Pharmacokinetic model

The simulations presented in the article are based on a notional cytotoxic drug, modeled after docetaxel. The pharmacokinetics are supposed to follow a standard 2-compartment pharmacokinetic model with central and peripheral compartments having volumes $V_c$ and $V_p$, respectively, and drug concentrations $C_c(t)$ and $C_p(t)$. Denoting the intercompartmental and peripheral-compartment clearances $Q$ and $CL$, respectively, and (time-dependent) cumulative dose by $D(t)$, this model is a system of two ordinary differential equations (ODEs):

$$\dot{C}_c = \frac{\dot{D}}{V_c} - \frac{CL}{V_c}C_c - \frac{Q}{V_c}(C_c - C_p)$$
$$\dot{C}_p = \frac{Q}{V_p}(C_c - C_p).$$

Myelosuppression model

Chemotherapy-induced neutropenia (CIN) is supposed to occur according to the semimechanistic model of Friberg et al. (2002). The model may be expressed by the following system of ODEs:

$$\dot{Prol} = k_{tr} \cdot Prol \cdot (1 - E_{drug}) \left( \frac{Circ0}{\text{Circ}} \right) \gamma \cdot k_{tr} \cdot Prol$$
$$\dot{T}_{x1} = k_{tr} (Prol - T_{x1})$$
$$\dot{T}_{x2} = k_{tr} (T_{x1} - T_{x2})$$
$$\dot{T}_{x3} = k_{tr} (T_{x2} - T_{x3})$$
$$\dot{Circ} = k_{tr} (T_{x3} - Circ)$$
Simulated population (inter-individual heterogeneity)

A population of 25 individuals is simulated, using population pharmacokinetic parameters reported for docetaxel in Onoue et al. (2016) and parameters estimated in Friberg et al. (2002) for their myelosuppression model.

N <- 25
dtx.mm <- 0.808 # molar mass (mg/µM) of docetaxel

pop <- data.frame(id=1:N)
  # From Friberg et al 2002 (Table 4, row 1), taking sdlog ~= CV
  Circ0=rlnorm(N, meanlog=log(5050), sdlog=0.42) # units=cells/mm^3
  MTT=rlnorm(N, meanlog=log(89.3), sdlog=0.16) # mean transit time
  gamma=rlnorm(N, meanlog=log(0.163), sdlog=0.039) # feedback factor
  Emax=rlnorm(N, meanlog=log(83.9), sdlog=0.33)
  EC50=rlnorm(N, meanlog=log(7.17*dtx.mm), sdlog=0.50)
  # PK params from 2-compartment docetaxel model of Onoue et al (2016)
  CL=rlnorm(N, meanlog=log(32.6), sdlog=0.295)
  Q=rlnorm(N, meanlog=log(5.34), sdlog=0.551)
  Vc=rlnorm(N, meanlog=log(5.77), sdlog=0.1) # Onoue gives no CV% for V1
  Vp=rlnorm(N, meanlog=log(11.0), sdlog=0.598) # Called 'V2' in Onoue

) pop <- upData(pop,
  ,kTR = 4/MTT
  ,units = c(Circ0="cells/mm^3"
    ,MTT="hours"
    ,kTR="1/hour"
    ,CL="L/h"
    ,Q="L/h"
    ,Vc="L"
    ,Vp="L"
  )
  ,print=FALSE
)

latex(describe(pop), file="")

11 Variables 25 Observations

id

<table>
<thead>
<tr>
<th>id</th>
<th>missing</th>
<th>distinct</th>
<th>Info</th>
<th>Mean</th>
<th>Gmd</th>
<th>0.05</th>
<th>0.10</th>
<th>0.25</th>
<th>0.50</th>
<th>0.75</th>
<th>0.90</th>
<th>0.95</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>0</td>
<td>25</td>
<td>1</td>
<td>13</td>
<td>8.867</td>
<td>2.2</td>
<td>3.4</td>
<td>7.0</td>
<td>13.0</td>
<td>19.0</td>
<td>22.6</td>
<td>23.8</td>
</tr>
</tbody>
</table>

lowest : 1 2 3 4 5, highest: 21 22 23 24 25

Circ0 [cells/mm^3]

