Covariate distributions in optimal design

Andrew Hooker

2018-05-10

Introduction

Lets assume that we have a model with a covariate included in the model description. Here we define a one-compartment PK model that has weight on both clearance and volume of distribution.

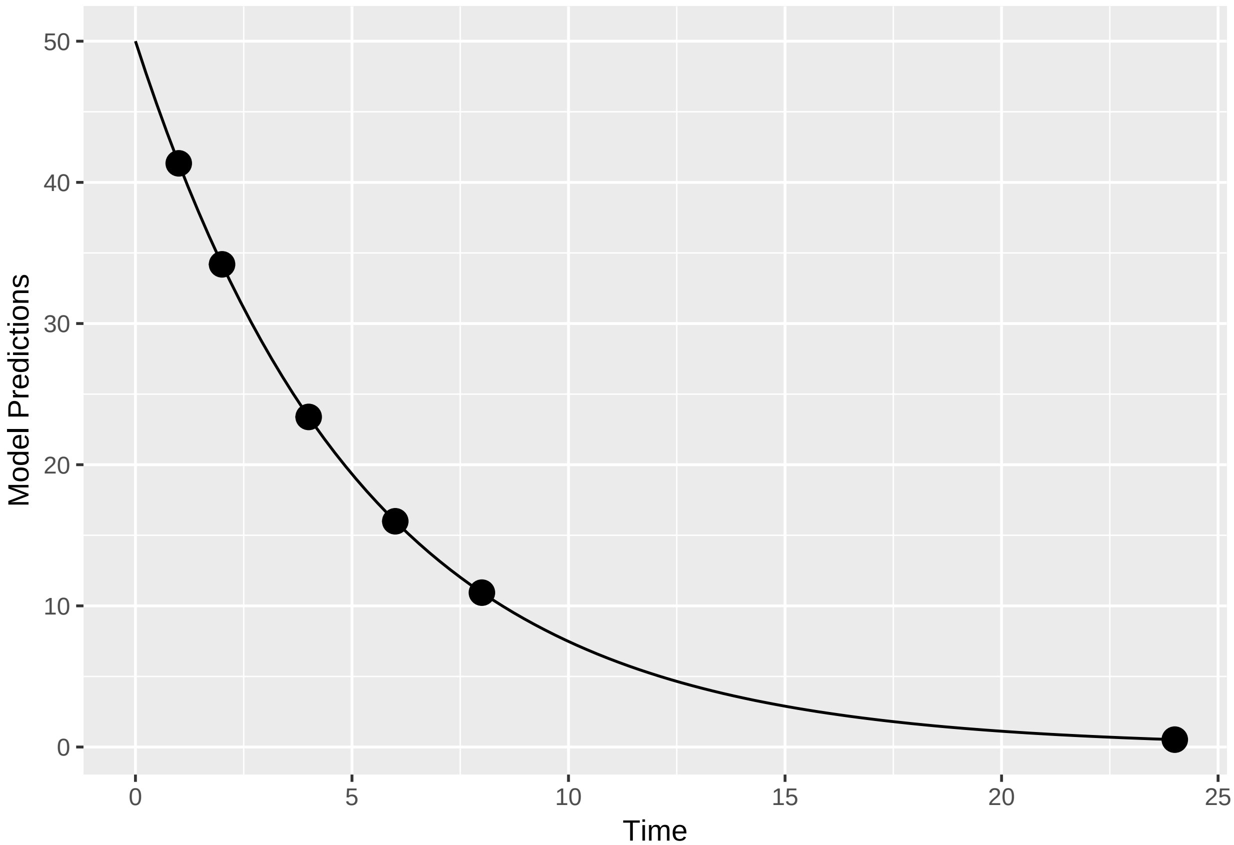
```
mod_1 <- function(model_switch,xt,parameters,poped.db){
  with(as.list(parameters),{
    y=xt

    CL=CL*(WT/70)^(WT_CL)
    V=V*(WT/70)^(WT_V)
    DOSE=1000*(WT/70)
    y = DOSE/V*exp(-CL/V*xt)

    return(list( y= y,poped.db=poped.db))
  })
}

par_1 <- function(x,a,bpop,b,bocc){
  parameters=c( CL=bpop[1]*exp(b[1]),
                V=bpop[2]*exp(b[2]),
                WT_CL=bpop[3],
                WT_V=bpop[4],
                WT=a[1])
  return( parameters )
}
```

Now we define a design. In this case one group of individuals, where we define the individuals' typical weight as 70 kg.

```
poped_db <- create.poped.database(ff_fun=mod_1,
                                  fg_fun=par_1,
                                  fError_fun=feps.add.prop,
                                  groupsize=50,
                                  m=1,
                                  sigma=c(0.015,0.0015),
                                  notfixed_sigma = c(1,0),
                                  bpop=c(CL=3.8,V=20,WT_CL=0.75,WT_V=1),
                                  d=c(CL=0.05,V=0.05),
                                  xt=c( 1,2,4,6,8,24),
                                  minxt=0,
                                  maxxt=24,
                                  bUseGrouped_xt=1,
                                  a=c(WT=70)
                                  )
```

We can create a plot of the model typical predictions:

```r
plot_model_prediction(poped_db)
```



