Table S1. Parameter estimates of the best base model of single-dose studies. The population mean values, between subject variability, and shrinkages are shown where appropriate.

Parameter	Population estimate	%SE
CL/F (L/h)	122	3.7
V_2/F (L)	845	4.1
Q/F (L/h)	156	3.8
V_3/F (L)	2320	3.7
KTR (h^{-1})	2.16	2.3
ETASCALE	-0.21	28.6
FORMF	0.714	9.9
FEDKTR	-0.556	2.6
FEDF	-0.249	11.2
BSV-%CV (% Shrinkage)		
BSV-CL/F	21.3 (29.6)	5.7
BSV-V_2/F	29.4 (45.7)	10.5
BSV-KTR	45.6 (3.6)	3.2
FVAR	58.1 (4.8)	4.1
RV-%CV; unless stated otherwise		
Proportional error-Fasted	29.1	3.2
Additive error-Fasted (ng/ml)	0.181	9.2
Proportional error-Fed	40.4	3.3
Additive error-Fed (ng/ml)	0.239	23.1

SE: standard error; BSV: between subject variability; RV; residual variability; CL/F: apparent clearance; V_2/F: apparent central volume of distribution; Q/F: apparent inter-compartmental clearance; V_3/F: apparent peripheral volume of distribution; KTR: Transit compartment rate constant; ETASCALE: Effect of formulation on bioavailability variability; FORMF: Effect of formulation on bioavailability; FEDKTR: Effect of fed status on transit compartment rate constant; FEDF: Effect of fed status on bioavailability; FVAR: Common random effect for bioavailability.

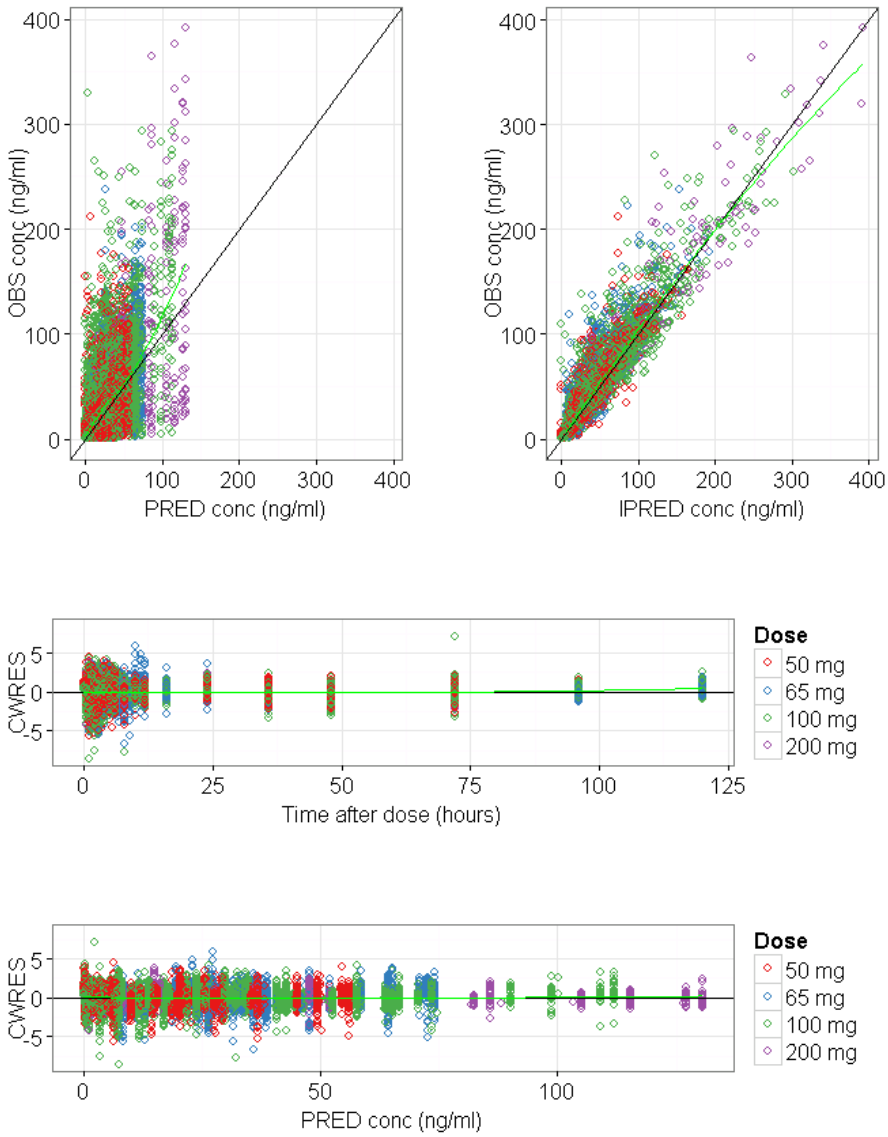

Fig S1. Goodness-of-fit plots for the best base model of single-dose data. In each plot, symbols are data points, the solid black line is a line with slope 1 or 0, and the green line is a Loess smooth of the data. Figure legends defining itraconazole data points for the different itraconazole doses used in the pharmacokinetic studies coloured in red (50 mg), blue (65 mg), green (100mg), and purple (200 mg). OBS conc: observed concentrations; PRED conc: population predicted concentrations; IPRED conc: individual predicted concentration; CWRES: conditional weighted residuals.

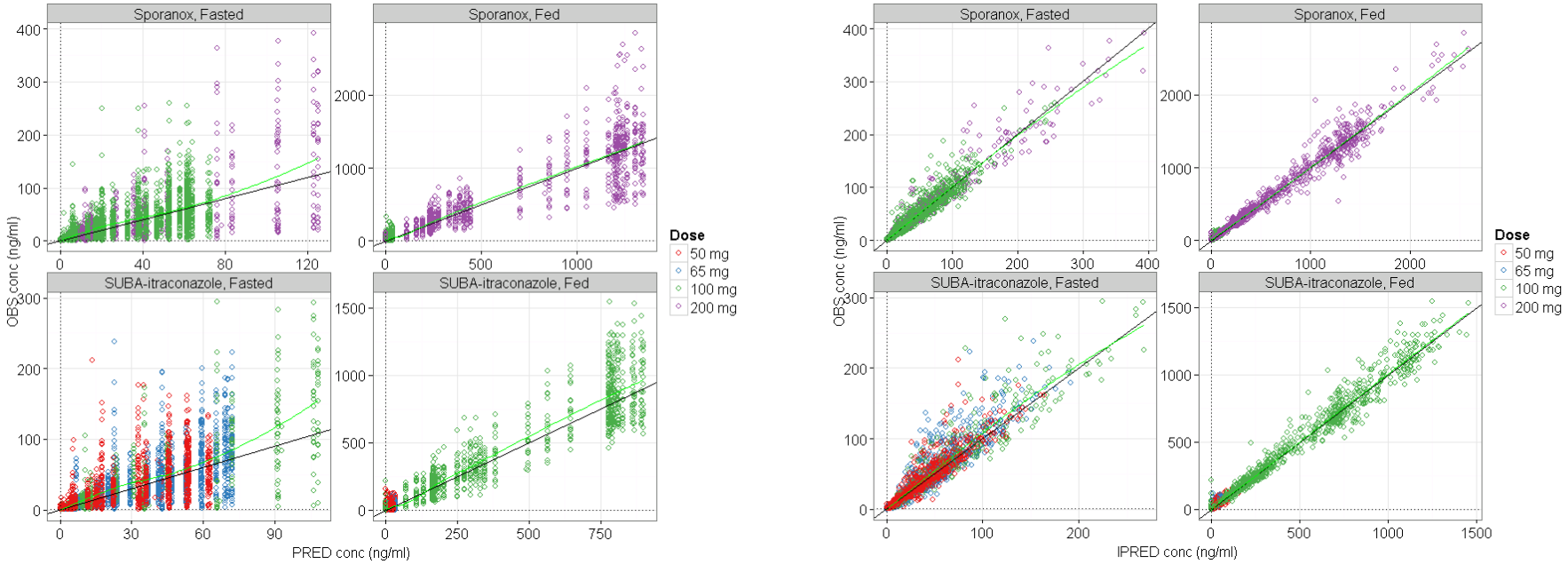

Fig S2. Goodness-of-fit plots of the final itraconazole model for single- and multi-dose data conditioned on formulation and fed status. In each plot, symbols are data points, the solid black line is a line of identity with slope 1 or 0, and the solid green line is a Loess smooth of the data. Left hand graph: Observed concentration (OBS) versus population predicted concentrations (PRED). Data are evenly distributed about the line of identify, indicating no major bias in the population component of the model. Right hand graph: Observed concentrations versus individual predicted concentration (IPRED). Data are evenly distributed about the line of identify, indicating an appropriate structural model could be found for most individuals. Figure legends defining itraconazole data points for the different itraconazole doses used in the pharmacokinetic studies coloured in red (50 mg), blue (65 mg), green (100mg), and purple (200 mg).

