**Assignment #3 – Optimization and Sensitivity (Updated 2/21/2024)**

**Due: Tuesday, 2024-02-27, 11:59pm**

**PLEASE READ AND FOLLOW THESE INSTRUCTIONS FOR THE ASSIGNMENT**

**Submit to Canvas: the solution (this document, filled out) as a word doc and one zipped folder that includes all your code files. You don’t need to include the output files that your code generates (in fact, please don’t! don’t clog our hard drives), because we’ll generate those when we run the codes.**

**Your codes must work; that means we must be able to open and run your Matlab code to obtain the data; we must then be able to open and run your R code to obtain the visualizations (because the R code uses as input a file that your Matlab code generates, make sure they’re in the same subfolder!). Label the code filenames appropriately, e.g. “yourname\_Q1\_runcode.m”.**

\* Codes and figures should also be included in this report document. Figures should be annotated with a brief figure caption. Graphs should have legible and informative axes.

\* **All** data visualizations (Q1e, Q2, Q3) must be completed using R and ggplot2. See the course website for introductory information and guidance on R and ggplot2.

\* Use this document to insert your solutions and submit. Expand the space between each question as needed to incorporate your solution.

**\* Fonts: use Arial 12 for text; use Courier 11 for code.**

\* Don’t handwrite parts of this assignment. Use Equation Editor, MathType or similar to enter any necessary equations.

**Name: Vivek Booshan**

**Q1. (40 pts)** Let's use a **caffeine** model, but a simpler model than the one we’ve used before: just caffeine, no paraxanthine, no theophylline – a 1-compartment model with just absorption and clearance (the matlab equations file is attached). Even with just a few processes, we still expect variability from person to person. We know that the **volume of distribution is weight-dependent** (~0.6 L/kg in adults; 0.8-0.9 L/kg in neonates) and has some interindividual variability on top of that. We know that **clearance is also age dependent**: 3-7 hour half-life in adults (and twice that in pregnant women), 65-130 hours in neonates. Caffeine also has variability in absorption from the gut. **Remember** just because the given levels of variability are ranges (e.g. 3-7 hours) doesn’t mean that all subjects are within that range; it might be an interquartile range (e.g. give the range of the middle 50% of people or some other definition). So values for individual subjects may or may not be within the range. For parts of Q1 that ask for a number from one of these ranges, use the midpoint of the range.

In the table below, we have **blood measurements of caffeine from five adults**, none of whom is pregnant, at multiple timepoints. Each of them is part of a caffeine study, for which, during their Systems Pharmacology class, at 11am they quickly drink **a 16oz hot brewed coffee (310 mg caffeine).** Unfortunately, **before class**, the five subjects all visited a local coffee shop, and because it was a warm day they each had an **iced café mocha (175 mg caffeine)**, right at 10am. The subjects did not mention this additional coffee at first to the people running the study. You might expect this additional caffeine to somewhat mess with the results; the study proceeded and samples of caffeine concentration in the blood were measured at the following times. You can assume that no-one had caffeine in their system before the events described above.

Note: read all sections of the question first; it’s simplest to set up a code that can simulate all the scenarios listed so plan ahead.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Subject 1** | | **Subject 2** | **Subject 3** | **Subject 4** | | **Subject 5** |
| **weight (lb)** | **111** | | **128** | **143** | **197** | | **222** |
| 11:15 am | 8.8 | | 7.7 | 8.0 | 5.2 | | 6.0 |
| 11:30 am | 9.8 | | 8.6 | 9.0 | 6.0 | | 6.5 |
| 11:45 am | 9.0 | | 7.7 | 8.5 | 5.5 | | 6.1 |
| 1pm | 6.4 | | 5.0 | 7.2 | 3.9 | | 4.5 |
| 4pm | 4.6 | | 3.3 | 6.0 | 2.7 | | 3.2 |
| 7pm | 3.0 | | 1.7 | 4.5 | 1.7 | | 2.0 |
|  |  | |  |  |  | |  |
| *Concentrations in mg/L* | |  | |  |  |  | |
| *Same doses (in mg) for each subject* | | | |  |  |  | |
| **Table 1. Five Patients – concentrations of caffeine in blood over time, following each receiving doses of caffeine described in the question above.** | | | | | | | |