<table>
<thead>
<tr>
<th>Circ0 [cells/mm^3]</th>
<th>missing</th>
<th>distinct</th>
<th>Info</th>
<th>Mean</th>
<th>Gmd</th>
<th>0.05</th>
<th>0.10</th>
<th>0.25</th>
<th>0.50</th>
<th>0.75</th>
<th>0.90</th>
<th>0.95</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>0</td>
<td>25</td>
<td>1</td>
<td>5063</td>
<td>2079</td>
<td>3442</td>
<td>3469</td>
<td>3766</td>
<td>4485</td>
<td>5810</td>
<td>7768</td>
<td>8654</td>
</tr>
</tbody>
</table>

lowest : 1563.57 3439.05 3452.3 3493.87 3664.58, highest: 6484.65 7689.94 7819.93 8862.37 9867.58

MTT [hours]

<table>
<thead>
<tr>
<th>MTT [hours]</th>
<th>missing</th>
<th>distinct</th>
<th>Info</th>
<th>Mean</th>
<th>Gmd</th>
<th>0.05</th>
<th>0.10</th>
<th>0.25</th>
<th>0.50</th>
<th>0.75</th>
<th>0.90</th>
<th>0.95</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>0</td>
<td>25</td>
<td>1</td>
<td>88.45</td>
<td>17.71</td>
<td>70.25</td>
<td>70.33</td>
<td>76.18</td>
<td>84.97</td>
<td>102.10</td>
<td>109.96</td>
<td>110.87</td>
</tr>
</tbody>
</table>

lowest : 67.959 70.2257 70.3491 72.8133 75.1749, highest: 107.017 108.722 110.787 110.889 126.859

γ:

<table>
<thead>
<tr>
<th>γ</th>
<th>missing</th>
<th>distinct</th>
<th>Info</th>
<th>Mean</th>
<th>Gmd</th>
<th>0.05</th>
<th>0.10</th>
<th>0.25</th>
<th>0.50</th>
<th>0.75</th>
<th>0.90</th>
<th>0.95</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>0</td>
<td>25</td>
<td>1</td>
<td>0.1616</td>
<td>0.00735</td>
<td>0.1516</td>
<td>0.1539</td>
<td>0.1574</td>
<td>0.1613</td>
<td>0.1661</td>
<td>0.1689</td>
<td>0.1715</td>
</tr>
</tbody>
</table>