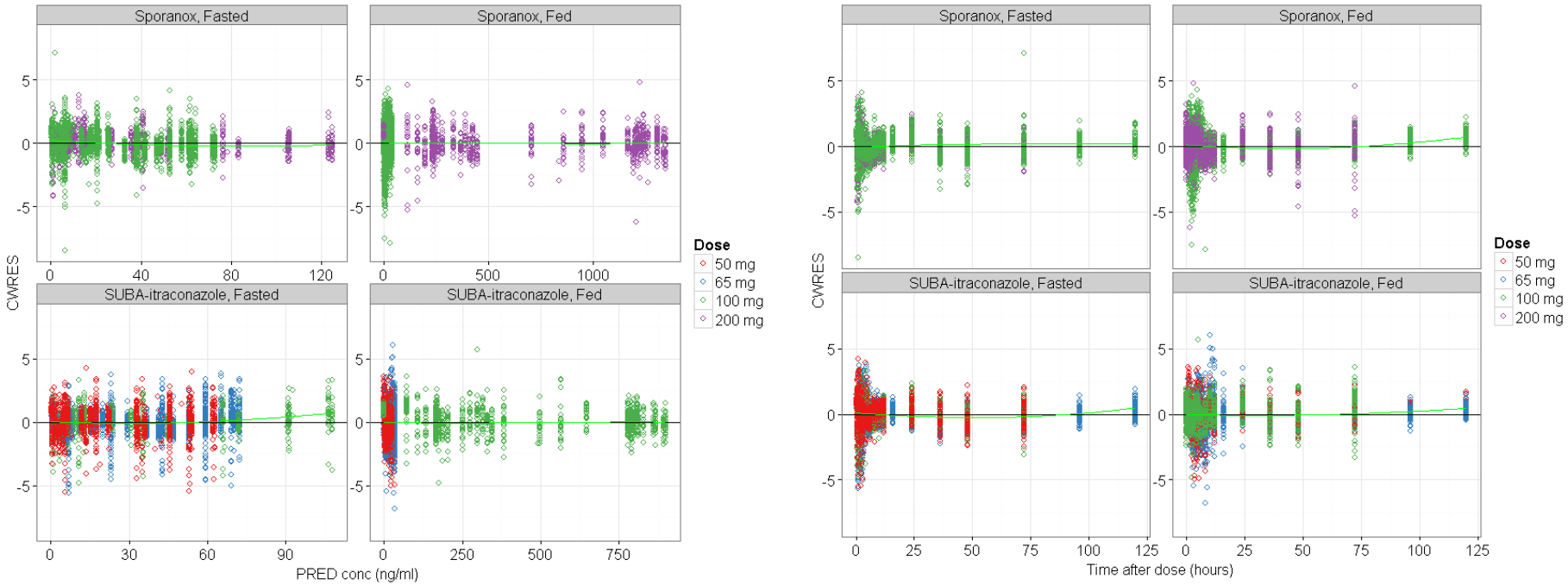

Fig S3. Goodness-of-fit plots of the final itraconazole model for single- and multi-dose data conditioned on formulation and fed status. In each plot, symbols are data points, the solid black line is a line of identity with slope 1 or 0, and the solid green line is a Loess smooth of the data. Left hand graph: conditional weighted residuals (CWRES) versus population predicted concentrations (PRED). Data are evenly distributed about zero, indicating no major bias in the structural model. Right hand graph: CWRES versus time after dose. Data are evenly distributed about zero, indicating no major bias in the residual error model. Figure legends defining itraconazole data points for the different itraconazole doses used in the pharmacokinetic studies coloured in red (50 mg), blue (65 mg), green (100mg), and purple (200 mg).

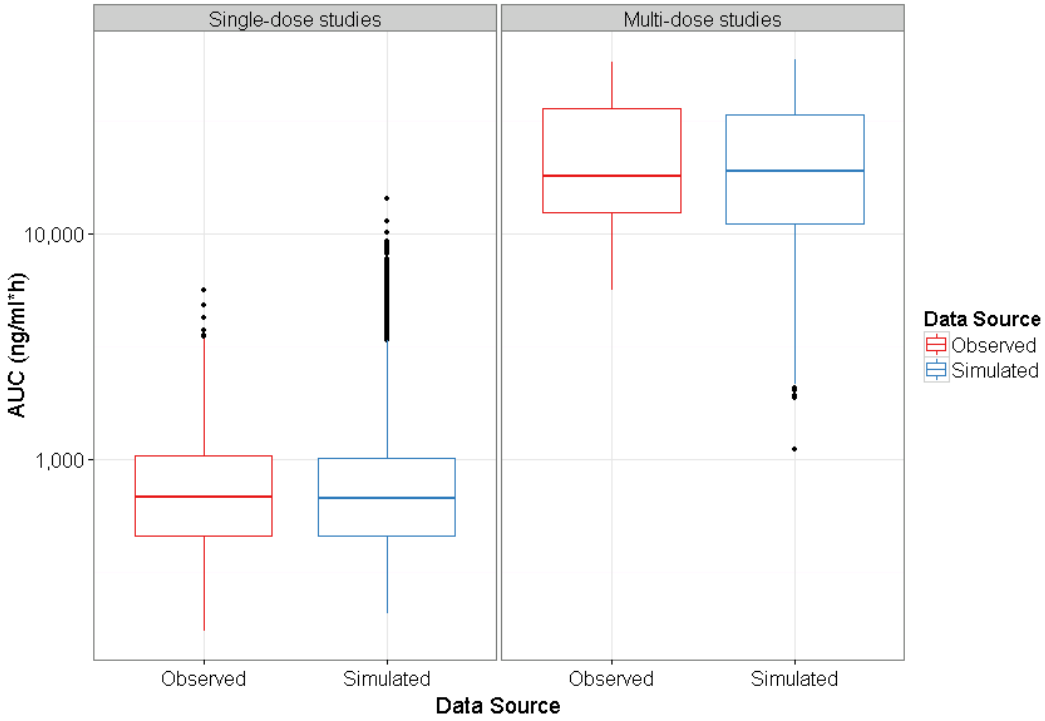

Fig S4. Visual predictive checks (VPC) of itraconazole exposure by area under the concentration-time curve (AUC_{0-t}) for single and multi-dose studies. Itraconazole AUC was calculated from zero to the last measurable concentration of the observed data and from 200 datasets simulations using the parameter values of the final itraconazole pharmacokinetic model. The bottom and top of each box represent the 25th and 75th percentiles and the line in the middle is the median. The whiskers represent ≤ 1.5 times the interquartile range. The dotted points represent outliers outside the whiskers.

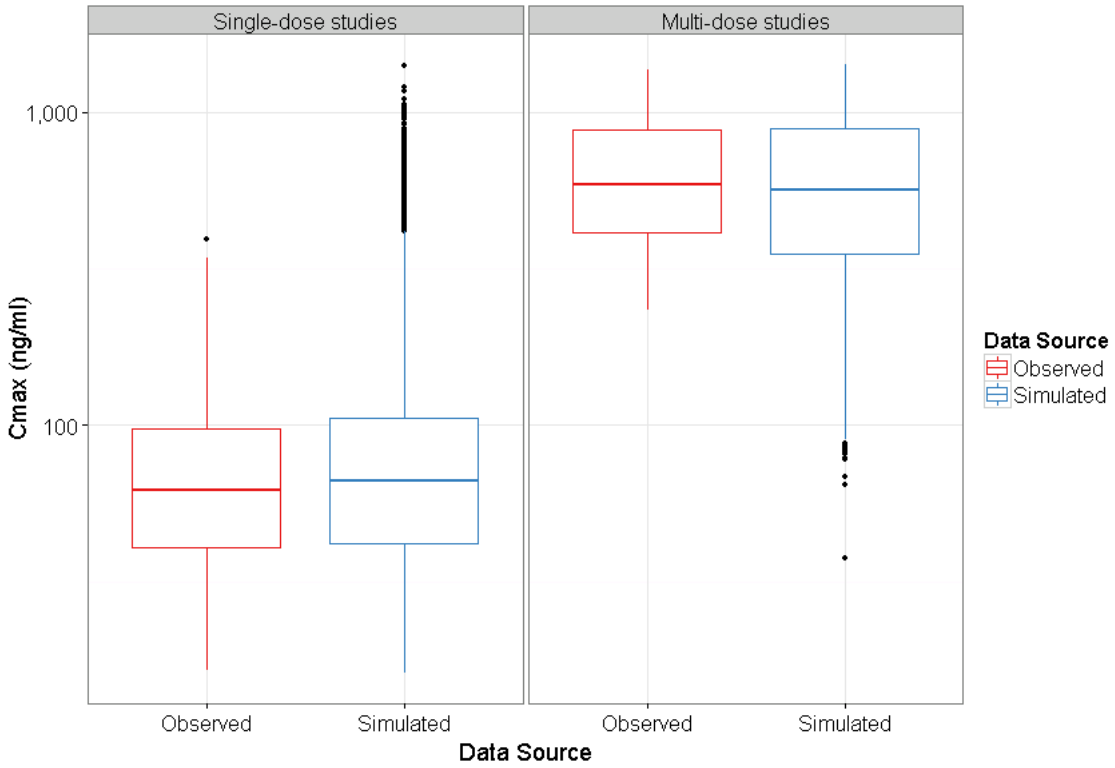

Fig S5. Visual predictive checks (VPC) of itraconazole exposure by maximum concentration (C_{max}) for single and multi-dose studies. Itraconazole C_{max} was calculated for the observed data and from 200 datasets simulations using the parameter values of the final itraconazole pharmacokinetic model. The bottom and top of each box represent the 25th and 75th percentiles and the line in the middle is the median. The whiskers represent ≤ 1.5 times the interquartile range. The dotted points represent outliers outside the whiskers.