**1a, 1b, 1c.** Complete the twelve rows & five columns in the tables below; in other words, using optimization, identify, for each of the five subjects, **the best values for the two (1b) or three (1c) parameters indicated that the researchers would find, assuming (and this is key) that they don’t know about the earlier caffeine.** They have the experimental data from the subjects, but they assume for the purposes of their model that only the 11am caffeine intake took place. For 1a, there’s no optimization – you’re just giving estimates based on the weights of the subjects.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | | | | | |
| **1a. Assuming the absorption rate constant (*ka*) is the same for each person (half-life of 7 mins), what are the predicted values of *ka*, *kcl* and *Vd* based on weight alone (i.e. not using optimization)?** | | | | | |
|  | **Subject 1** | **Subject 2** | **Subject 3** | **Subject 4** | **Subject 5** |
| ***ka* (hr-1)** | **5.9413** | **5.9413** | **5.9413** | **5.9413** | **5.9413** |
| ***kcl* (hr-1)** | **0.1386** | **0.1386** | **0.1386** | **0.1386** | **0.1386** |
| ***Vd* (L)** | **29.97** | **34.56** | **38.61** | **53.19** | **59.94** |
| ***Vd* (L/kg)** | **0.6** | **0.6** | **0.6** | **0.6** | **0.6** |
|  |  |  |  |  |  |
| **1b. Assuming the absorption rate constant (*ka*) is the same for each person (half-life of 7 mins), use optimization to find best-fit values for *kcl* and *Vd*** | | | | | |
|  | **Subject 1** | **Subject 2** | **Subject 3** | **Subject 4** | **Subject 5** |
| ***ka* (hr-1)** | **5.9413** | **5.9413** | **5.9413** | **5.9413** | **5.9413** |
| ***kcl* (hr-1)** | **0.1677** | **0.2247** | **0.0972** | **0.1834** | **0.1674** |
| ***Vd* (L)** | **19.824** | **21.560** | **22.897** | **31.996** | **29.622** |
| ***Vd* (L/kg)** | **0.3969** | **0.3743** | **0.3558** | **0.3609** | **0.2965** |
|  |  |  |  |  |  |
| **1c. Assuming the absorption rate constant (*ka*) is NOT the same for each person, use optimization to find best-fit values for *ka*, *kcl*, *Vd*** | | | | | |
|  | **Subject 1** | **Subject 2** | **Subject 3** | **Subject 4** | **Subject 5** |
| ***ka* (hr-1)** | **9.3586** | **9.3944** | **9.3619** | **9.2079** | **8.6905** |
| ***kcl* (hr-1)** | **0.1493** | **0.2002** | **0.0838** | **0.1641** | **0.1512** |
| ***Vd* (L)** | **21.136** | **23.080** | **24.321** | **34.087** | **31.337** |
| ***Vd* (L/kg*)*** | **0.4231** | **0.4007** | **0.3780** | **0.3845** | **0.3137** |

**1d.** Differences between the optimization values and the weight-based estimates (which hopefully you saw in comparing 1c vs 1a) would often be assumed to be due to interindividual variation.Here, however, the researchers felt that the optimized values were consistently outside the ranges typically seen. For the four main parameters, explain in which direction the estimates are consistently off (if they are), and why:

Direction (1c vs 1a) Reason

***ka Too high : 10 am coffee leads to more concentration measured in system than otherwise would’ve have been with just the 11 am coffee.***

***kcl*** **ok : dosing regimen does not change clearance rate**

***Vd (L)*** **Too low : the total volume is far smaller than the original volumes of each subject. A smaller volume leads to higher concentration, so the 10 am dose has probably led to this shift because of the increased caffeine in the system.**

