

Supplementary materials 4

R code for data simulation

```
#####  
# The simulation model #  
#####  
  
model<- function(tm, y, parameter) {  
  
  yd1 <- parameter["s"] + parameter["r"]*(1-(y[1] +  
    y[2])/parameter["Tmax"])*y[1] -  
    parameter["beta"]*y[3]*y[1] - parameter["d"]*y[1]  
    #dT/dt  
  
  yd2 <- parameter["beta"]*y[3]*y[1] - parameter["delta"]*y[2]  
    #dI/dt  
  
  yd3 <- (1-parameter["epsilon"])*parameter["p"]*y[2] -  
    parameter["c"]*y[3]  
  
  list(c(yd1,yd2,yd3))  
}  
  
#####  
# Simulation of individual profiles #  
#####  
  
library(deSolve)  
simi<- function(fixed, random, error, samptime, LOD) {  
  
# Simulation of individualparameters  
  eta_Tmax<- rnorm(1,mean=0,sd=random["Tmax"])  
  Tmax<- fixed["Tmax"]*exp(eta_Tmax)  
  eta_r<- rnorm(1,mean=0,sd=random["r"])  
  r <- fixed["r"]*exp(eta_r)  
  eta_d<- rnorm(1,mean=0,sd=random["d"])  
  d <- fixed["d"]*exp(eta_d)  
  eta_beta<- rnorm(1,mean=0,sd=random["beta"])  
  beta<- fixed["beta"]*exp(eta_beta)  
  eta_delta<- rnorm(1,mean=0,sd=random["delta"])  
  delta<- fixed["delta"]*exp(eta_delta)  
  eta_p<- rnorm(1,mean=0,sd=random["p"])  
  p <- fixed["p"]*exp(eta_p)  
  eta_c<- rnorm(1,mean=0,sd=random["c"])  
  c <- fixed["c"]*exp(eta_c)  
  eta_epsilon<- rnorm(1,mean=0,sd=random["epsilon"])  
  epsilon<- fixed["epsilon"]/(fixed["epsilon"]+(1-  
    fixed["epsilon"])*exp(-eta_epsilon))  
  eta_s<- rnorm(1,mean=0,sd=random["s"])  
  eta_s<- rnorm(1,mean=0,sd=random["s"])  
  s <- fixed["s"]*exp(eta_s)
```

```

# Simulation of individual profiles
parameter<- c(Tmax, r, d, beta, delta, p, c, epsilon, s)
random<- c(eta_Tmax=eta_Tmax, eta_r=eta_r, eta_d=eta_d,
          eta_beta=eta_beta,eta_delta=eta_delta, eta_p=eta_p,
          eta_c=eta_c, eta_epsilon=eta_epsilon, eta_s = eta_s)
T0 <- delta*c/beta/p
I0 <- (s + r*T0*(1-T0/Tmax) - d*T0)/(r*T0/Tmax+delta)
V0 <- p*I0/c
res<- lsoda(c(T0, I0, V0),samptime,model,parameter,rtol=1e-
          8,atol=0)
colnames(res) <- c("Time","Tcells","Icells","VLoad")
res<- as.data.frame(res)

# Add residual errors to simulated viral load
er<- rnorm(length(samptime),mean=0,sd=error)
res$Log_VL<- log10(res$VLoad)+er
res$VL<- res$VLoad*10^er

# Taking into account the cure boundary
cure_boundary<- 1/15000 # 1 cell/15 L fluid
length_omit<- sum(res$Icells<cure_boundary)
datasim<- res
if (length_omit> 0) {
omit<- which(res$Icells<cure_boundary)[1]
datasim$VL[c(omit:length(samptime))] <- LOD
datasim$Log_VL[c(omit:length(samptime))] <- log10(LOD)
datasim$Log_VL[datasim$VL<45] <- log10(45)
datasim$VL[datasim$VL<45] <- 45
}

# Define simulated response
if (sum(res$Icells[res$Time<=168]<cure_boundary)>0) {
parameter<- c(parameter,SVR=1)
} else parameter <- c(parameter,SVR=0)
if (sum(res$VL[res$Time==28]<LOD)>0) {
parameter<- c(parameter,RVR4=1)
} else parameter <- c(parameter,RVR4=0)
return(list(datasim=datasim,ipar=parameter,irandom = random))
}

#####
# Simulation individual profiles of n_patients (dataset) #
#####

simall<- function(n_patients,fixed,random,error,samptime,LOD) {

alldata<- c()
allpar<- c()
alleta<- c()

for (i in 1:n_patients) {
sim<- simi(fixed,random,error,samptime,LOD)
idata<- cbind(sim$datasim,ID=rep(i,dim(sim$datasim)[1]))
alldata<- rbind(alldata,idata)
ipar<- c(sim$ipar,id=i)
allpar<- rbind(allpar,ipar)
}
}

```

```

iran<- c(sim$irandom,id=i)
alleta<- rbind(alleta,iran)}

return(list(allpar=allpar,alleta=alleta,alldata=allldata)
}

#####
# Data simulation #
#####

fixed<- c(Tmax=1.3e7, r=0.006, d=0.001, beta=1e-7, delta=0.32, p=50,
          c=9, epsilon=0.996, s=60000)

random<- c(Tmax=0, r=0, d=0, beta=0.5, delta=0.5, p=0, c=0.5,
          epsilon=2.5, s=0)

error<- 0.2 # instead of 0.461 because in the simulation code, we used
10^error instead of exp(error)

samptime<- c(0, 1, 4, 7, 14, 28, 42, 56, 84, 112, 140, 168, 196,
            224, 252, 280, 308, 336)
# we first used a 48 week design then change to 24 week design

set.seed(123456)
simulation<- simall(1000,fixed,random,error,samptime,LOD=45)

indiv_parameter<- as.data.frame(simulation$allpar)

dataD24w <- simulation$alldata[-which(simulation$alldata$Time>168),]
dataD4w <- dataD24w[which(dataD24w$Time<=28),]
dataD4w_sparse <- dataD4w[which((dataD4w$Time!=1)&
(dataD4w$Time!=4)),]
dataD4w_challenge <- dataD24w[which((dataD24w$Time==0)&
(dataD24w$Time==28)),]

# Simulated data for the two first patients

```

Time	ID	log ₁₀ (viral load)	Viral load	CENS
0	1	6.61	4117836.51	0
1	1	3.83	6771.48	0
4	1	3.48	3054.90	0
7	1	3.36	2304.10	0
14	1	2.26	180.97	0
28	1	1.65	45.00	1
42	1	1.65	45.00	1
56	1	1.65	45.00	1
84	1	1.65	45.00	1
112	1	1.65	45.00	1
140	1	1.65	45.00	1
168	1	1.65	45.00	1
0	2	5.90	802621.54	0
1	2	3.98	9586.79	0

4	2	3.58	3824.79	0
7	2	2.81	651.04	0
14	2	1.72	52.56	0
28	2	1.65	45.00	1
42	2	1.65	45.00	1
56	2	1.65	45.00	1
84	2	1.65	45.00	1
112	2	1.65	45.00	1
140	2	1.65	45.00	1
168	2	1.65	45.00	1