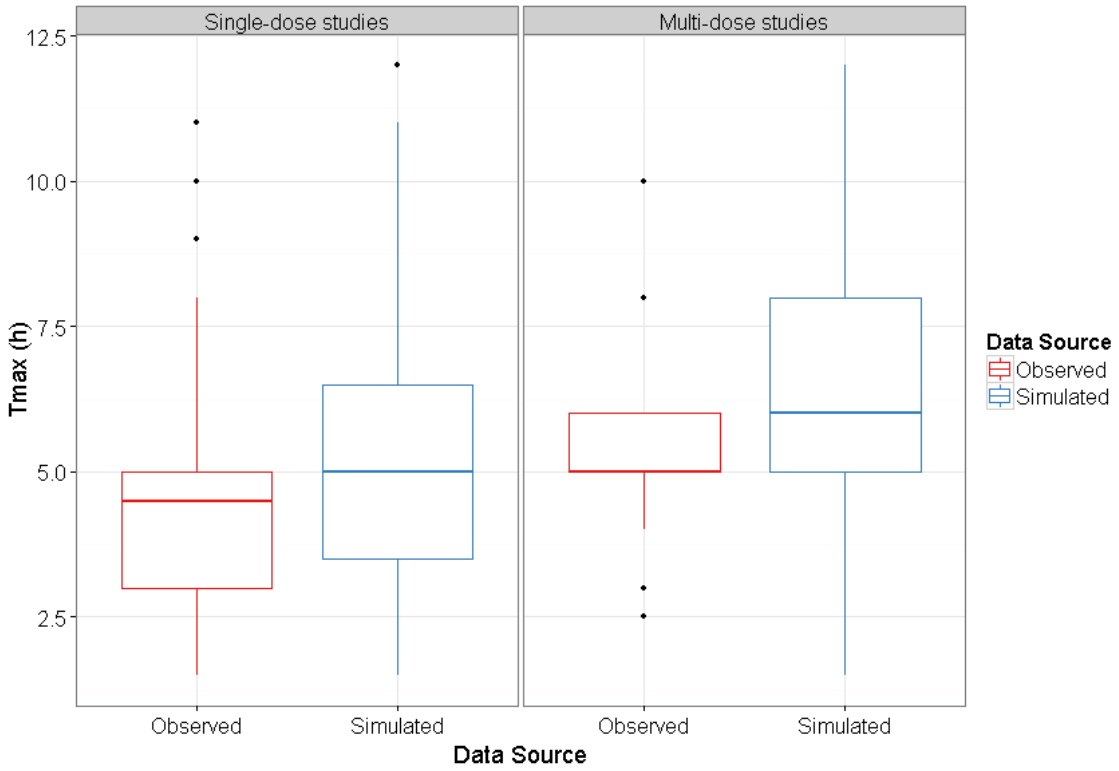

Fig S6. Visual predictive checks (VPC) of itraconazole exposure by time at maximum concentration (T_{max}) for single and multi-dose studies. Itraconazole T_{max} was calculated for the observed data and from 200 datasets simulations using the parameter values of the final itraconazole pharmacokinetic model. The bottom and top of each box represent the 25th and 75th percentiles and the line in the middle is the median. The whiskers represent ≤ 1.5 times the interquartile range. The dotted points represent outliers outside the whiskers.

Appendix S1: NONMEM control stream for the final itraconazole model

```
$PROBLEM - PK MODELLING OF ITRACONAZOLE DATA FOR SINGLE & MULTIPLE DOSE STUDIES
;UNITS ARE UG, L (NG/ML) AND H
;NM7.3

$INPUT CID ID STUDY TRT DRUG DOSE TAFD=TIME TAD TNOM DV DVID BLQ MDV AMT EVID CMT ADDL II PERIOD OCCI
   OCCS SEQ DAY TYPE SEX AGE ETHNIC RACE TOBACCO HEIGHT WEIGHT BMI FED SPORORG FFM

$DATA …\combdataEVID1.csv IGNORE=C IGNORE=(DVID.GE.2)

$SUBROUTINES  ADVAN5 TRANS1

$MODEL NCOMPARTMENTS=7 NPARAMETERS=8
COMP=(DEPOT,DEFDOSE,INITIALOFF)
COMP=(CENTRAL,DEFOBS)
COMP=(PERIPH)
COMP=(TRANSIT1)
COMP=(TRANSIT2)
COMP=(TRANSIT3)
COMP=(TRANSIT4)

$ABBR REPLACE THETA(CL)=THETA(1)
$ABBR REPLACE THETA(V2)=THETA(2)
$ABBR REPLACE THETA(Q)=THETA(3)
$ABBR REPLACE THETA(V3)=THETA(4)
$ABBR REPLACE THETA(KTR)=THETA(5)
$ABBR REPLACE ETA(CL)=ETA(1)
$ABBR REPLACE ETA(V2)=ETA(2)
$ABBR REPLACE ETA(KTR)=ETA(3)
$ABBR REPLACE ETA(FVAR)=ETA(4)

$PK
   ;ADDING THE EFFECT OF FORMULATION ON BA (F)
    IF (DRUG.EQ.0) THEN  ;SPORANOX
       DRUGF=1
    ELSE
       DRUGF=(1+THETA(8))  ;SUBACAP
    ENDIF
```

```
;VARIABILITY IN F - REFERENCE F IS SPORANOX
  IF (DRUG.EQ.0) THEN  ;SPORANOX
     ETASCALE=1
  ELSE
     ETASCALE=(1+THETA(11))  ;SUBACAP
  ENDIF

;REFERENCE IS FASTED
  IF(FED.EQ.1) THEN
    FEDF=(1+THETA(12))
    FEDKTR=(1+THETA(13))
  ELSE
    FEDF=1
    FEDKTR=1
  ENDIF

;TOTAL DAILY DOSE AND TIME_DEPENDENT CHANGE IN CLEARNCE
  IF(STUDY.EQ.706) THEN
     DDOSE=DOSE*2
  ELSE
     DDOSE=DOSE
  ENDIF

;EFFECTIVE DAILY DOSE: BIOAVAILABILITY CORRECTION FOR DAILY DOSE
  IF(DRUG.EQ.0) THEN    ; SPORANOX
     DDOSEF=DDOSE/(1+THETA(8))
     DDK=THETA(16)
  ELSE
     DDOSEF=DDOSE     ;SUBACAP
     DDK=THETA(17)
  ENDIF

;RATE CONSTANT FOR RISE TO STEADY STATE
  KSS=0.5  ; NEAR SS BY 7 DAYS AS PER LEHR

;DAILY DOSE ON CL  - EXPONENTIAL RELATIONSHIP
  DDMAX=THETA(18)

  IF (STUDY.GE.705) THEN
     DDCLSS=(1-DDMAX)*EXP(-1*DDK*DDOSEF)+DDMAX
```

```
   ELSE
      DDCLSS=1
   ENDIF
   DDCL=((1-DDCLSS)*EXP(-1*KSS*(DAY-1)))+DDCLSS  ;RISE TO STEADY STATE

 ;DAILY DOSE ON F - DDSLPF IS 1 FOR REFERENCE DOSE
   DDINTF=THETA(19)
   IF (STUDY.GE.705) THEN
      DDFSS=DDINTF
   ELSE
      DDFSS=1
   ENDIF
  DDF=((1-DDFSS)*EXP(-1*KSS*(DAY-1)))+DDFSS  ;RISE TO STEADY STATE
;STUDY ORIGIN COVARIATE
   IF(SPORORG.EQ.1)THEN ; US STUDIES
      STUDYORGKTR=1
      STUDYORGCL=1
   ELSE
      STUDYORGKTR=1 + THETA(20)          ; UK STUDIES
      STUDYORGCL=1 + THETA(21)
   ENDIF

   TVF1=1*DRUGF*FEDF*DDF
   FVAR=EXP(ETA(FVAR)*ETASCALE)
   F1=TVF1*FVAR

   CL=THETA(CL)*EXP(ETA(CL))*DDCL*STUDYORGCL
   V2=THETA(V2)*EXP(ETA(V2))
   Q=THETA(Q)
   V3=THETA(V3)

   KTR=THETA(KTR)*EXP(ETA(KTR))*FEDKTR*STUDYORGKTR

   K20=CL/V2
   K23=Q/V2
   K32=K23*V2/V3

   K14=KTR
   K45=KTR
   K56=KTR
   K67=KTR
```

```
    K72=KTR

    S2=V2

  $ERROR
    A1=A(1)
    A2=A(2)
    A3=A(3)
    A4=A(4)
    A5=A(5)
    A6=A(6)
    A7=A(7)

 ;THIS CODE WAS USED TO KEEP THE MODEL PREDICTIONS ABOVE 1/10 OF THE LLOQ
   IF(F.LE.0.05) THEN
     IPRE=LOG(0.05)
   ELSE
     IPRE=LOG(F)
   ENDIF

   IF(STUDY.LE.703.AND.FED.EQ.0)  W=SQRT(THETA(6)**2+THETA(7)**2/EXP(IPRE)**2)  ;SINGLE-DOSE FASTED
   IF(STUDY.LE.703.AND.FED.EQ.1)  W=SQRT(THETA(9)**2+THETA(10)**2/EXP(IPRE)**2) ;SINGLE-DOSE FED
   IF(STUDY.GT.703)               W=SQRT(THETA(14)**2+THETA(15)**2/EXP(IPRE)**2) ;MULTI-DOSE STUDIES

   IRES=DV-IPRE
   IWRES=IRES/(W)
   Y=IPRE + W*EPS(1)

   SIM=IREP   ;FOR COUNTING THE NUMBER OF SUBPROBLEMS

;-----------------PARAMETERS---------------------------------

  $THETA
    (0.01,123.,)  ; POPCL
    (0.01,868.,)  ; POPV2
    (0.01,154.,)  ; POPQ
    (0.01,2330.,) ; POPV3
    (0,2.05,)     ; POPKTR

    (0.001,0.292,)   ;RUVCVSFAST
    (0.001,0.169,)   ;RUVADDSFED
```

```
  (-0.99,0.735,)   ;POPDRUGF
  (0.001,0.399,)   ;RUVCVFED
  (0.001,0.14,)    ;RUVADDFED

  (-0.99,-0.215,)  ;ETASCALE
  (-0.99,-0.248,)  ;FEDF
  (-0.99,-0.583,)  ;FEDKTR
  (0.001,0.14,)    ;RUVCVM
  (0.001,0.267,)   ;RUVADDM

  (0,0.00624,)     ;DDKSPOR
  (0,0.00564,)     ;DDKSUB
   0 FIX           ;DDMAX
  (0,3.14,)        ;DDINTF
  (-0.99,0.326,)   ;STUDYORGKTR
  (-0.99,0.001,)   ;STUDYORGCL

$OMEGA
 0.0558 ; PPVCL
 0.0983 ; PPVV2
 0.202  ; PPVKTR
 0.312  ; PPVFVAR

$SIGMA
  1. FIX

;$SIMULATION (1234567)  ONLYSIM SUBPROBLEMS=1

$ESTIMATION METHOD=1 INTERACTION MAXEVALS=9999 POSTHOC NOABORT NSIG=3 SIGL=9 PRINT=5
$COVARIANCE  UNCONDITIONAL SIGL=12 PRINT=E

$TABLE CID ID STUDY TRT DRUG DOSE DDOSE PERIOD OCCI OCCS SEQ DAY TAFD TNOM AMT DV MDV EVID
DVID SEX FED ETHNIC RACE TOBACCO AGE HEIGHT WEIGHT BMI SPORORG BLQ IPRE IRES IWRES
CWRES NOPRINT ONEHEADER FORMAT S1PE15.7 FILE=7_27_id2_2comp_effectivedd_studyorg_ktr_cl.fit

;FOR XPOSE
$TABLE CID ID STUDY TRT DRUG DOSE DDOSE PERIOD BLQ OCCI OCCS SEQ DAY TAFD TAD TNOM MDV EVID AMT DV
DVID IPRE IRES IWRES CWRES NOPRINT ONEHEADER FORMAT S1PE15.7 FILE=SDTAB1
$TABLE CID ID SEX ETHNIC RACE FED
NOPRINT NOAPPEND ONEHEADER FORMAT S1PE15.7 FILE=CATAB1
$TABLE CID ID AGE HEIGHT WEIGHT BMI TOBACCO SPORORG FFM
```

```
NOPRINT NOAPPEND ONEHEADER FORMAT S1PE15.7 FILE=COTAB1
$TABLE CID ID CL V2 Q V3 KTR ETA1 ETA2 ETA3 ETA4
NOPRINT NOAPPEND ONEHEADER FORMAT S1PE15.7 FILE=PATAB1
```