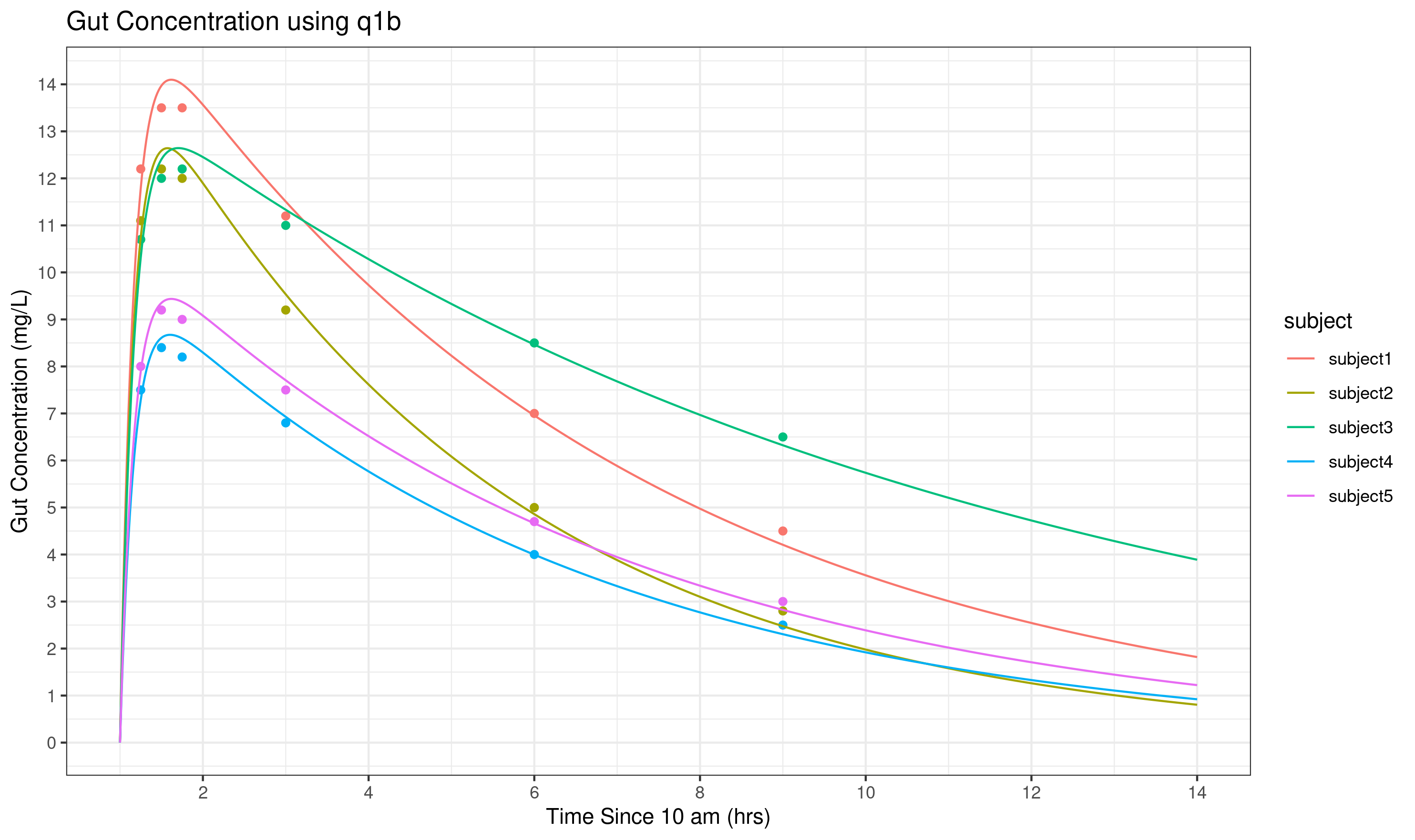
***Vd (L/kg)*** **Too low : It can vary, but it should be much closer to 0.6. Same logic as Vd (L)**

