub <- cbind(this.row.stub,data.frame(as.numeric(this.dose),stringsAsFactors=FALSE))
      this.row.stub <- cbind(this.row.stub,data.frame(this.dose.units,stringsAsFactors=FALSE))
      while (in.vivo.data[this.row.index,this.col] != "")
      {
        this.time <- in.vivo.data[this.row.index,this.first.col]
        this.value <- in.vivo.data[this.row.index,this.col]
        this.row <- cbind(this.row.stub,data.frame(as.numeric(this.time),stringsAsFactors=FALSE))
        this.row <- cbind(this.row,data.frame(this.time.units,stringsAsFactors=FALSE))
        this.row <- cbind(this.row,data.frame(this.media,stringsAsFactors=FALSE))
        this.row <- cbind(this.row,data.frame(this.media.units,stringsAsFactors=FALSE))
        this.row <- cbind(this.row,data.frame(as.numeric(this.value),stringsAsFactors=FALSE))
        colnames(this.row) <- c("Compound","CAS","Reference","Species","Species Weight","Species
Weight Units","Dose","Dose Units and Type","Time","Time Units","Media","Media Units","Value")
        PK.data.in.vivo <- rbind(PK.data.in.vivo,this.row)
        print(paste(this.row.index,this.col))
        this.row.index <- this.row.index + 1
        while(is.na(in.vivo.data[this.row.index,this.col]))
        {
          this.row.index <- this.row.index + 1
          if (this.row.index > dim(in.vivo.data)[1]) break()
#           browser()
        }
        if (this.row.index > dim(in.vivo.data)[1]) break()
      }
      this.col <- this.col + 1
      if (this.col > dim(in.vivo.data)[2]) break()
    }
  }
}

#save(PK.data.in.vivo,file="PK.data.in.vivo-01.RData")
```

```
unique(PK.data.in.vivo$Compound[is.na(PK.data.in.vivo$CAS)])

chem.prop <-
chem.prop[chem.prop$Human=="Y",c("Compound","CAS","MW","logP","pKa","Human.Clint","Human.Fub
")]
colnames(chem.prop)[6] <- "Clint"
colnames(chem.prop)[7] <- "Fub"




plot.chem.pk <-
function(this.CAS=NULL,this.name=NULL,xlim=NULL,ylim=NULL,cex.lab=1.5,cex=2,log="y",col.vec=c("bl
ue","red","green","blueviolet","springgreen","navyblue"),pch.vec=c(19,15,17,21,22,24,25,23))
{
  if (is.null(this.CAS) & is.null(this.name))
  {
    cat("Must specifiy compound name or CAS.\n")
  } else if ((!is.null(this.CAS) & !any(PK.data.in.vivo$CAS==this.CAS)) & (!is.null(this.name) &
!any(PK.data.in.vivo$Compound==this.name)))
  {
    cat("Compound not found.\n")
  } else {
    if (!is.null(this.CAS))
    {
      this.chem.data <- subset(PK.data.in.vivo,CAS==this.CAS)
      this.name <- this.chem.data[1,"Compound"]
    } else {
      this.chem.data <- subset(PK.data.in.vivo,Compound==this.name)
      this.CAS <- this.chem.data[1,"CAS"]
    }
    if (is.null(xlim)) xlim <- c(0,max(this.chem.data$Time))
    if (is.null(ylim)) ylim <- c(min(this.chem.data$Value)/10,max(this.chem.data$Value)*5)
    if (length(unique(this.chem.data[,"Time Units"]))>1) browser()
    time.units <- this.chem.data[1,"Time Units"]
    if (length(unique(this.chem.data[,"Media"]))>1) browser()
    media <- this.chem.data[1,"Media"]
    if (length(unique(this.chem.data[,"Media Units"]))>1) browser()
    media.units <- this.chem.data[1,"Media Units"]
    plot(-1,-
1,xlim=xlim,ylim=ylim,cex.lab=cex.lab,xlab=paste("Time",time.units),ylab=paste(media,media.units),main
=paste(this.name," (",this.CAS,")",sep=""),log=log)
    if (regexpr("x",log)!=-1)
    {
      legend.x1 <- exp(log(xlim[1])+(log(xlim[2])-log(xlim[1]))*0.7)
      legend.x2 <- exp(log(xlim[1])+(log(xlim[2])-log(xlim[1]))*0.72)
    } else {
      legend.x1 <- xlim[1]+(xlim[2]-xlim[1])*0.7
      legend.x2 <- xlim[1]+(xlim[2]-xlim[1])*0.72
    }
    for (this.dose in unique(this.chem.data$Dose))
    {
      this.index <- which(unique(this.chem.data$Dose)==this.dose)

matplot(this.chem.data[this.chem.data$Dose==this.dose,"Time"],this.chem.data[this.chem.data$Dose==t
```