Appendix S2: NONMEM control stream for the final metabolite model

```
$PROBLEM - PK MODELLING OF ITRACONAZOLE AND HYDROXY-ITRACONAZOLE DATA FOR SINGLE & MULTIPLE DOSE STUDIES
;UNITS ARE UG, L (NG/ML) AND H
;NM7.3

$INPUT CID ID STUDY TRT DRUG DOSE TAFD=TIME TAD TNOM DV DVID BLQ MDV AMT EVID CMT ADDL II PERIOD OCCI
   OCCS SEQ DAY TYPE SEX AGE ETHNIC RACE TOBACCO HEIGHT WEIGHT BMI FED SPORORG FFM

$DATA …\combdataEVID1.csv IGNORE=C

$SUBROUTINES  ADVAN6 TOL=6

$MODEL
COMP=(DEPOT,DEFDOSE,INITIALOFF)
COMP=(CENTRAL,DEFOBS)   ; CENTRAL COMPARTMENT-PARENT
COMP=(PERIPH)
COMP=(TRANSIT1)
COMP=(TRANSIT2)
COMP=(TRANSIT3)
COMP=(TRANSIT4)
COMP=(METAB)            ; CENTRAL COMPARTMENT-METABOLITE

$ABBR REPLACE THETA(CLP)=THETA(1)
$ABBR REPLACE THETA(V2P)=THETA(2)
$ABBR REPLACE THETA(QP)=THETA(3)
$ABBR REPLACE THETA(V3P)=THETA(4)
$ABBR REPLACE THETA(KTR)=THETA(5)

$ABBR REPLACE THETA(CLM)=THETA(22)
$ABBR REPLACE THETA(V1M)=THETA(23)

$ABBR REPLACE ETA(CLP)=ETA(1)
$ABBR REPLACE ETA(V2P)=ETA(2)
$ABBR REPLACE ETA(KTR)=ETA(3)
$ABBR REPLACE ETA(FVAR)=ETA(4)
$ABBR REPLACE ETA(CLM)=ETA(5)
```

```
$PK
 ;ADDING THE EFFECT OF FORMULATION ON BA (F)
    IF (DRUG.EQ.0) THEN  ;SPORANOX
       DRUGF=1
    ELSE
       DRUGF=(1+THETA(8))  ;SUBACAP
    ENDIF

 ;VARIABILITY IN F - REFERENCE F IS SPORANOX
    IF (DRUG.EQ.0) THEN  ;SPORANOX
       ETASCALE=1
    ELSE
       ETASCALE=(1+THETA(11))  ;SUBACAP
    ENDIF
 ;REFERENCE IS FASTED
    IF(FED.EQ.1) THEN
       FEDF=(1+THETA(12))
       FEDKTR=(1+THETA(13))
    ELSE
       FEDF=1
       FEDKTR=1
    ENDIF

 ;TOTAL DAILY DOSE AND TIME_DEPENDENT CHANGE IN CLEARNCE
    IF(STUDY.EQ.706) THEN
      DDOSE=DOSE*2
    ELSE
      DDOSE=DOSE
    ENDIF

; EFFECTIVE DAILY DOSE: BIOAVAILABILITY CORRECTION FOR DAILY DOSE
    IF(DRUG.EQ.0) THEN    ; SPORANOX
       DDOSEF=DDOSE/(1+THETA(8))
       DDK=THETA(16)
    ELSE
       DDOSEF=DDOSE     ;SUBACAP
       DDK=THETA(17)
    ENDIF

 ;RATE CONSTANT FOR RISE TO STEADY STATE
    KSS=0.5  ; NEAR SS BY 7 DAYS AS PER LEHR
```

```
;DAILY DOSE ON CL  - EXPONENTIAL RELATIONSHIP
   DDMAX=THETA(18)

   IF (STUDY.GE.705) THEN
      DDCLSS=(1-DDMAX)*EXP(-1*DDK*DDOSEF)+DDMAX
   ELSE
      DDCLSS=1
   ENDIF
   DDCL=((1-DDCLSS)*EXP(-1*KSS*(DAY-1)))+DDCLSS  ;RISE TO STEADY STATE

;DAILY DOSE ON F - DDSLPF IS 1 FOR REFERENCE DOSE
   DDINTF=THETA(19)
   IF (STUDY.GE.705) THEN
      DDFSS=DDINTF
   ELSE
      DDFSS=1
   ENDIF
  DDF=((1-DDFSS)*EXP(-1*KSS*(DAY-1)))+DDFSS  ;RISE TO STEADY STATE
;STUDY ORIGIN COVARIATE
   IF(SPORORG.EQ.1)THEN ; US STUDIES
      STUDYORGKTR=1
      STUDYORGCL=1
   ELSE
      STUDYORGKTR=1 + THETA(20)          ; UK STUDIES
      STUDYORGCL=1 + THETA(21)
   ENDIF
;TIME DEPENDENT METABOLITE CLEARANCE
   IF (STUDY.LT.705) TIMECLM=1
   IF (STUDY.EQ.705) TIMECLM=(1+THETA(24))
   IF (STUDY.EQ.706) TIMECLM=(1+THETA(25))

   TVF1=1*DRUGF*FEDF*DDF
   FVAR=EXP(ETA(FVAR)*ETASCALE)
   F1=TVF1*FVAR

   CLP=THETA(CLP)*EXP(ETA(CLP))*DDCL*STUDYORGCL

   V2P=THETA(V2P)*EXP(ETA(V2P))
   QP=THETA(QP)
   V3P=THETA(V3P)
```

```
   VMAX=THETA(30)                  ; MAXIMUM RATE OF NON-LINEAR ELIMINATION FOR THE METABOLITE
   KM=THETA(31)                    ; CONCENTRATION AT 50% VMAX

   CLM=THETA(CLM)*EXP(ETA(CLM))*TIMECLM
   V1M=THETA(V1M)                    ; VD OF THE METABOLITE

   KTR=THETA(KTR)*EXP(ETA(KTR))*FEDKTR*STUDYORGCL

   K28=CLP/V2P
   K23=QP/V2P
   K32=K23*V2P/V3P
   K80=CLM/V1M

   S2=V2P
   S8=V1M

$DES
   C8=A(8)/V1M   ; METABOLITE COMPARTMENT CONCENTRATION
;ABSORPTION COMPARTMENT
   DADT(1)=-KTR*A(1)
;SYSTEMIC MODEL
   DADT(2)=KTR*A(7)+K32*A(3)- K23*A(2)-K28*A(2)
   DADT(3)=K23*A(2)-K32*A(3)
   DADT(8)=K28*A(2)-K80*A(8)
;TRANSIT COMPARTMENTS
   DADT(4)=KTR*A(1)-KTR*A(4)
   DADT(5)=KTR*A(4)-KTR*A(5)
   DADT(6)=KTR*A(5)-KTR*A(6)
   DADT(7)=KTR*A(6)-KTR*A(7)

   IF(STUDY.LE.703)THEN
      DADT(8)=K28*A(2)-K80*A(8)-((VMAX*C8)/(KM+C8))
   ENDIF

$ERROR
   A1=A(1)
   A2=A(2)
   A3=A(3)
   A4=A(4)
   A5=A(5)
```

```
   A6=A(6)
   A7=A(7)
   A8=A(8)

 ; THIS CODE WAS USED TO KEEP THE MODEL PREDICTIONS ABOVE 1/10 OF THE LLOQ (PAGE35)
   IF(F.LE.0.05) THEN
      IPRE=LOG(0.05)
   ELSE
      IPRE=LOG(F)
   ENDIF

 IF(DVID.EQ.1.AND.STUDY.LE.703.AND.FED.EQ.0)  W=SQRT(THETA(6)**2+THETA(7)**2/EXP(IPRE)**2) ;SINGLE-FASTED
 IF(DVID.EQ.1.AND.STUDY.LE.703.AND.FED.EQ.1)  W=SQRT(THETA(9)**2+THETA(10)**2/EXP(IPRE)**2) ;SINGLE-FED
 IF(DVID.EQ.1.AND.STUDY.GT.703)          W=SQRT(THETA(14)**2+THETA(15)**2/EXP(IPRE)**2) ;MULTI STUDIES
 IF(DVID.EQ.2.AND.STUDY.LE.703)    W=SQRT(THETA(26)**2+THETA(27)**2/EXP(IPRE)**2) ;SINGLE-DOSE-METABOLITE
 IF(DVID.EQ.2.AND.STUDY.GT.703)    W=SQRT(THETA(28)**2+THETA(29)**2/EXP(IPRE)**2) ;MULTI-DOSE METABOLITE

   IRES=DV-IPRE
   IWRES=IRES/(W)
   Y=IPRE + W*EPS(1)

    SIM=IREP   ;FOR COUNTING THE NUMBER OF SUBPROBLEMS

;-----------------PARAMETERS---------------------------------
$THETA
   (0.01,129.,)  FIX ; POPCL
   (0.01,861.,)  FIX ; POPV2
   (0.01,153.,)  FIX ; POPQ
   (0.01,2340.,) FIX ; POPV3
   (0,2.05,)     FIX ; POPKTR

   (0.001,0.293,)  FIX ;RUVCVSFAST
   (0.001,0.168,)  FIX ;RUVADDSFED
   (-0.99,0.729,)  FIX ;POPDRUGF
   (0.001,0.399,)  FIX ;RUVCVFED
   (0.001,0.14,)   FIX ;RUVADDFED

   (-0.99,-0.213,) FIX ; ETASCALE
   (-0.99,-0.269,) FIX ; FEDF
   (-0.99,-0.583,) FIX ; FEDKTR
   (0.001,0.141,)  FIX ; RUVCVM
```

```
   (0.001,0.267,)  FIX ; RUVADDM

   (0,0.0065,)    FIX  ; DDKSPOR
   (0,0.00596,)   FIX  ; DDKSUB
      0 FIX            ; DDMAX
   (0,3.24,)       FIX  ; DDINTF
   (-0.99,0.338,)  FIX ; STUDYORGKTR
   (-0.99,-0.166,) FIX ; STUDYORGCL

   (0.01,45,)  ; POPCLM
   (2,8,)      ; POPV1M
   (-0.99,-0.32,) ; TIMECLM705
   (-0.99,-0.55,) ; TIMECLM706

   (0.001, 0.25,) ; RESMSIN
   (0.001, 0.22,) ; RESADDMSIN
   (0.001, 0.15,) ; RESMMULTI
   (0.001, 0.2,)  ; RESADDMMULTI

   (0.0001, 390.,) ; POPVMAX
   (0.0001, 1.45,) ; POPKM

$OMEGA
   0.0489 FIX ; PPVCL
   0.0944 FIX; PPVV2
   0.201 FIX; PPVKTR
   0.318 FIX; PPVFVAR
   0.1 ; PPVCLM

$SIGMA
   1. FIX

;$SIMULATION (1234567)  ONLYSIM SUBPROBLEMS=200

 $ESTIMATION METHOD=1 INTERACTION MAXEVALS=9999 POSTHOC NOABORT NSIG=3 SIGL=9 PRINT=5
 $EST METHOD=IMP  INTERACTION EONLY=1 NITER=10 ISAMPLE=300 PRINT=1 SIGL=8 NOPRIOR=1
 $COVARIANCE  UNCONDITIONAL SIGL=12 PRINT=E

 $TABLE CID ID STUDY TRT DRUG DOSE DDOSE PERIOD OCCI OCCS SEQ DAY TAFD TNOM AMT DV MDV EVID
 DVID SEX FED ETHNIC RACE TOBACCO AGE HEIGHT WEIGHT BMI FFM SPORORG BLQ IPRE IRES IWRES CWRES NOPRINT
 ONEHEADER FORMAT S1PE15.7 FILE=15_1_base_mixed_firstorder_mm_timeclm56_noscaling_ifsingle.fit
```

```
;FOR XPOSE
 $TABLE CID ID STUDY TRT DRUG DOSE DDOSE PERIOD BLQ OCCI OCCS SEQ DAY TAFD TAD TNOM MDV EVID AMT DV
 DVID IPRE IRES IWRES CWRES NOPRINT ONEHEADER FORMAT S1PE15.7 FILE=SDTAB1
 $TABLE CID  ID SEX ETHNIC RACE FED
 NOPRINT NOAPPEND ONEHEADER FORMAT S1PE15.7 FILE=CATAB1
 $TABLE CID  ID AGE HEIGHT WEIGHT BMI TOBACCO SPORORG FFM
 NOPRINT NOAPPEND ONEHEADER FORMAT S1PE15.7 FILE=COTAB1
 $TABLE CID ID CLM V1M CLP V2P QP V3P KTR ETA1 ETA2 ETA3 ETA4 ETA5
 NOPRINT NOAPPEND ONEHEADER FORMAT S1PE15.7 FILE=PATAB1
```