lowest : 0.147769 0.151288 0.152986 0.155289 0.155991, highest: 0.167548 0.168357 0.169222 0.172062 0.173726
<table>
<thead>
<tr>
<th></th>
<th>n missing</th>
<th>distinct</th>
<th>Info</th>
<th>Mean</th>
<th>Gmd</th>
<th>.05</th>
<th>.10</th>
<th></th>
<th>.25</th>
<th>.50</th>
<th>.75</th>
<th>.90</th>
<th>.95</th>
<th></th>
<th>lowest</th>
<th>highest</th>
</tr>
</thead>
<tbody>
<tr>
<td>Emax</td>
<td>25</td>
<td>0</td>
<td>25</td>
<td>1</td>
<td>96.19</td>
<td>40.92</td>
<td>52.42</td>
<td>59.63</td>
<td>74.28</td>
<td>89.82</td>
<td>104.67</td>
<td>120.58</td>
<td>129.87</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>39.5723</td>
<td>69.1917</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>109.461</td>
<td>125.794</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>130.888</td>
<td>162.071</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2.9568</td>
<td>4.77465</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>15.055</td>
<td>21.1949</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>21.6482</td>
<td>24.9759</td>
</tr>
<tr>
<td>CL</td>
<td>25</td>
<td>0</td>
<td>25</td>
<td>1</td>
<td>32.72</td>
<td>9.214</td>
<td>19.86</td>
<td>19.95</td>
<td>30.34</td>
<td>32.75</td>
<td>37.57</td>
<td>40.88</td>
<td>41.28</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>18.5327</td>
<td>19.8474</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>19.9772</td>
<td>20.9673</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>40.0639</td>
<td>40.3992</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>41.1974</td>
<td>41.2991</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>52.6234</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1.6063</td>
<td>2.05371</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2.61263</td>
<td>2.76334</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2.94361</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>9.35977</td>
<td>9.37026</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>13.0015</td>
<td>15.762</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>18.82</td>
<td></td>
</tr>
<tr>
<td>Vc</td>
<td>25</td>
<td>0</td>
<td>25</td>
<td>1</td>
<td>5.003</td>
<td>0.558</td>
<td>5.078</td>
<td>5.204</td>
<td>5.409</td>
<td>5.788</td>
<td>6.110</td>
<td>6.449</td>
<td>6.503</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>4.83721</td>
<td>5.14713</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>5.38205</td>
<td>5.38205</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>6.39435</td>
<td>6.40294</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>6.45242</td>
<td>6.5153</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>6.55067</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2.72992</td>
<td>4.83309</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>6.43276</td>
<td>7.07247</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>7.21422</td>
<td>40.4367</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>46.6759</td>
<td>67.4465</td>
</tr>
<tr>
<td>kTR</td>
<td>25</td>
<td>0</td>
<td>25</td>
<td>1</td>
<td>0.0465</td>
<td>0.008853</td>
<td>0.03608</td>
<td>0.03638</td>
<td>0.03918</td>
<td>0.04707</td>
<td>0.05251</td>
<td>0.05609</td>
<td>0.05694</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.031531</td>
<td>0.0360722</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.0361055</td>
<td>0.0367909</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.0373771</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.052092</td>
<td>0.054935</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.0568939</td>
<td>0.0569592</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.058859</td>
<td></td>
</tr>
</tbody>
</table>
PK/PD simulation model

To simulate the pharmacokinetics and (myelosuppressive) pharmacodynamics of our notional cytotoxic drug, we define a `pomp` model as follows:

```r
pkpd.skel <- "
  double c2p = Q*( Cc - Cp ); // central-to-peripheral flux
  DCc = (dose/duration)*(t < duration ? 1.0 : 0.0)/Vc - (CL/Vc)*Cc - c2p/Vc;
  DCp = c2p/Vp;
  // Myelosuppression model (Emax model, then dynamics per eqs 3-7 from Friberg et al 2002
  double Edrug = Emax*Cc/(Cc + EC50); // classic 'Emax model'
  DPro1 = (1-Edrug) * Pro1 * kTR * pow((Circ0 / Circ), gamma) - kTR * Pro1;
  DTx_1 = kTR * (Pro1 - Tx_1);
  DTx_2 = kTR * (Tx_1 - Tx_2);
  DTx_3 = kTR * (Tx_2 - Tx_3);
  DCirc = kTR * (Tx_3 - Circ);
"

pkpd.txform <- "
  T_Circ0 = log(Circ0);
  T_kTR = log(kTR);
  T_Emax = log(Emax);
  T_EC50 = log(EC50);
  T_CL = log(CL);
  T_Q = log(Q);
  T_Vc = log(Vc);
  T_Vp = log(Vp);
  T_sigma = log(sigma);
  T_dose = log(dose);
  T_duration = log(duration);
"

pkpd.txback <- "
  Circ0 = exp(T_Circ0);
  kTR = exp(T_kTR);
  Emax = exp(T_Emax);
  EC50 = exp(T_EC50);
  CL = exp(T_CL);
  Q = exp(T_Q);
  Vc = exp(T_Vc);
  Vp = exp(T_Vp);
  sigma = exp(T_sigma);
  dose = exp(T_dose);
  duration = exp(T_duration);
"

Tmax <- 21*24 # solve for full 21 days of 3-week cycle
df <- data.frame(time=seq(0.0, 1.95, 0.05) # q3min for 2h,
  ,seq(2.0, Tmax, 1.0)) # then hourly until Tmax
  ,y=NA)
pkpd <- pomp(data = df
  ,times="time", t0=0
  ,skeleton=vectorfield(Csnippet(pkpd.skel))
  ,statenames=c("Cc", "Cp", "Pro1", "Tx.1", "Tx.2", "Tx.3", "Circ")
  ,paramnames=c("Circ0", "kTR", "gamma", "Emax", "EC50", "CL", "Q", "Vc", "Vp"
  ,"sigma", "dose", "duration"
  ,"Cc.0", "Cp.0", "Pro1.0", "Tx.1.0", "Tx.2.0", "Tx.3.0", "Circ.0")
"
In each subject, we now infuse the same initial dose over 1 hour, and integrate the PK and myelosuppression ODEs to obtain time series for plotting:

```r
# initial conditions and output times
dose1 <- 100 # mg
tInfusion <- 1 # 1-hour infusion
allout <- data.frame() # accumulator for nrow(pop) individual ODE solutions
parms <- function(id, dose, duration){
  parms <- unlist(pop[id, c('Circ0', 'kTR', 'gamma', 'Emax', 'EC50', 'CL', 'Q', 'Vc', 'Vp'))]
  parms['sigma'] <- 0.05
  parms['dose'] <- dose
  parms['duration'] <- duration
  parms['Cc.0'] <- parms$Cp.0 <- 0.0
  parms[c('Prol.0', 'Tx.1.0', 'Tx.2.0', 'Tx.3.0', 'Circ.0')] <- parms$Circ0
  parms
}

for (id in 1:nrow(pop)) {
  Circ0 <- pop$Circ0[id]
  out <- trajectory(pkpd,
    params=c(pop$pop$id==id, which(names(pop) %in% c('id','MTT')))
    , sigma=0.05
    , dose=dose1
    , duration=tInfusion
    , Cc.0 = 0.0
    , Cp.0 = 0.0
    , Prol.0 = Circ0
    , Tx.1.0 = Circ0
    , Tx.2.0 = Circ0
    , Tx.3.0 = Circ0
    , Circ.0 = Circ0)) |> as.data.frame()
  out <- out[,c('time', 'Cc', 'Cp', 'Prol', 'Tx.1', 'Tx.2', 'Tx.3', 'Circ')]
  out$id <- paste("id", id, sep="")
  allout <- rbind(allout, out)
}

library(data.table)
```

```r
## Attaching package: 'data.table'