```
his.dose,"Value"],pch=pch.vec[this.index],col=col.vec[this.index],cex=cex,add=T)
      if (regexpr("y",log)!=-1) legend.y <- exp(log(ylim[1])+(log(ylim[2])-log(ylim[1]))*0.1*this.index)
      else legend.y <- ylim[1]+(ylim[2]-ylim[1])*0.1*this.index
      matplot(legend.x1,legend.y,pch=pch.vec[this.index],col=col.vec[this.index],cex=1.25*cex,add=T)
      text(legend.x2,legend.y,paste(this.dose,this.chem.data[this.chem.data$Dose==this.dose,"Dose Units
and Type"][1]),cex=cex/1.5,pos=4)
    }
    return(this.chem.data)
  }
}

plot.chem.pk(this.name="Deltamethrin")
```

```
plot.chem.pk <-
function(this.CAS=NULL,this.name=NULL,predict=T,compartment="Cven",xlim=NULL,ylim=NULL,cex.lab
=1.5,cex=2,log="y",col.vec=c("blue","red","green","blueviolet","springgreen","navyblue"),pch.vec=c(19,15,
17,21,22,24,25,23))
{
  if (is.null(this.CAS) & is.null(this.name))
  {
    cat("Must specifiy compound name or CAS.\n")
  } else if ((!is.null(this.CAS) & !any(PK.data.in.vivo$CAS==this.CAS)) & (!is.null(this.name) &
!any(PK.data.in.vivo$Compound==this.name)))
  {
    cat("Compound not found.\n")
  } else {
    if (!is.null(this.CAS))
    {
      this.chem.data <- subset(PK.data.in.vivo,CAS==this.CAS)
      this.name <- this.chem.data[1,"Compound"]
    } else {
      this.chem.data <- subset(PK.data.in.vivo,Compound==this.name)
      this.CAS <- this.chem.data[1,"CAS"]
    }
    chem.prop.index <- which(chem.prop$CAS==this.CAS)

    if (regexpr("x",log)!=-1) this.chem.data <- this.chem.data[this.chem.data[,"Time"]!=0,]
    if (regexpr("y",log)!=-1)  this.chem.data <- this.chem.data[this.chem.data[,"Value"]>0,]
    if (is.null(xlim)) xlim <- c(0,max(this.chem.data$Time))
    if (is.null(ylim)) ylim <- c(min(this.chem.data$Value)/30,max(this.chem.data$Value)*5)
    if (length(unique(this.chem.data[,"Time Units"]))>1) browser()
    time.units <- this.chem.data[1,"Time Units"]
    if (length(unique(this.chem.data[,"Media"]))>1) browser()
    media <- this.chem.data[1,"Media"]
    if (length(unique(this.chem.data[,"Media Units"]))>1) browser()
```