Appendix S3: R codes for itraconazole Shiny app

To run itraconazole Shiny app

1. Copy and save the **server.R**, **ui.R**, and **ADVAN_2compOral_4Transit_Function.R** separately into the same folder directory.
2. Open **server.R** and **ui.R** files in the latest version of RStudio
3. Click "RunApp" in the top right corner of the source pane

N.B. You need to have the following packages installed in R Studio before running the app. If not already installed, you may run the following code to install them:

```
install.packages(c("shiny", " ggplot2","plyr", "reshape2","grid", "scales","doParallel"))
```

ui.R

```
library(shiny)

shinyUI(fluidPage(

  #Application Title
  fluidRow(
    column(12,
           h2("Population Pharmacokinetic Modelling of Itraconazole for Oral SUBAÂ®-itraconazole and SporanoxÂ® Capsule Formulations in Healthy Subjects in Fed and Fasted States", align = "center"))
    ),   #Closing fluidrow

  hr(),  #Inserting line break and horizontal line

  #Choosing overall page layout
  sidebarLayout(
```

```
#Sidebar panel with widgets that demonstrate available options
sidebarPanel(
  #Create the action button to update plot
  actionButton("calculatePK", label = "Start Simulation"),
  p("Press this button after selecting the desired settings to update the plot"),
  p(strong("Select Fed status and dosing design:")),

  radioButtons("FED_STATUS", label = "Fed status:",
                    choices = list("Fasted" = 1, "Fed" = 2),
                    selected = 1),

  #Creates a dropdown menu for selecting the formulation
  selectInput("SELECT", "Formulation:",
              c("SporanoxÂ®"=1, "SUBAÂ®-itraconazole"=2)
  ),   #This closes selectInput widget

  #Creates a dropdown menu for dosing regimen
  selectInput("FREQ", "Dosing frequency:",
              c("Once daily"=1, "Twice daily"=2),
              selected = 1),   #This closes selectInput widget

  #Create a conditional panel for each dosing scenario
  conditionalPanel(condition = "input.SELECT == 1",

                   #Slider input for dose
                   sliderInput("SPO_DOSE", "SporanoxÂ® dose (mg):",
                               min = 10, max = 600, value = 100, step = 5)
  ),   #Close conditional panel

  #Create a conditional panel for each dosing scenario
  conditionalPanel(condition = "input.SELECT == 1",

                   #Slider input for dose
                   sliderInput("SPO_DAYS", "SporanoxÂ® treatment duration (days):",
                               min = 1, max = 28, value = 1, step = 1)
  ),   #Close conditional panel


conditionalPanel(condition = "input.SELECT == 2",

                   #Numeric input for dose
```

```
                  sliderInput("LOZ_DOSE", "SUBA-itraconazole dose (mg):",
                              min = 10, max = 600, value = 50, step = 5)

  ),  #Close conditional panel

  conditionalPanel(condition = "input.SELECT == 2",

                  #Numeric input for dose
                  sliderInput("LOZ_DAYS", "SUBA-itraconazole treatment duration (days):",
                              min = 1, max = 28, value = 1, step = 1)

  ),  #Close conditional panel


  #Create a download button to download dataset
  downloadButton("downloadsimdata",
                 "Download Simulated Data"),

  align = "left"),   #Closing sidebarpanel and aligning left side of page

#starting the main panel
mainPanel(

  #Create panel with tabs
  tabsetPanel(

    #First tab
    tabPanel("Itraconazole",
             br(),

             fluidRow(

               column(5, checkboxInput("CIITRACONAZOLE", "Show 90% confidence interval",
                           value = FALSE))

             ),    #close fluidrow

             plotOutput("plotCONC", height = 550, width = 800)


    ), #Closing first tab
```

```
        #Second tab
        tabPanel("About",
                 br(),
                p("This", a("Shiny",
                            href = "http://www.rstudio.com/shiny"),
                  "application is based on a population pharmacokinetic model of itraconazole developed
using",strong("single- and multi-dose"), "pharmacokinetic data of seven Phase I clinical trials comparing
SUBA-itraconazole with SporanoxÂ®."),
                 p("The final pharmacokinetic model of itraconazole was a 2-compartment model with oral
absorption described by 4-transit compartments",em("(See Figure below).")),
                 p("Simulated concentrations in this application are based on 200 simulated subjects with
the selected dosing design."),
                 p("SporanoxÂ® doses below 100mg and above 400mg and SUBAÂ®-itraconazole doses below 50mg
and above 200mg were not used to develop the model, and therefore the simulations are extrapolations."),
                 p(strong("Important!"),"Click the", strong(em("Start Simulation")),  "button to update
the plot every time you change the dosing regemin or the slider values."),
                 br(),
                 img(src = "FinalPKmodel.png"),
                 p(strong("Fig.1 A schematic diagram for the final PopPK model of itraconazole")),
                 br(),
                 br(),
                  p(strong("Developer:")),
                  p("Ahmad Abuhelwa"),
                  p("University of South Australia"),
                  a("ahmad.abuhelwa@mymail.unisa.edu.au",href                                                    =
"mailto:ahmad.abuhelwa@mymail.unisa.edu.au")


        ) #Closing About tab

      ) #Closing tabsetpanel

    )  #Closing main panel

  )   #Closing sidebarlayout

))  #Closing fluidpage and shiny UI
```