## The following object is masked from 'package:pomp':
## melt

## When a function y(x) is sampled around a minimum at equispaced (x-dx, x, x+dx)
## with corresponding values (y-dy1, y, y+dy2) such that dy1 < 0 < dy2 (i.e., with
## the middle value y being lowest value), then a quadratic interpolation yields
```
## estimated minimum at \((x_\star, y_\star)\) given by:

\[ x_\star = x - \frac{dx}{2} \times \frac{(dy1 + dy2)}{(dy2 - dy1)} \]

\[ y_\star = y - \frac{1}{8} \left( \frac{(dy1 + dy2)}{(dy2 - dy1)} \right)^2. \]

for (.id in unique(allout$id)) {
    with(subset(allout, id == .id), {
        nadirIdx <- which.min(Circ)
        Dy1Dy2 <- diff(Circ[nadirIdx + (-1:1)])
        SdyDdy <- sum(Dy1Dy2) / diff(Dy1Dy2)
        allout[id == .id, CircMin := Circ[nadirIdx] - \(\frac{1}{8}\) * SdyDdy^2]
        allout[id == .id, tNadir := time[nadirIdx] - 0.5 * diff(time[nadirIdx+(0:1)]) * SdyDdy]
    })
}

allout <- upData(allout
    , id = ordered(id, levels=paste("id",1:N,sep="")) # Note that we may
    , units=c
        , Prol="cells/mm^3" # treat the non-circulating compartments
        , Tx.1="cells/mm^3" # on a 'circulating-equivalent basis';
        , Tx.2="cells/mm^3" # thus we attach 'cells/mm^3' units to
        , Tx.3="cells/mm^3" # these quantities as well.
        , Circ="cells/mm^3"
        , Cc="ng/mL"
        , Cp="ng/mL"
        , time="hours")
    , print=FALSE
)
cout <- `gather(allout, key="Series", value="Concentration"
          , Cc, Cp,
          , factor_key = TRUE)

label(cout$Concentration) <- "Drug Concentration"

xYplot(Concentration ~ time | id, group=Series
       , data=cout, subset=time<6
       , layout=c(5,5)
       , type='l', as.table=TRUE
       , label.curves=FALSE
       , par.settings=list(superpose.line=list(lwd=2,col=brewer.pal(4,"PRGn")[c(1,4)]))
       , scales=list(y=list(log=TRUE, lim=c(10^-3,10^1)))
       , auto.key=list(points=FALSE, lines=TRUE, columns=2))

Figure 1: Two-compartment pharmacokinetics of the drug.
mout <- gather(allout, key="Series", value="ANC"
    , Prol, Tx.1, Tx.2, Tx.3, Circ
    , factor_key = TRUE)

mout <- upData(mout
    , time = time/24
    , units = c(time="days")
    , print = FALSE)

xYplot(ANC ~ time | id, group=Series
    , data=mout
    , layout=c(5,5)
    , type='l', as.table=TRUE
    , label.curves=FALSE
    , par.settings=list(superpose.line=list(lwd=2, col=brewer.pal(11,"RdYlBu")[c(1,3,4,8,10)]))
    , scales=list(y=list(log=TRUE, lim=c(100,15000), at=c(200, 500, 1000, 2000, 5000, 10000)))
    , auto.key=list(points=FALSE, lines=TRUE, columns=5))
Figure 2: Myelosuppression in the 5-compartment model of Friberg et al., with an infused dose of 100 mg. **Prol**: proliferative compartment; **Tx.n**: transit compartments; **Circ**: circulating cells.
Adaptive dosing based on a simple Newton-Raphson heuristic

The doChemo function defined below implements multiple, 3-week courses of chemotherapy with optional adaptive dosing based on the heuristic of Newton-Raphson root-finding. When adapt.dosing==FALSE, the infusion doses are stepped from 50 up to 500 mg, in increments that are evenly spaced on the scale of \( \sqrt{dose} \).