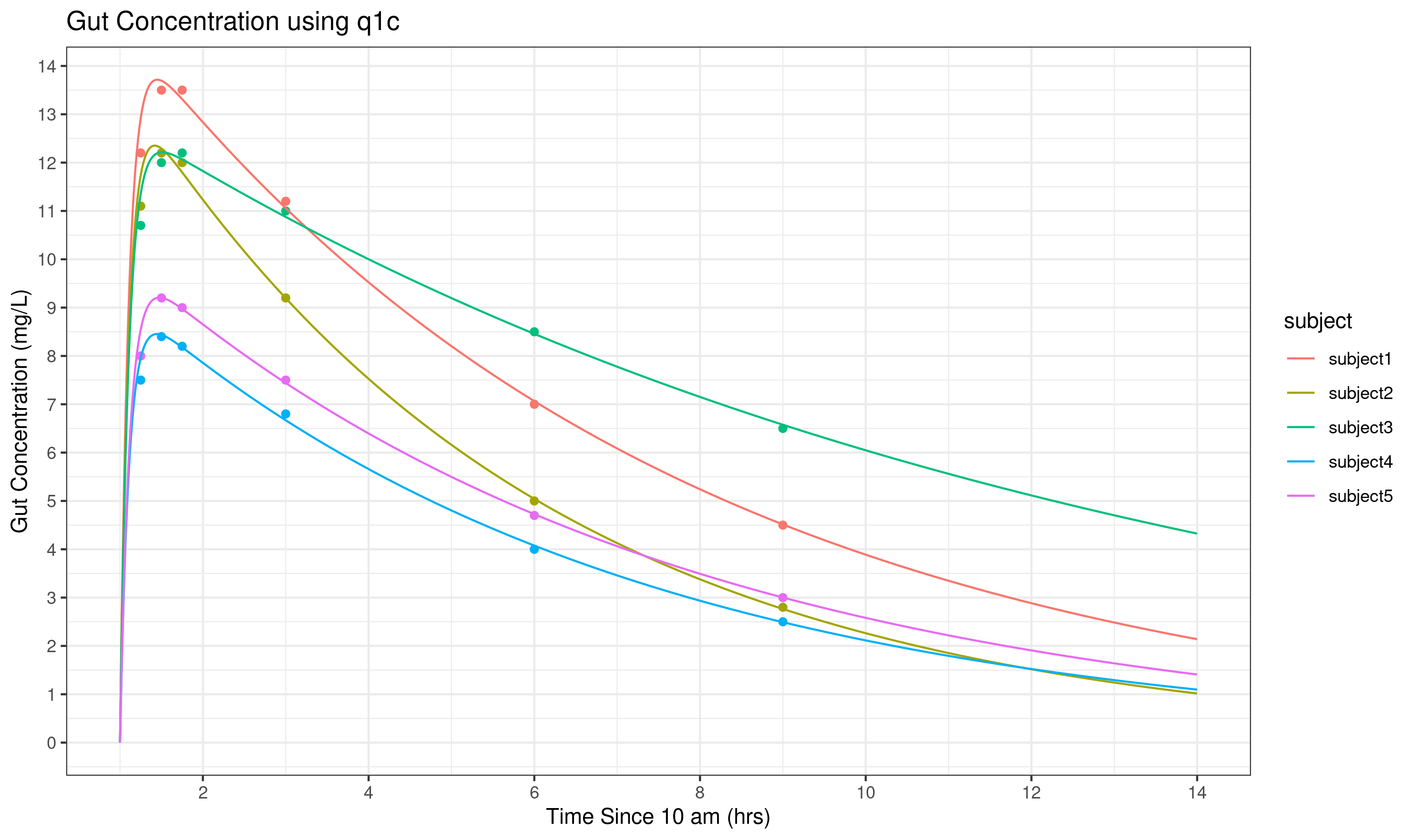
For *direction*, your options are: ‘too high’ (i.e. 1c parameter values are too high), ‘too low’, and ‘ok’