server.R

```
#Load package libraries
library(shiny)        #Creating the app
library(ggplot2)      #Plotting
library(plyr)         #Split and rearrange data, ddply function
library(reshape2)
library(scales)
library(grid)
#library(doParallel)

#ADVAN-style analytical solution for itraconazole model
source("ADVAN_2compOral_4Transit_Function.R")
#------------------------------------------------------------------------------
#Customize ggplot2 theme - R 2.15.1
theme_bw2 <- theme_set(theme_bw(20))
theme_bw2 <- theme_update(plot.margin = unit(c(1.5,1.5,3,1.5), "lines"),
                          axis.title.x=element_text(size = 16, vjust = 0),
                          axis.title.y=element_text(size = 16, vjust = 1, angle = 90))

#-----------------------------------------------------------------
# This part of the code is used set up multiple cores to run the application over
#cl <- makePSOCKcluster(detectCores())
#clusterEvalQ(cl, list(library(foreach)))
#Registers the parallel backend with the foreach package (automatically loaded when
#doParallel is loaded)
#registerDoParallel(cl)
#-----------------------------------------------------------------
#Define population values of PK parameters in the model
CLpop  <- 129
V2pop  <- 861
Qpop   <- 153
V3pop  <- 2340
KTRpop <- 2.05
F1pop  <- 1

#Set number of individuals to simulate
nID  <- 200
ID   <- 1:nID

#Define Between subject variability
```

```
BSVCL      <- rnorm(nID, mean = 0, sd = 0.2213)
BSVV2      <- rnorm(nID, mean = 0, sd = 0.3075)
BSVKTR     <- rnorm(nID, mean = 0, sd = 0.4483)
FVAR       <- rnorm(nID, mean = 0, sd = 0.5639)

#Define residual error (Specified later in the reactive function!)
#Single-dose fasted
PROPRES_S_Fasted <- 0.168
ADDRES_S_Fasted  <- 0.293

#Single-dose FED
PROPRES_S_Fed    <- 0.399
ADDRES_S_Fed     <- 0.14

#Multiple dosing
PROPRES_M         <- 0.141
ADDRES_M          <- 0.267

#Define covariate parameters
DRUGF      <- 1.729  #For SUBA (Ref Sporanox=1)
ETASCALE   <- 0.787  #For SUBA (Ref Sporanox=1)
FEDF       <- 0.731   #For Fed (Ref Fasted=1)
FEDKTR     <- 0.417  #For Fed (Ref Fasted=1)
DDFSS      <- 3.26  #Scale factor for bioavailability (multidose)

KSS        <- 0.5    #Rate constant for rise to steady state
DDKSPO     <- 0.0065
DDKLOZ     <- 0.00596


shinyServer(function(input,output){

  SIM.data <- reactive({

    input$calculatePK  #Make the execution of the reactive function occur only after clicking Simulate
button.

  isolate({           #Isolate the rest of the reactive function, so it doesnt execute when all the sliders
are changed
    SELECT     <- input$SELECT
```

```
#The first Formulation: Sporanox
  if (SELECT=="1") {
    SPO_DOSE   <- input$SPO_DOSE
    SPO_DAYS   <- input$SPO_DAYS
    FREQ       <- input$FREQ
    FED_STATUS <- input$FED_STATUS
#set individual PK parameters
    df.para    <- data.frame(ID)
    df.para$CL <- CLpop*exp(BSVCL)  #need to add DDCL later based on the dosing regimen
    df.para$V2 <- V2pop*exp(BSVV2)
    df.para$Q  <- Qpop
    df.para$V3 <- V3pop
    df.para$KTR <- KTRpop*exp(BSVKTR)  #this is for FASTED
    df.para$F1 <- F1pop*exp(FVAR)      #this is for FASTED
  #KTR and F1 (fed versus fasted)
  if(FED_STATUS==2) {df.para$KTR <- KTRpop*exp(BSVKTR)*FEDKTR; df.para$F1 <- F1pop*exp(FVAR)*FEDF} #Fed

#Make a data frame
    #set dosing times
    if(FREQ==1) { dosetimes <- c(seq(0,(SPO_DAYS-1)*24,by=24)) }
    if(FREQ==2) { dosetimes <- c(seq(0,(SPO_DAYS-0.5)*24,by=12)) }

 #Now define finer sample times for after a dose to capture Cmax
   doseseq <- c(0,0.5,1,1.5,2,2.5,3,3.5,4,4.5,5,5.5,6,7,8,9,10)
 #Use the outer product but with addition to expand this doseseq for all dosetimes
   PKtimes <- outer(dosetimes,doseseq,"+")

 #Now define a background sequence with a wider time interval to fill the gaps between doses
   tlast        <- SPO_DAYS*24
   sampletimes  <- sort(unique(c(seq(0,tlast,1),PKtimes)))
   if(SPO_DAYS >= 5){ sampletimes  <- sort(unique(c(seq(0,tlast,2),PKtimes)))}
    df           <- expand.grid("ID"=ID,"TIME"=sampletimes,"AMT"=0)
    doserows     <- subset(df, TIME%in%dosetimes)
    doserows$AMT <- SPO_DOSE*1000 #to get conc in (ng/ml)
    #Add back dose information
    df <- rbind(df,doserows)
    df <- df[order(df$ID,df$TIME),]
    df <- subset(df, (TIME==0 & AMT==0)==F)        #remove the row that has a TIME=0 and AMT=0
#For the accumulation
    df$DAY    <- floor(df$TIME/24)+1
    df$DOSE   <- SPO_DOSE
```

```
      df$DDKSPO <- DDKSPO
      if(FREQ==1){df$DDOSEF <- df$DOSE/DRUGF}    #single dose
      if(FREQ==2){df$DDOSEF <- df$DOSE*2/DRUGF}  #twice daily dose
      df$DDCLSS <- exp(-1*df$DDKSPO*df$DDOSEF)
      df$DDCL   <- ((1-df$DDCLSS)*exp(-KSS*(df$DAY-1)))+df$DDCLSS
      df$DDF    <- ((1-DDFSS)*exp(-KSS*(df$DAY-1)))+DDFSS

      dfADVAN <- join(df,df.para,by="ID")

      #add DDCL and DDF on central clearance, bioavailability, respectively
      dfADVAN$CL <- dfADVAN$CL*dfADVAN$DDCL
      dfADVAN$F1 <- dfADVAN$F1*dfADVAN$DDF

      #Calculate micro-rate constants:
      dfADVAN$k20 <- dfADVAN$CL/dfADVAN$V2
      dfADVAN$k23 <- dfADVAN$Q/dfADVAN$V2
      dfADVAN$k32 <- dfADVAN$Q/dfADVAN$V3

      #Specify residual errors
      if(SPO_DAYS==1 & FREQ==1 & FED_STATUS==1) {PROPRES <- PROPRES_S_Fasted; ADDRES <- ADDRES_S_Fasted}
      if(SPO_DAYS==1 & FREQ==1 & FED_STATUS==2) {PROPRES <- PROPRES_S_Fed; ADDRES <- ADDRES_S_Fed}
      if(SPO_DAYS==1 & FREQ==2) {PROPRES <- PROPRES_M; ADDRES <- ADDRES_M}
      if(SPO_DAYS>=2) {PROPRES <- PROPRES_M; ADDRES <- ADDRES_M}

   } #end of select ==1

#start SELECT ==2 ==> SUBA-itraconazole
   if (SELECT=="2") {
      LOZ_DOSE   <- input$LOZ_DOSE
      LOZ_DAYS   <- input$LOZ_DAYS
      FREQ       <- input$FREQ
      FED_STATUS <- input$FED_STATUS
      #set individual PK parameters
      df.para     <- data.frame(ID)
      df.para$CL  <- CLpop*exp(BSVCL)  #need to add DDCL later based on the dosing regimen
      df.para$V2  <- V2pop*exp(BSVV2)
      df.para$Q   <- Qpop
      df.para$V3  <- V3pop
      df.para$KTR <- KTRpop*exp(BSVKTR)  #this is for FASTED
      df.para$F1  <- F1pop*DRUGF*exp(FVAR*ETASCALE)      #this is for FASTED
    #KTR and F1 (fed)
```

```
      if(FED_STATUS==2)      {df.para$KTR      <-      KTRpop*exp(BSVKTR)*FEDKTR;      df.para$F1      <-
F1pop*DRUGF*exp(FVAR*ETASCALE)*FEDF} #Fed
 #Make a data frame
      #set dosing times
      if(FREQ==1) { dosetimes <- c(seq(0,(LOZ_DAYS-1)*24,by=24)) }
      if(FREQ==2) { dosetimes <- c(seq(0,(LOZ_DAYS-0.5)*24,by=12)) }

   #Now define finer sample times for after a dose to capture Cmax
      doseseq <- c(0,0.5,1,1.5,2,2.5,3,3.5,4,4.5,5,5.5,6,7,8,9,10)
   #Use the outer product but with addition to expand this doseseq for all dosetimes
      PKtimes <- outer(dosetimes,doseseq,"+")

      #Now define a background sequence with a wider time interval to fill the gaps between doses
      tlast        <- LOZ_DAYS*24
      sampletimes  <- sort(unique(c(seq(0,tlast,1),PKtimes)))
      if(LOZ_DAYS >= 5){ sampletimes  <- sort(unique(c(seq(0,tlast,2),PKtimes)))}
      df           <- expand.grid("ID"=ID,"TIME"=sampletimes,"AMT"=0)
      doserows     <- subset(df, TIME%in%dosetimes)
      doserows$AMT <- LOZ_DOSE*1000    #to get conc in (ng/ml)
      #Add back dose information
      df <- rbind(df,doserows)
      df <- df[order(df$ID,df$TIME),]
      df <- subset(df, (TIME==0 & AMT==0)==F)
 #For the accumulation
      df$DAY    <- floor(df$TIME/24)+1
      df$DOSE   <- LOZ_DOSE
      df$DDKLOZ <- DDKLOZ
      if(FREQ==1){df$DDOSEF <- df$DOSE}    #single dose
      if(FREQ==2){df$DDOSEF <- df$DOSE*2}  #twice daily dose
      df$DDCLSS <- exp(-1*df$DDKLOZ*df$DDOSEF)
      df$DDCL   <- ((1-df$DDCLSS)*exp(-KSS*(df$DAY-1)))+df$DDCLSS
      df$DDF    <- ((1-DDFSS)*exp(-KSS*(df$DAY-1)))+DDFSS

      dfADVAN <- join(df,df.para,by="ID")

 #add DDCL and DDF on central clearance, bioavailability, respectively
      dfADVAN$CL <- dfADVAN$CL*dfADVAN$DDCL
      dfADVAN$F1 <- dfADVAN$F1*dfADVAN$DDF

      #Calculate micro-rate constants:
      dfADVAN$k20 <- dfADVAN$CL/dfADVAN$V2
```

```
      dfADVAN$k23 <- dfADVAN$Q/dfADVAN$V2
      dfADVAN$k32 <- dfADVAN$Q/dfADVAN$V3

      #Specify residual errors
      if(LOZ_DAYS==1 & FREQ==1 & FED_STATUS==1) {PROPRES <- PROPRES_S_Fasted; ADDRES <- ADDRES_S_Fasted}
      if(LOZ_DAYS==1 & FREQ==1 & FED_STATUS==2) {PROPRES <- PROPRES_S_Fed; ADDRES <- ADDRES_S_Fed}
      if(LOZ_DAYS==1 & FREQ==2) {PROPRES <- PROPRES_M; ADDRES <- ADDRES_M}
      if(LOZ_DAYS>=2) {PROPRES <- PROPRES_M; ADDRES <- ADDRES_M}

   } #end of select ==2

   SIM.data <- ddply(dfADVAN, .(ID), TwoCompOralFourTranist)
   #Calculate IPRED
    SIM.data$IPRED <- SIM.data$A2/SIM.data$V2
    SIM.data$IPRED <- ifelse(SIM.data$IPRED <= 0.05, 0.05, SIM.data$IPRED)  #constrain the prediction to
be above 1/10th of LLOQ
    SIM.data$logIPRED <- log(SIM.data$IPRED)
    #add residual error
    SIM.data$W <- sqrt(PROPRES**2+ADDRES**2/exp(SIM.data$logIPRED)**2)
    SIM.data$W<-ifelse(SIM.data$W >= 1,0.1,SIM.data$W)
    SIM.data$logDV <- SIM.data$logIPRED+SIM.data$W
    SIM.data$DV <- exp(SIM.data$logDV)          #DV in ng/ml

  }) #end of isolate

SIM.data <- as.data.frame(SIM.data)

}) #end of reactive function

#Generate a plot for the data
output$plotCONC <- renderPlot({

  plotobj <- ggplot(SIM.data())

 titletext <- expression(atop("Simulated Itraconazole Concentrations",
                              atop("Red line is the median (200 subjects)",italic("Press the button above
to show 90% prediction interval"))))

  plotobj <- plotobj + stat_summary(aes(x=TIME, y=DV), fun.y=median, geom="line", colour="red", size=1)
  plotobj <- plotobj + ggtitle(titletext)
  plotobj <- plotobj + scale_y_continuous("Itraconazole plasma concentration (ng/mL)", labels=comma)
```

```
  plotobj <- plotobj +  scale_x_continuous("Time after first dose (hours)")

if(input$CIITRACONAZOLE==TRUE) {
  plotobj <- plotobj + stat_summary(aes(x=TIME, y=DV), geom="ribbon", fun.ymin="CI90lo", fun.ymax="CI90hi",
alpha=0.5)
   } #end of CIitraconazole

print(plotobj)
}) #end of render plot

#For downloading simulated data
output$downloadsimdata <- downloadHandler(
  filename = function() {
    "SIM.data.itraconazole.csv"
  },

  content = function(file) {
    write.csv(SIM.data(),row.names=F, file)
  }
)   #Close downloadHandler


}) #end shiny server
```