And evaluate the initial design

```r
evaluate_design(poped_db)
```

```
## $ofv
## [1] -Inf
##
## $fim
##             [,1]       [,2] [,3] [,4]       [,5]       [,6]       [,7]
## [1,] 65.8889583 -0.7145374    0    0    0.00000    0.00000      0.000
## [2,] -0.7145374  2.2798156    0    0    0.00000    0.00000      0.000
## [3,]  0.0000000  0.0000000    0    0    0.00000    0.00000      0.000
## [4,]  0.0000000  0.0000000    0    0    0.00000    0.00000      0.000
## [5,]  0.0000000  0.0000000    0    0 9052.31524   29.49016   1424.255
## [6,]  0.0000000  0.0000000    0    0   29.49016 8316.09464   2483.900
## [7,]  0.0000000  0.0000000    0    0 1424.25450 2483.90024 440009.144
##
## $rse
##    bpop[1]    bpop[2]    bpop[3]    bpop[4]     D[1,1]     D[2,2]
##   3.247502   3.317107   0.000000   0.000000  21.026264  21.950179
## SIGMA[1,1]
##  10.061292
```

We see that the covariate parameters can not be estimated according to this design calculation (RSE of bpop[3]=0 and bpop[4]=0). Why is that? Well, the calculation being done is assuming that every individual in the group has the same covariate (to speed up the calculation). This is clearly a poor prediction in this case!

distribution of covariates

We can improve the computation by assuming a distribution of the covariate (WT) in the individuals in the study. We set `groupsize=1`, the number of groups to be 50 (`m=50`) and assume that WT is sampled from a normal distribution with mean=70 and sd=10 (`a=as.list(rnorm(50,mean = 70,sd=10))`).

```
poped_db_2 <- create.poped.database(ff_fun=mod_1,
                                    fg_fun=par_1,
                                    fError_fun=feps.add.prop,
                                    groupsize=1,
                                    m=50,
                                    sigma=c(0.015,0.0015),
                                    notfixed_sigma = c(1,0),
                                    bpop=c(CL=3.8,V=20,WT_CL=0.75,WT_V=1),
                                    d=c(CL=0.05,V=0.05),
                                    xt=c( 1,2,4,6,8,24),
                                    minxt=0,
                                    maxxt=24,
                                    bUseGrouped_xt=1,
                                    a=as.list(rnorm(50,mean = 70,sd=10))
                                    )
```

```
evaluate_design(poped_db_2)
```

```
## $ofv
## [1] 42.41762
##
## $fim
##              [,1]          [,2]          [,3]         [,4]       [,5]
## [1,] 65.888287848 -0.7136861038  0.5585881517  0.002432872    0.00000
## [2,] -0.713686104  2.2793196181  0.0004622458  0.105518175    0.00000
## [3,]  0.558588152  0.0004622458 23.6740918128 -1.357228443    0.00000
## [4,]  0.002432872  0.1055181749 -1.3572284428 22.667124072    0.00000
## [5,]  0.000000000  0.0000000000  0.0000000000  0.000000000 9052.13487
## [6,]  0.000000000  0.0000000000  0.0000000000  0.000000000   29.42624
## [7,]  0.000000000  0.0000000000  0.0000000000  0.000000000 1424.86466
##            [,6]       [,7]
## [1,]    0.00000      0.000
## [2,]    0.00000      0.000
## [3,]    0.00000      0.000
## [4,]    0.00000      0.000
## [5,]   29.42624   1424.865
## [6,] 8312.48968   2483.981
## [7,] 2483.98140 439623.011
##
## $rse
##    bpop[1]    bpop[2]    bpop[3]    bpop[4]     D[1,1]     D[2,2]
##   3.247838   3.317820  27.453204  21.042441  21.026482  21.954963
## SIGMA[1,1]
##  10.065726
```

Here we see that, given this distribution of weights, the covariate effect parameters (bpop[3] and bpop[4]) would be well estimated.

However, we are only looking at one sample of 50 individuals. Maybe a better approach is to look at the distribution of RSEs over a number of experiments given the expected weight distribution.

```r
nsim <- 10
rse_list <- c()
for(i in 1:nsim){
  poped_db_tmp <- create.poped.database(ff_fun=mod_1,
                                        fg_fun=par_1,
                                        fError_fun=feps.add.prop,
                                        groupsize=1,
                                        m=50,
                                        sigma=c(0.015,0.0015),
                                        notfixed_sigma = c(1,0),
                                        bpop=c(CL=3.8,V=20,WT_CL=0.75,WT_V=1),
                                        d=c(CL=0.05,V=0.05),
                                        xt=c( 1,2,4,6,8,24),
                                        minxt=0,
                                        maxxt=24,
                                        bUseGrouped_xt=1,
                                        a=as.list(rnorm(50,mean = 70,sd=10)))
  rse_tmp <- evaluate_design(poped_db_tmp)$rse
  rse_list <- rbind(rse_list,rse_tmp)
}
apply(rse_list,2,quantile)
```

```
##       bpop[1]  bpop[2]  bpop[3]  bpop[4]   D[1,1]   D[2,2] SIGMA[1,1]
## 0%   3.247964 3.317812 24.72016 18.95437 21.02506 21.95166   10.06241
## 25%  3.254492 3.324437 27.10213 20.77113 21.02616 21.95322   10.06412
## 50%  3.281798 3.352476 28.71941 22.00939 21.02816 21.95661   10.06769
## 75%  3.319606 3.390923 32.19956 24.67480 21.02952 21.96031   10.07153
## 100% 3.364808 3.437034 37.67678 28.86794 21.03021 21.96150   10.07251
```