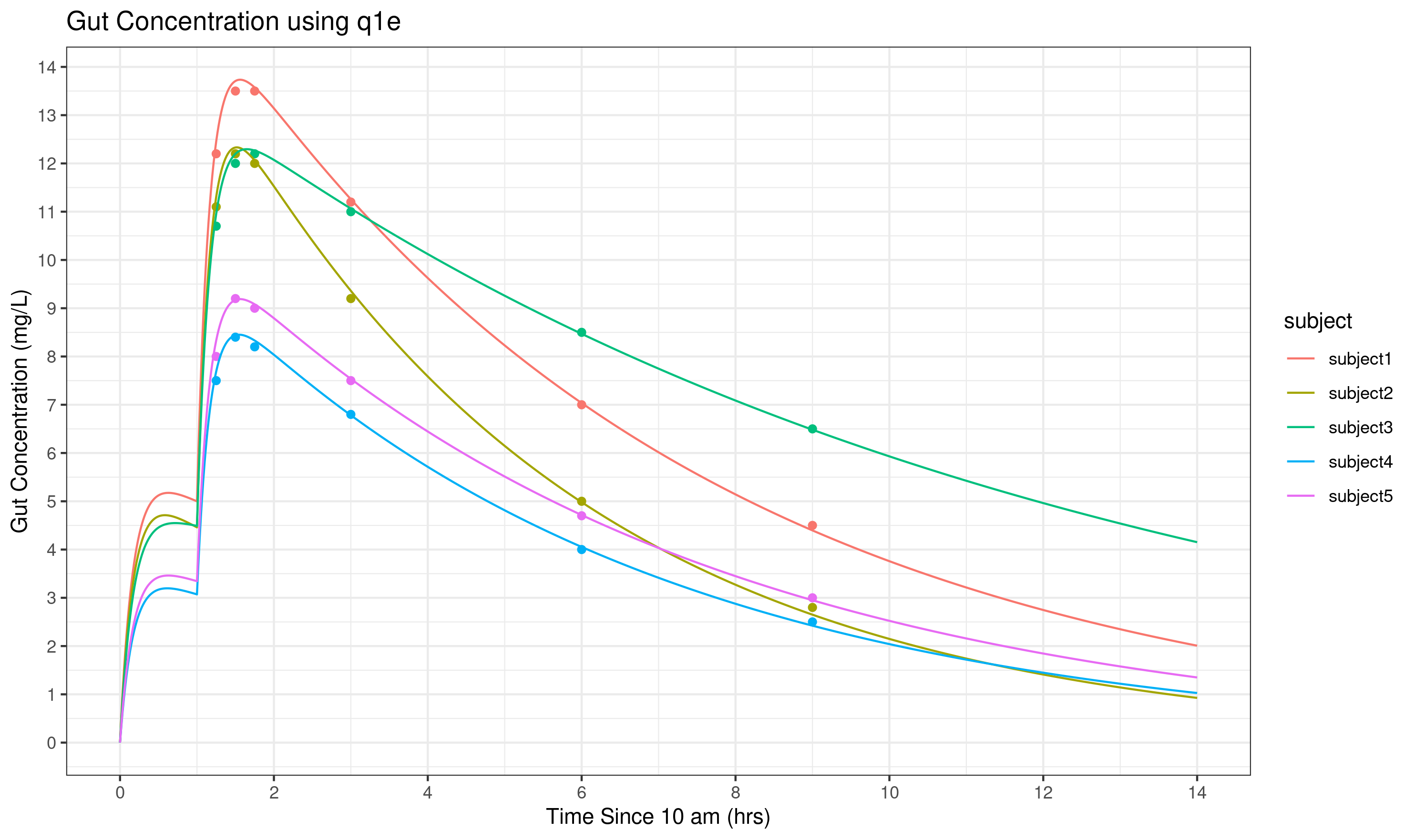
**1e, 1f.** On questioning the subjects, the researchers discovered the existence of the earlier caffeine. Rather than being annoyed at the subjects, they realized that they could adapt their model to simulate their new knowledge! Re-run the 1b and 1c optimizations, using the same experimental data, but now incorporating all the caffeine intake into the simulation. Complete the four rows and five columns in the tables below; the results might not be perfect but they should be ‘better behaved’ than 1b, 1c:

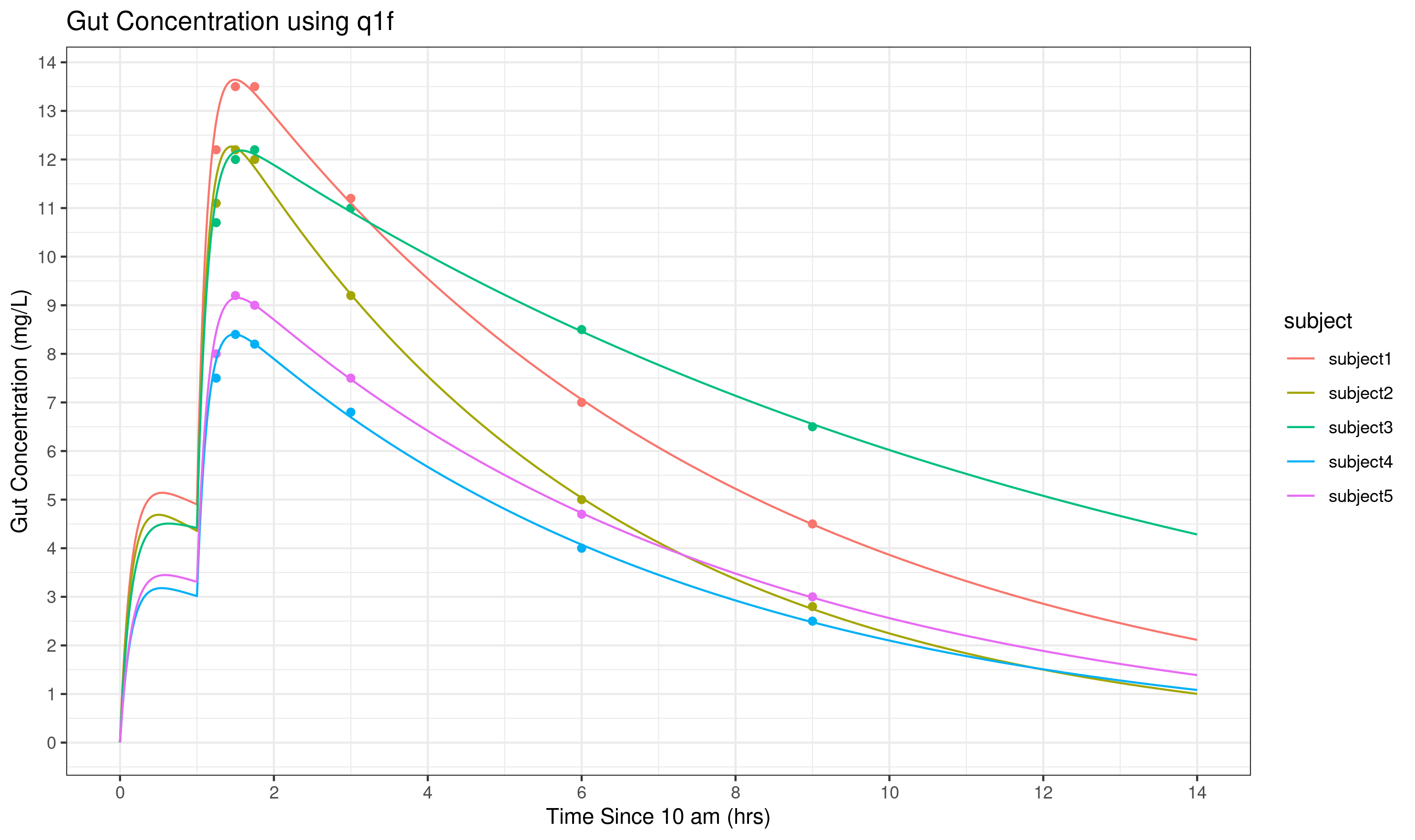
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **1e. Assuming the absorption rate constant (ka) is the same for each person (half-life of 7 mins), and assuming that we DO know about (i.e. do include in the simulation) the 10am dose, use optimization to find best-fit values for *kcl and Vd*** | | | | | |
|  | **Subject 1** | **Subject 2** | **Subject 3** | **Subject 4** | **Subject 5** |
| ***ka* (hr-1)** | **5.9413** | **5.9413** | **5.9413** | **5.9413** | **5.9413** |
| ***kcl* (hr-1)** | **0.1567** | **0.2103** | **0.0891** | **0.1716** | **0.1563** |
| ***Vd* (L)** | **30.648** | **32.857** | **36.092** | **49.269** | **45.825** |
| ***Vd* (L/kg)** | **0.6136** | **0.5704** | **0.5609** | **0.5558** | **0.4587** |
|  | | | | | |
| **1f. Assuming the absorption rate constant (*ka*) is NOT the same for each person, and assuming that we DO know about (i.e. do include in the simulation) the 10am dose, use optimization to find best-fit values for *ka*, *kcl*, *Vd*** | | | | | |
|  | **Subject 1** | **Subject 2** | **Subject 3** | **Subject 4** | **Subject 5** |
| ***ka* (hr-1)** | **7.1948** | **7.3571** | **7.0743** | **7.0956** | **6.5796** |
| ***kcl* (hr-1)** | **0.1508** | **0.2017** | **0.0851** | **0.1656** | **0.1530** |
| ***Vd* (L)** | **31.339** | **33.729** | **36.793** | **50.324** | **46.404** |
| ***Vd* (L/kg)** | **0.6274** | **0.5856** | **0.5718** | **0.5677** | **0.4645** |