```r
scaled <- function(dose, a=4.0) dose^(1/a)
dose.scale <- list(lim=scaled(c(40,550)),
  at=scaled(c(50,100,250,500)),
  lab=c('50','100','250','500'))
doChemo <- function(draw.days=NULL, Tcy=3*7/24, adapt.dosing=c('Newton'), omega=0.75) {
  # Find the ANC nadirs of all 20 IDs, checking ANCs on (integer-vector) draw.days
  # We will accumulate data about each course of treatment into this data frame.
  # TODO: Consider doing away entirely with columns Cc...Circ, since these inits are available in 'out'
  hourly <- which(abs(time(pkpd)-round(time(pkpd)))<.Machine$double.eps*0.5)
  anc.ts <- data.frame() # This will be used to collect an hourly 'Circ' time series
  course <- expand.grid(grid=cycle=1:10, id=1:nrow(pop), Cc=0.0, Cp=0.0,
    Prol=NA, Tx.1=NA, Tx.2=NA, Tx.3=NA, Circ=NA
  , dose=NA, ANC.nadir=NA, time.nadir=NA, scaled.dose=NA
  , ANC.nadir.est=NA, time.nadir.est=NA)
  for (day in draw.days) {
    newcolumn <- paste("ANC", day, sep="_d")
    course[,newcolumn] <- NA
    units(course[,newcolumn]) <- "cells/mm^3"
    label(course[,newcolumn]) <- paste("Day-",day," ANC", sep="")
  }
  course$dose <- seq(scaled, from=50, to=500, length.out=max(course$cycle), digits=0)[course$cycle]
  statevector <- c('Cc','Cp','Prol', 'Tx.1','Tx.2','Tx.3','Circ')
  course[,statevector[-(1:2)]] <- pop$Circ0[course$id] # Prol(0)=Tx.1(0)=Tx.2(0)=Tx.3(0)=Circ(0)=Circ0
  for (id in 1:nrow(pop)) {
    # outer loop over IDs permits state cycling
    params["sigma"] <- 0.05
    params["duration"] <- Tinfusion
    params[c('Cc.0','Cp.0')]<- 0.0
    params[c('Prol.0','Tx.1.0','Tx.2.0','Tx.3.0','Circ.0')] <- params['Circ0']
    for (cycle in 1:max(course$cycle)) {
      idx <- which(course$cycle==cycle & course$id==id)
      if (cycle>1) {
        lag_1 <- which(course$cycle==(cycle-1) & course$id==id)
        if (adapt.dosing=='Newton') { # Override preconfigured dose
          omega*delta.scaleddose <- ifelse(is.na(slope), 0, log(500 / course[lag_1,'ANC.nadir']) / slope)
        }
      }
    }
  }
}
```

Adaptive dosing based on a simple Newton-Raphson heuristic

The doChemo function defined below implements multiple, 3-week courses of chemotherapy with optional adaptive dosing based on the heuristic of Newton-Raphson root-finding. When adapt.dosing==FALSE, the infusion doses are stepped from 50 up to 500 mg, in increments that are evenly spaced on the scale of \( \sqrt{dose} \).
new.scaleddose <- scaled(course[lag_1,'dose']) + delta.safer

params['dose'] <- course$dose[idx]

traj <- trajectory(pkpd, params=params) |> as.data.frame()

to.add <- data.frame(id=rep(id,length(hourly))
    , time=traj$time[hourly]*(cycle-1)*Tmax
    , ANC=traj$Circ[hourly])

anc.ts <- rhbind(anc.ts, to.add)

course[idx,c('ANC.nadir','time.nadir')] <- traj[which.min(traj$Circ),c('Circ','time')]

course[idx,statevector] <- traj[which.max(traj$time),statevector]

for (day in draw.days) {
    day.idx <- which(traj$time==day*24)
    course[idx,paste("ANC", day, sep="_d")]<- traj[day.idx,'Circ']
}

if (length(draw.days)) {
    # TODO: Here, estimate nadir level and timing based on fitted spline
    ys <- course[idx,paste("ANC", draw.days, sep="_d")]
    fit <- spline(draw.days, log(ys), n=200)
    course[idx,"ANC.nadir.est"] <- exp(min(fit$y))
    course[idx,"time.nadir.est"] <- fit$x[which.min(fit$y)]
}

}

course <- upData(course[order(course$cycle),]
    , id = ordered(paste("id", id,sep=""))
    , levels=paste("id",1:N,sep=""))
    , time.nadir = time.nadir/24
    , scaled.dose = scaled(dose)
    , labels=c("ANC.nadir=ANC nadir"
        , time.nadir="Time of ANC nadir"
        , dose="Drug Dose"
        , scaled.dose="Drug Dose (rescaled)"
        , units=c("ANC.nadir="cells/mm^3"
        , time.nadir="days"
        , dose="mg"
        , scaled.dose="mg")
    , print=FALSE
)

anc.ts <- upData(anc.ts
    , id = ordered(paste("id", id,sep=""))
    , levels=paste("id",1:N,sep=""))
    , time = time/(24+7)
    , labels=c("ANC=ANC"
        , units=c("ANC="cells/mm^3"
        , time="weeks")
    , print=FALSE
)

list(course=course, anc.ts=anc.ts)
Linearize dynamics

We now demonstrate graphically an approximate linearization of ANC nadir level and timing, under logarithmic transformation of ANC and fourth-root transformation of dose.

