# RでPK解析Simulation　実習

# Jupyter Webにアクセス

# https://jupyter.org/try

# 画面下のほうの ”How to use Jupyter with R” をクリック、しばらく待つ

# File → New Notebook → R

# mrgsolveパッケージをインストール

install.packages("mrgsolve")

library(mrgsolve)

library(tidyverse)

# 1-コンパートメントモデル経口投与

code<-"

$PARAM

Ke = 0.05, Vd = 50, Ka = 0.08333

$INIT

Xa=10000, X=0

$ODE

dxdt\_Xa = -Ka\*Xa;

dxdt\_X = Ka\*Xa - Ke\*X;

$CAPTURE

C = X/Vd; "

# グラフを描く

mod<-mcode("oral", code) %>% update(end = 60, delta = 0.1)

mod %>% mrgsim %>% plot

# 血中濃度のグラフを描く

out <-

mod %>%

mrgsim(end=60)

plot(out, C~time, col="black")

# 縦軸が片対数の血中濃度のグラフを描く

plot(out, C~time, logy="true", ylim=c(1,100), col="black")

# 0.5分までのデータを見る

head(out)

# 2から60分のデータを見る

filter(out, time %in% c(2,5,10,12,15,20,30,40,50,60))

# Vdが違っていた場合の血中濃度変化

code2<-"

$PARAM

Ke = 0.0625, Vd = 40, Ka = 0.08333

$INIT

Xa=10000, X=0

$ODE

dxdt\_Xa = -Ka\*Xa;

dxdt\_X = Ka\*Xa - Ke\*X;

$CAPTURE

C = X/Vd; "

mod2<-mcode("oral2", code2) %>% update(end = 60, delta = 0.1)

out2 <- mod2 %>% mrgsim(end=60)

plot(out2, C~time, logy="true", ylim=c(1,100), col="red")

# 2から60分のデータを見る

filter(out2, time %in% c(2,5,10,12,15,20,30,40,50,60))

# 流速が2.0mL/minだった場合の血中濃度変化

code3<-"

$PARAM

Ke = 0.04, Vd = 50, Ka = 0.06667

$INIT

Xa=10000, X=0

$ODE

dxdt\_Xa = -Ka\*Xa;

dxdt\_X = Ka\*Xa - Ke\*X;

$CAPTURE

C = X/Vd; "

mod3<-mcode("oral3", code3) %>% update(end = 60, delta = 0.1)

out3 <- mod3 %>% mrgsim(end=60)

plot(out3, C~time, logy="true", ylim=c(1,100), col="blue")

# 2から60分のデータを見る

filter(out3, time %in% c(2,5,10,12,15,20,30,40,50,60))

# 投与量が違っていた場合の血中濃度変化

code4<-"

$PARAM

Ke = 0.05, Vd = 50, Ka = 0.08333

$INIT

Xa=8000, X=0

$ODE

dxdt\_Xa = -Ka\*Xa;

dxdt\_X = Ka\*Xa - Ke\*X;

$CAPTURE

C = X/Vd; "

mod4<-mcode("oral4", code4) %>% update(end = 60, delta = 0.1)

out4 <- mod4 %>% mrgsim(end=60)

plot(out4, C~time, logy="true", ylim=c(1,100), col="dark green")

# 2から60分のデータを見る

filter(out4, time %in% c(2,5,10,12,15,20,30,40,50,60))

# グラフを重ねて表示

d<-select(out, time, C)

d2<-select(out2, time, C)

d3<-select(out3, time, C)

d4<-select(out4, time, C)

g <- ggplot() +

geom\_line(data = d, aes(time, C)) +

geom\_line(data = d2, aes(time, C), col="red") +

geom\_line(data = d3, aes(time, C), col="blue") +

geom\_line(data = d4, aes(time, C), col="dark green")

print(g)

g2 <- g + scale\_y\_log10(limits=c(1,110))

print(g2)

# データをまとめて表示

tp<- c(2,5,10,12,15,20,30,40,50,60)

df <- filter(d, time %in% tp)

df2 <- filter(d2, time %in% tp)

df3 <- filter(d3, time %in% tp)

df4 <- filter(d4, time %in% tp)

dfa = cbind(df,df2,df3,df4)

dfa