ADVAN_2compOral_4Transit_Function.R

```
#for calculating 90%CI
CI90lo <- function(x) quantile(x, probs=0.05)
CI90hi <- function(x) quantile(x, probs=0.95)


#-----------------------------------------------------------------------
# ADVAN-style analytical solution for 2 compartment oral_4Transit_Model
#-----------------------------------------------------------------------
TwoCompOralFourTranist <- function(d){
#Accepts a NONMEM style data frame for 1 subject with columns for TIME, AMT,MDV,DV, CL, V2, Q, V3, KTR &
F1
#Returns a dataframe with populated columns for A1, A2, A3 and DV

  #set initial values in the compartments
  d$A1[d$TIME==0] <- d$AMT[d$TIME==0]*d$F1[1]   # Amount in the absorption (GUT) compartment at time zero.
  d$A4[d$TIME==0] <- 0                    # Amount in the first transit
  d$A5[d$TIME==0] <- 0                    # Amount in the 2nd transit
  d$A6[d$TIME==0] <- 0                    # Amount in the 3rd transit
  d$A7[d$TIME==0] <- 0                    # Amount in the 4th transit
  d$A2[d$TIME==0] <- 0                    # Amount in the central compartment at time zero.
  d$A3[d$TIME==0] <- 0                    # Amount in the peripheral compartment at time zero.
  #This loop calculates micro-rate constants based on individual's PK parameter values.
  #It uses these values to calculate macro-rate constants (Lambda1/lambda2).
  #Rate constants(micro- and macro), along other parameters, are used in the equations to calculate drug
amounts in each compartment.
  #The loop advances the solution from one time interval to the next.
  #It also calculates the concentration in the central compartment.

    k23 <- d$k23[1] #it is not changing with time so we can put it outside the for-loop
    k32 <- d$k32[1] #Q/d$V3
    KTR <- d$KTR[1]
    k30 <- 0

  for(i in 2:nrow(d))
  {
    k20 <- d$k20[i]  #clearance is changing with time, so k20 is changing with time as well. So, it should
be inside the loop!
```

```
    E2 <- k20+k23
    E3 <- k32+k30

    #calculate hybrid rate constants
    lambda1 = 0.5*((E2+E3)+sqrt((E2+E3)^2-4*(E2*E3-k23*k32)))
    lambda2 = 0.5*((E2+E3)-sqrt((E2+E3)^2-4*(E2*E3-k23*k32)))

    t <- d$TIME[i]-d$TIME[i-1]

    A4last <- d$A4[i-1]   #1st transit
    A5last <- d$A5[i-1]   #2nd transit
    A6last <- d$A6[i-1]   #3rd tranist
    A7last <- d$A7[i-1]   #4th tranist
    A2last <- d$A2[i-1]   #central compartment
    A3last <- d$A3[i-1]   #peripheral compartment
    A1last <- d$A1[i-1]   #Gut compartment

    #Transit compartments
    d$A4[i] = A4last*exp(-t*KTR)+KTR*A1last*t*exp(-t*KTR)
    d$A5[i] = A5last*exp(-t*KTR)+KTR*A4last*t*exp(-t*KTR)+0.5*KTR**2*A1last*t**2*exp(-t*KTR)
    d$A6[i]           =           A6last*exp(-t*KTR)+KTR*A5last*t*exp(-t*KTR)+0.5*KTR**2*A4last*t**2*exp(-t*KTR)+(1/6)*KTR**3*A1last*t**3*exp(-t*KTR)
    d$A7[i]           =           A7last*exp(-t*KTR)+KTR*A6last*t*exp(-t*KTR)+0.5*KTR**2*A5last*t**2*exp(-t*KTR)+(1/6)*KTR**3*A4last*t**3*exp(-t*KTR)+(1/24)*KTR**4*A1last*t**4*exp(-t*KTR)

    A2term3               =               (exp(-t*lambda1)*((A2last*k32+A3last*k32)-A2last*lambda1)-exp(-t*lambda2)*((A2last*k32+A3last*k32)-A2last*lambda2))/(lambda2-lambda1)

    A2term2      =      KTR*k32*(A7last*(exp(-t*KTR)/((lambda1-KTR)*(lambda2-KTR))+exp(-t*lambda1)/((KTR-lambda1)*(lambda2-lambda1))+exp(-t*lambda2)/((KTR-lambda2)*(lambda1-lambda2)))
    +A6last*KTR*(exp(-t*KTR)*(-lambda1-lambda2+2*KTR)/((lambda1-KTR)**2*(KTR-lambda2)**2)
                 -exp(-t*lambda1)/((lambda1-lambda2)*(lambda1-KTR)**2)
                 +exp(-t*lambda2)/((lambda1-lambda2)*(lambda2-KTR)**2)
                 -exp(-t*KTR)*t/((lambda1-KTR)*(KTR-lambda2)))
    +A5last*KTR**2*((exp(-t*KTR)*(-lambda1**2-lambda1*lambda2+3*lambda1*KTR-lambda2**2+3*lambda2*KTR-3*KTR**2))/((lambda1-KTR)**3*(KTR-lambda2)**3)
                    -exp(-t*KTR)*t**2/(2*(lambda1-KTR)*(KTR-lambda2))+exp(-t*KTR)*t*(-lambda1-lambda2+2*KTR)/((lambda1-KTR)**2*(KTR-lambda2)**2)
                    +exp(-t*lambda1)/((lambda1-lambda2)*(lambda1-KTR)**3)-exp(-t*lambda2)/((lambda1-lambda2)*(lambda2-KTR)**3))
```

```
    +A4last*KTR**3*((exp(-t*KTR)*t*(-lambda1**2-lambda1*lambda2+3*lambda1*KTR-lambda2**2+3*lambda2*KTR-
3*KTR**2))/((lambda1-KTR)**3*(KTR-lambda2)**3)
                    +exp(-t*KTR)*(-lambda1**3-lambda1**2*lambda2+4*lambda1**2*KTR-
lambda1*lambda2**2+4*lambda1*lambda2*KTR-6*lambda1*KTR**2-lambda2**3+4*lambda2**2*KTR-
6*lambda2*KTR**2+4*KTR**3)/((lambda1-KTR)**4*(KTR-lambda2)**4)
                    -exp(-t*KTR)*t**3/(6*(lambda1-KTR)*(KTR-lambda2))
                    +exp(-t*KTR)*t**2*(-lambda1-lambda2+2*KTR)/(2*(lambda1-KTR)**2*(KTR-lambda2)**2)-
exp(-t*lambda1)/((lambda1-lambda2)*(lambda1-KTR)**4)
                    +exp(-t*lambda2)/((lambda1-lambda2)*(lambda2-KTR)**4))
    +A1last*KTR**4*((exp(-t*KTR)*t**2*(-lambda1**2-lambda1*lambda2+3*lambda1*KTR-
lambda2**2+3*lambda2*KTR-3*KTR**2))/(2*(lambda1-KTR)**3*(KTR-lambda2)**3)
                    +exp(-t*KTR)*t*(-lambda1**3-lambda1**2*lambda2+4*lambda1**2*KTR-
lambda1*lambda2**2+4*lambda1*lambda2*KTR-6*lambda1*KTR**2-lambda2**3+4*lambda2**2*KTR-
6*lambda2*KTR**2+4*KTR**3)/((lambda1-KTR)**4*(KTR-lambda2)**4)
                    +(exp(-t*KTR)/((lambda1-KTR)**5*(KTR-lambda2)**5))*(-lambda1**4-
lambda1**3*lambda2+5*lambda1**3*KTR-lambda1**2*lambda2**2+5*lambda1**2*lambda2*KTR-10*lambda1**2*KTR**2-
lambda1*lambda2**3+5*lambda1*lambda2**2*KTR-10*lambda1*lambda2*KTR**2+10*lambda1*KTR**3-
lambda2**4+5*lambda2**3*KTR-10*lambda2**2*KTR**2+10*lambda2*KTR**3-5*KTR**4)
                    -exp(-t*KTR)*t**4/(24*(lambda1-KTR)*(KTR-lambda2))+exp(-t*KTR)*t**3*(-lambda1-
lambda2+2*KTR)/(6*(lambda1-KTR)**2*(KTR-lambda2)**2)
                    +exp(-t*lambda1)/((lambda1-lambda2)*(lambda1-KTR)**5)-exp(-t*lambda2)/((lambda1-
lambda2)*(lambda2-KTR)**5)))