**1g.** On one graph each for 1b, 1c, 1e, and 1f, overlay the experimental data points (Table 1, above, graphed as symbols) and the simulation line-curves for caffeine predicted in the blood over time, for the above parameters and for all five subjects (i.e. five lines per graph).You only need to graph from 10am to midnight (fourteen hours).

**Fig 1:** Gut concentration in (mg/L) shown from 11am to midnight. The time scale has been zeroed at 10am for consistency with other plots. Absorption rate constants were held constant (assumed half-life of 7 minutes), while clearance rate and volume of distribution (total and per kg) were optimized using Matlab's lsqnonlin. Model has been optimized **without** knowledge of the 10am caffeine intake.

**Fig 2:** Gut concentration in (mg/L) shown from 11am to midnight. The time scale has been zeroed at 10am for consistency with other plots. Absorption rate, clearance rate and volume of distribution (total and per kg) were optimized using Matlab's lsqnonlin. Model has been optimized without knowledge of the 10am caffeine intake.

**Fig 3:** Gut concentration in (mg/L) shown from 11am to midnight. The time scale has been zeroed at 10am for consistency with other plots. Absorption rate constants were held constant (assumed half-life of 7 minutes), while clearance rate and volume of distribution (total and per kg) were optimized using Matlab's lsqnonlin. Model has been optimized **with** knowledge of the 10am caffeine intake.

**Fig 4:** Gut concentration in (mg/L) shown from 11am to midnight. The time scale has been zeroed at 10am for consistency with other plots. Absorption rate, clearance rate and volume of distribution (total and per kg) were optimized using Matlab's lsqnonlin. Model has been optimized **with** knowledge of the 10am caffeine intake.

**Q2.** **(30 pts) For each of the individuals in Q1, perform a local sensitivity analysis**; for the baseline(s), use your identified subject-specific values from Q1e.

The relevant **outputs** are:

(1) the area under the blood caffeine concentration curve over the 14 hours following 10am (**AUC for 14 hours**);

(2) the area under the blood caffeine concentration curve over the first 60 mins after the 11am dose (**AUC for 60 minutes**); and

(3) the **cost function sum of squares** (a metric of the ‘goodness of fit’ with experimental data) for that subject.