```
    media.units <- this.chem.data[1,"Media Units"]
    dose.units <- unique(this.chem.data[,"Dose Units and Type"])
    plot(-1,-
1,xlim=xlim,ylim=ylim,cex.lab=cex.lab,xlab=paste("Time",time.units),ylab=paste(media,media.units),main
=paste(this.name," (",this.CAS,")",sep=""),log=log)
    if (regexpr("x",log)!=-1)
    {
      legend.x1 <- exp(log(xlim[1])+(log(xlim[2])-log(xlim[1]))*0.7)
      legend.x2 <- exp(log(xlim[1])+(log(xlim[2])-log(xlim[1]))*0.72)
    } else {
      legend.x1 <- xlim[1]+(xlim[2]-xlim[1])*0.3
      legend.x2 <- xlim[1]+(xlim[2]-xlim[1])*0.32
    }
    if (regexpr("y",log)!=-1)  this.chem.data <- this.chem.data[this.chem.data[,"Value"]>0,]
    this.index <- 0
    for (this.study in unique(this.chem.data[,"Reference"]))
    {
      this.chem.species <- this.chem.data[this.chem.data[,"Reference"]==this.study,"Species"][1]
      this.phys.species <- this.chem.species
      if (predict)
      {
        if (!(this.chem.species %in% colnames(chem.prop)))
        {
          if (toupper(this.chem.species) %in% toupper(colnames(chem.prop)))
          {
            new.species <-
colnames(chem.prop)[toupper(colnames(chem.prop))==toupper(this.chem.species)]
            warning(paste(this.chem.species,"coerced to",new.species))
            this.chem.species <- new.species
          } else {
            warning(paste("Chemical specific data for ",this.chem.species,"not found, using Human"))
            this.chem.species <- "Human"
          }
        }
        if (chem.prop[chem.prop.index,this.chem.species]=="N")
        {
          if (chem.prop[chem.prop.index,"Human"]=="Y")
          {
            warning(paste("Chemical specific data for ",this.chem.species,"incomplete, using Human"))
            this.chem.species <- "Human"
          } else {
            warning("Incomplete chemical specific data, predict=FALSE")
          }
        }
      }

      this.study.subset <- this.chem.data[this.chem.data[,"Reference"]==this.study,]
      for (this.dose.type in unique(this.study.subset[,"Dose Units and Type"]))
      {
        this.dose.subset <- this.study.subset[this.study.subset[,"Dose Units and Type"]==this.dose.type,]
        for (this.dose in unique(this.dose.subset[,"Dose"]))
        {
          this.index <- this.index + 1

matplot(this.dose.subset[this.dose.subset[,"Dose"]==this.dose,"Time"],this.dose.subset[this.dose.subset[,
"Dose"]==this.dose,"Value"],pch=pch.vec[this.index],col=col.vec[this.index],cex=cex,add=T)
```

```
        if (regexpr("y",log)!=-1) legend.y <- exp(log(ylim[1])+(log(ylim[2])-log(ylim[1]))*0.06*this.index)
        else legend.y <- ylim[1]+(ylim[2]-ylim[1])*0.06*this.index
        matplot(legend.x1,legend.y,pch=pch.vec[this.index],col=col.vec[this.index],cex=1.25*cex,add=T)
        text(legend.x2,legend.y,paste(this.dose," ",this.dose.type," ",this.phys.species," 
(",this.study,")",sep=""),cex=cex/1.5,pos=4)

        if (predict)
        {
          this.weight <- this.chem.data[this.chem.data[,"Reference"]==this.study,"Species Weight"][1]
          PBPK.params <- Parameterize_vLiverPBPK(chem.prop[chem.prop.index,"MW"],
                        chem.prop[chem.prop.index,paste(this.chem.species,"Clint",sep=".")],
                        chem.prop[chem.prop.index,paste(this.chem.species,"Fub",sep=".")],
                        chem.prop[chem.prop.index,"pKa"],
                        chem.prop[chem.prop.index,"logP"],
                        species=this.phys.species,
                        BW=this.weight)
          browser()

          if (regexpr("mg/kg",this.dose.type)!=-1)
dose.units.convert=as.numeric(chem.prop$MW[chem.prop.index])/1000
          else browser()
          if (regexpr("iv",this.dose.type)!=-1)
          {
            dose.times <- 0
            params <-
c(PBPK.params,list(ivdose=this.dose*dose.units.convert,DietInput=matrix(c(0,0),ncol=2)))
          } else {
            dose.times <- 0
            params <-
c(PBPK.params,list(DietInput=matrix(c(0,24,48,this.dose*dose.units.convert,0,0),ncol=2)))

          }
          out <- vliver_pbpk(seq(xlim[1],xlim[2],0.1),params)$result
          matplot(out[,"time"],out[,compartment]/dose.units.convert,type="l",col=col.vec[this.index],add=T)
          browser()
        }
      }
    }
  }
  return(this.chem.data)
 }
}


plot.chem.pk(this.name="Hexobarbitone")


plot.chem.pk(this.name="Tolbutamide")
```