    A2term1             =             KTR*(A7last*(exp(-t*KTR)*KTR/((lambda1-KTR)*(KTR-lambda2))+exp(-
t*lambda2)*lambda2/((lambda1-lambda2)*(lambda2-KTR))-exp(-t*lambda1)*lambda1/((lambda1-lambda2)*(lambda1-
KTR)))
    +A6last*KTR*(exp(-t*KTR)*(lambda1*lambda2-KTR**2)/((lambda1-KTR)**2*(KTR-lambda2)**2)+exp(-
t*lambda1)*lambda1/((lambda1-lambda2)*(lambda1-KTR)**2)-exp(-t*lambda2)*lambda2/((lambda1-
lambda2)*(lambda2-KTR)**2)+exp(-t*KTR)*t*KTR/((lambda1-KTR)*(KTR-lambda2)))
    +A5last*KTR**2*(exp(-t*KTR)*(lambda1**2*lambda2+lambda1*lambda2**2-
3*lambda1*lambda2*KTR+KTR**3)/((lambda1-KTR)**3*(KTR-lambda2)**3)
                    +exp(-t*KTR)*t*(lambda1*lambda2-KTR**2)/((lambda1-KTR)**2*(KTR-lambda2)**2)+exp(-
t*KTR)*KTR*t**2/(2*(lambda1-KTR)*(KTR-lambda2))
                    -exp(-t*lambda1)*lambda1/((lambda1-lambda2)*(lambda1-KTR)**3)+exp(-
t*lambda2)*lambda2/((lambda1-lambda2)*(lambda2-KTR)**3))
    +A4last*KTR**3*(exp(-t*KTR)*t*(lambda1**2*lambda2+lambda1*lambda2**2-
3*lambda1*lambda2*KTR+KTR**3)/((lambda1-KTR)**3*(KTR-lambda2)**3)
                    +exp(-t*KTR)*(lambda1**3*lambda2+lambda1**2*lambda2**2-
4*lambda1**2*lambda2*KTR+lambda1*lambda2**3-4*lambda1*lambda2**2*KTR+6*lambda1*lambda2*KTR**2-
KTR**4)/((lambda1-KTR)**4*(KTR-lambda2)**4)
                    +exp(-t*KTR)*t**2*(lambda1*lambda2-KTR**2)/(2*(lambda1-KTR)**2*(KTR-lambda2)**2)
```

```
                    +exp(-t*KTR)*KTR*t**3/(6*(lambda1-KTR)*(KTR-lambda2))
                    +exp(-t*lambda1)*lambda1/((lambda1-lambda2)*(lambda1-KTR)**4)
                    -exp(-t*lambda2)*lambda2/((lambda1-lambda2)*(lambda2-KTR)**4))
    +A1last*KTR**4*((exp(-t*KTR)*t**2*(lambda1**2*lambda2+lambda1*lambda2**2-
3*lambda1*lambda2*KTR+KTR**3))/(2*(lambda1-KTR)**3*(KTR-lambda2)**3)
                    +exp(-t*KTR)*t*(lambda1**3*lambda2+lambda1**2*lambda2**2-
4*lambda1**2*lambda2*KTR+lambda1*lambda2**3-4*lambda1*lambda2**2*KTR+6*lambda1*lambda2*KTR**2-
KTR**4)/((lambda1-KTR)**4*(KTR-lambda2)**4)
                    +(exp(-t*KTR)/((lambda1-KTR)**5*(KTR-
lambda2)**5))*(lambda1**4*lambda2+lambda1**3*lambda2**2-5*lambda1**3*lambda2*KTR+lambda1**2*lambda2**3-
5*lambda1**2*lambda2**2*KTR+10*lambda1**2*lambda2*KTR**2+lambda1*lambda2**4-
5*lambda1*lambda2**3*KTR+10*lambda1*lambda2**2*KTR**2-10*lambda1*lambda2*KTR**3+KTR**5)
                    +exp(-t*KTR)*t**3*(lambda1*lambda2-KTR**2)/(6*(lambda1-KTR)**2*(KTR-lambda2)**2)
                    +exp(-t*KTR)*KTR*t**4/(24*(lambda1-KTR)*(KTR-lambda2))
                    -exp(-t*lambda1)*lambda1/((lambda1-lambda2)*(lambda1-KTR)**5)
                    +exp(-t*lambda2)*lambda2/((lambda1-lambda2)*(lambda2-KTR)**5)))

    d$A2[i] = A2term1+A2term2+A2term3

    A3term1                 =               (exp(-t*lambda1)*((A3last*E2+k23*A2last)-A3last*lambda1)-exp(-
t*lambda2)*((A3last*E2+k23*A2last)-A3last*lambda2))/(lambda2-lambda1)

    A3term2      =      KTR*k23*(A7last*(exp(-t*KTR)/((lambda1-KTR)*(lambda2-KTR))+exp(-t*lambda1)/((KTR-
lambda1)*(lambda2-lambda1))+exp(-t*lambda2)/((KTR-lambda2)*(lambda1-lambda2)))
                        +A6last*KTR*(exp(-t*KTR)*(-lambda1-lambda2+2*KTR)/((lambda1-KTR)**2*(KTR-
lambda2)**2)-exp(-t*lambda1)/((lambda1-lambda2)*(lambda1-KTR)**2)+exp(-t*lambda2)/((lambda1-
lambda2)*(lambda2-KTR)**2)-exp(-t*KTR)*t/((lambda1-KTR)*(KTR-lambda2)))
                        +A5last*KTR**2*((exp(-t*KTR)*(-lambda1**2-lambda1*lambda2+3*lambda1*KTR-
lambda2**2+3*lambda2*KTR-3*KTR**2))/((lambda1-KTR)**3*(KTR-lambda2)**3)
                                        -exp(-t*KTR)*t**2/(2*(lambda1-KTR)*(KTR-lambda2))+exp(-t*KTR)*t*(-
lambda1-lambda2+2*KTR)/((lambda1-KTR)**2*(KTR-lambda2)**2)
                                        +exp(-t*lambda1)/((lambda1-lambda2)*(lambda1-KTR)**3)-exp(-
t*lambda2)/((lambda1-lambda2)*(lambda2-KTR)**3))
                        +A4last*KTR**3*((exp(-t*KTR)*t*(-lambda1**2-lambda1*lambda2+3*lambda1*KTR-
lambda2**2+3*lambda2*KTR-3*KTR**2))/((lambda1-KTR)**3*(KTR-lambda2)**3)
                                        +exp(-t*KTR)*(-lambda1**3-lambda1**2*lambda2+4*lambda1**2*KTR-
lambda1*lambda2**2+4*lambda1*lambda2*KTR-6*lambda1*KTR**2-lambda2**3+4*lambda2**2*KTR-
6*lambda2*KTR**2+4*KTR**3)/((lambda1-KTR)**4*(KTR-lambda2)**4)
                                        -exp(-t*KTR)*t**3/(6*(lambda1-KTR)*(KTR-lambda2))
                                        +exp(-t*KTR)*t**2*(-lambda1-lambda2+2*KTR)/(2*(lambda1-
KTR)**2*(KTR-lambda2)**2)-exp(-t*lambda1)/((lambda1-lambda2)*(lambda1-KTR)**4)
```

```
                                          +exp(-t*lambda2)/((lambda1-lambda2)*(lambda2-KTR)**4))
                          +A1last*KTR**4*((exp(-t*KTR)*t**2*(-lambda1**2-lambda1*lambda2+3*lambda1*KTR-
lambda2**2+3*lambda2*KTR-3*KTR**2))/(2*(lambda1-KTR)**3*(KTR-lambda2)**3)
                                          +exp(-t*KTR)*t*(-lambda1**3-lambda1**2*lambda2+4*lambda1**2*KTR-
lambda1*lambda2**2+4*lambda1*lambda2*KTR-6*lambda1*KTR**2-lambda2**3+4*lambda2**2*KTR-
6*lambda2*KTR**2+4*KTR**3)/((lambda1-KTR)**4*(KTR-lambda2)**4)
                                          +(exp(-t*KTR)/((lambda1-KTR)**5*(KTR-lambda2)**5))*(-lambda1**4-
lambda1**3*lambda2+5*lambda1**3*KTR-lambda1**2*lambda2**2+5*lambda1**2*lambda2*KTR-10*lambda1**2*KTR**2-
lambda1*lambda2**3+5*lambda1*lambda2**2*KTR-10*lambda1*lambda2*KTR**2+10*lambda1*KTR**3-
lambda2**4+5*lambda2**3*KTR-10*lambda2**2*KTR**2+10*lambda2*KTR**3-5*KTR**4)
                                          -exp(-t*KTR)*t**4/(24*(lambda1-KTR)*(KTR-lambda2))+exp(-
t*KTR)*t**3*(-lambda1-lambda2+2*KTR)/(6*(lambda1-KTR)**2*(KTR-lambda2)**2)
                                          +exp(-t*lambda1)/((lambda1-lambda2)*(lambda1-KTR)**5)-exp(-
t*lambda2)/((lambda1-lambda2)*(lambda2-KTR)**5)))

    d$A3[i] = A3term1+A3term2

    A1last = A1last*exp(-t*KTR)
    d$A1[i] = A1last+d$AMT[i]*d$F1[i]   #F1 is changing with time as well!


  }
  d
}
```