```r
chemo <- doChemo(draw.days=c(5,6,7,8,11), adapt.dosing=FALSE)
course <- chemo$course
anc.ts <- chemo$anc.ts

xYplot(ANC.nadir ~ scaled.dose | id, data=course, as.table=TRUE
   , layout=c(5,5)
   , scales=list(y=list(log=TRUE, lim=c(100,10000), at=c(200, 500, 1000, 2000, 5000)),
   x=dose.scale)
```

Figure 3: ANC nadir vs cytotoxic dose. Note the dose axis is scaled to achieve near-linearity.
slopeForID <- function(x){
  fit <- lm(log(ANC.nadir) ~ scaled.dose, data=subset(course, id=x))
  slope <- fit$coefficients['scaled.dose']
  unname(slope)
}

slope <- sapply(levels(course$id), slopeForID)
densityplot(~slope, plot.points="rug")

Figure 4: Slopes of log(\(ANC_{nadir}\)) vs \(\sqrt{\text{dose}}\) for 25 simulated study subjects.
Figure 5: Timing of ANC nadir vs initial cytotoxic dose. Note the dose axis is scaled to achieve near-linearity.
Figure 6: Timing and level of ANC nadir vs initial cytotoxic dose. The doses are equally spaced on a power-law scale that yields a roughly linear parametrization of the nadir coordinate paths.
Would day-7 ANC suffice as a proxy for nadir?

Evidently not; see Figure 7. Thus, CIN monitoring constitutes a nontrivial aspect of any practical implementation of DTAT, and requires explicit modeling in follow-on work. No doubt, the implementation must take the form of an adaptive process designed to balance (a) the burden of multiple blood draws against (b) the need for early warning of an impending severely neutropenic nadir that would indicate prophylactic administration of colony-stimulating factor.

```r
dxYplot(ANC.nadir ~ ANC_d7, data=course, group=id, type='l', as.table=TRUE
        , aspect=1
        , label.curves=list(method="offset")
        , scales=list(x=list(log=TRUE, lim=c(40, 6000), equispaced.log=FALSE),
                      y=list(log=TRUE, lim=c(40, 6000), equispaced.log=FALSE))
        , par.settings=list(superpose.line=list(col="black"))
)
```

Figure 7: Day-7 ANC does not serve as suitable proxy for ANC nadir across the whole modeled population. Note that for some individuals (e.g., id3, id11, id20), the day-7 ANC may dangerously underestimate the actual nadir.
Simulated dose titration in 25 study subjects

```r
cemo <- doChemo(adapt.dosing='Newton')
newton <- chemo$course
new.ts <- chemo$anc.ts
anc.tics <- c(200, 500, 1500, 4000, 10000)
right <- xYplot(ANC ~ time | id, data=new.ts
    , as.table=TRUE, type="l"
    , layout=c(5, 5)
    , scales=list(y=list(log=TRUE, lim=c(100, 1.5e4)
    , at=anc.tics
    , lab=as.character(anc.tics)),
    x=list(at=seq(0, 30, 3)))
)
newton <- upData(newton
    , time = 3*(cycle-1)
    , labels = c(time="Time")
    , units = c(time="weeks")
    , print = FALSE)
dose.tics <- c(50, 200, 600, 1500, 3000)
left <- xYplot(scaled.dose ~ time | id, data=newton
    , as.table=TRUE, type='p', pch='+', cex=1.5
    , layout=c(5, 5)
    , scales=list(y=list(lim=scaled(c(30, 3200)))
    , at=scaled(dose.tics)
    , lab=as.character(dose.tics)),
    x=list(lim=c(-1, 31)
    , at=seq(0, 30, 3)
    , lab=c('0', ',', '6', ',', '12', ',', '18', ',', '24', ',', '30')))
update(doubleYScale(left, right, add.ylab2=TRUE)
    , par.settings = simpleTheme(col=brewer.pal(4, "PRGn"))[c(4, 1)])
```
Figure 8: Dose titration by Newton’s method, to a goal ANC nadir of 500 cells/mm$^3$. 
SessionInfo

This document was produced using:

```
sessionInfo()
```

```
## R version 4.4.0 (2024-04-24)
## Platform: x86_64-pc-linux-gnu
## Running under: PureOS 10 (Byzantium)
##
## Matrix products: default
## BLAS/LAPACK: /usr/lib/x86_64-linux-gnu/openblas-pthread/libopenblas-r0.3.13.so; LAPACK version 3.9
##
## locale:
## [1] LC_CTYPE=en_US.UTF-8   LC_NUMERIC=C   LC_TIME=en_US.UTF-8
## [4] LC_COLLATE=C          LC_MONETARY=en_US.UTF-8 LC_MESSAGES=en_US.UTF-8
## [7] LC_PAPER=en_US.UTF-8  LC_NAME=C       LC_ADDRESS=C
## [10] LC_TELEPHONE=C       LC_MEASUREMENT=en_US.UTF-8 LC_IDENTIFICATION=C
##
## time zone: America/New_York
## tzcode source: system (glibc)
##
## attached base packages:
## [1] stats   graphics  grDevices  utils  datasets  methods  base
##
## other attached packages:
## [1] data.table_1.15.4  pomp_5.8       RColorBrewer_1.1-3 latticeExtra_0.6-30
## [5] tidyr_1.3.1        rms_6.8-0       Hmisc_5.1-2       knitr_1.46
## [9] DTAT_0.3-7         survival_3.6-4  widgetframe_0.3.1  htmlwidgets_1.6.4
## [13] r2d3_0.2.6        lattice_0.22-6
##
## loaded via a namespace (and not attached):
## [1] tidyselect_1.2.1  dplyr_1.1.4       fastmap_1.1.1     TH.data_1.1-2
## [5] promises_1.3.0    digest_0.6.35    rpart_4.1.23      mime_0.12
## [9] lifecycle_1.0.4   cluster_2.1.6    magrittr_2.0.3    compiler_4.4.0
## [13] riang_1.1.3       sass_0.4.9       tools_4.4.0       utf8_1.2.4
## [17] yaml_2.3.8        interp_1.1-6     multcomp_1.4-25   polspline_1.1.24
## [21] withr_3.0.0       foreign_0.8-86   purrr_1.0.2       nnet_7.3-19
## [25] grid_4.4.0        fansi_1.0.6      xtable_1.8-4      colorspace_2.1-0
## [29] ggplot2_3.5.1      scales_1.3.0      MASS_7.3-60.2     tinytex_0.51
## [33] cli_3.6.2         mvtnorm_1.2-4    rmarkdown_2.26    generics_0.1.3
## [37] rstudioapi_0.16.0 km.ci_0.5-6      cachem_1.0.8      stringr_1.5.1
## [41] splines_4.4.0     base64enc_0.1-3  vctrs_0.6.5       Matrix_1.7-0
## [45] sandwich_3.1-0    jsonlite_1.8.8    SparseM_1.81      Formula_1.2-5
## [49] htmlTable_2.4.2   jpeg_0.1-10       jqr_0.1.4         glue_1.7.0
## [53] codetools_0.2-20   stringi_1.8.4    gtable_0.3.5      later_1.3.2
## [57] deldir_2.0-4       munsell_0.5.1    tibble_3.2.1      pillar_1.9.0
## [61] htmltools_0.5.8.1  quantreg_5.97    deSolve_1.40      R6_2.5.1
## [65] evaluate_0.23     shiny_1.8.1.1     higher_0.10       png_0.1-8
## [69] backports_1.4.1    httpuv_1.6.15     bslib_0.7.0       MatrixModels_0.5-3
## [73] Rcpp_1.0.12       coda_0.19-4.1     gridExtra_2.3     nlme_3.1-164
## [77] checkmate_2.3.1    xfun_0.43        zoo_1.8-12        pkgconfig_2.0.3
```