The relevant **input parameters** are:

(1) *D1* (the 10am caffeine dose)

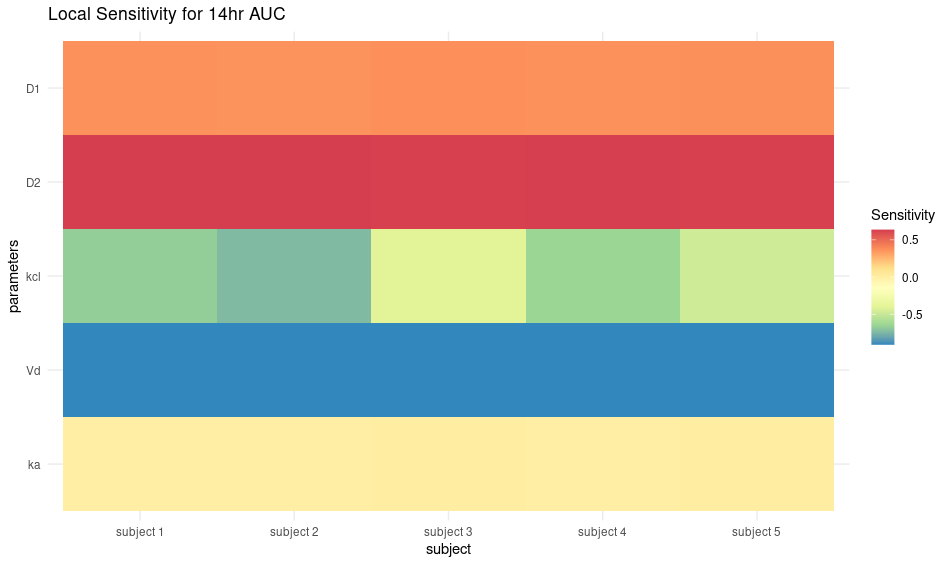
(2) *D2* (the 11am caffeine dose)

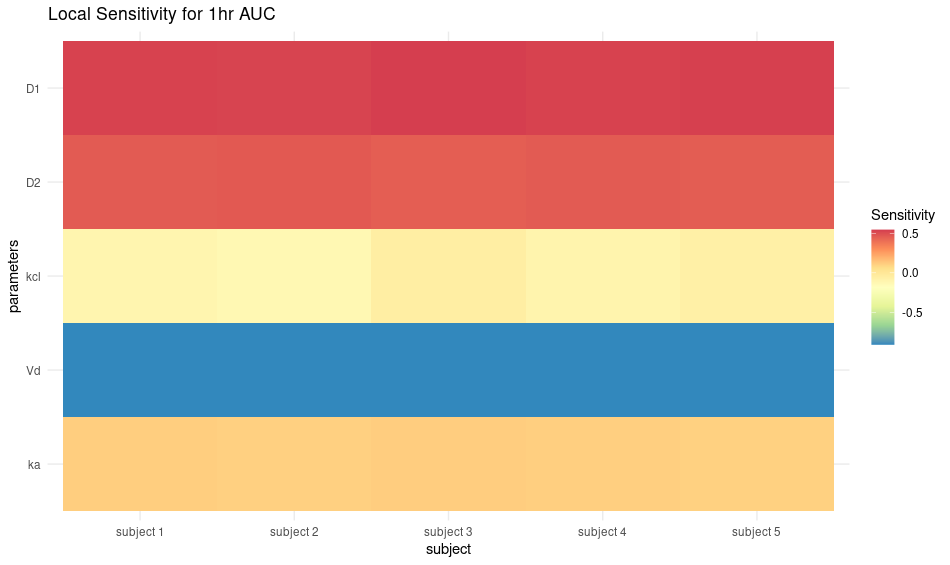
(3) *kCL* (clearance rate constant)

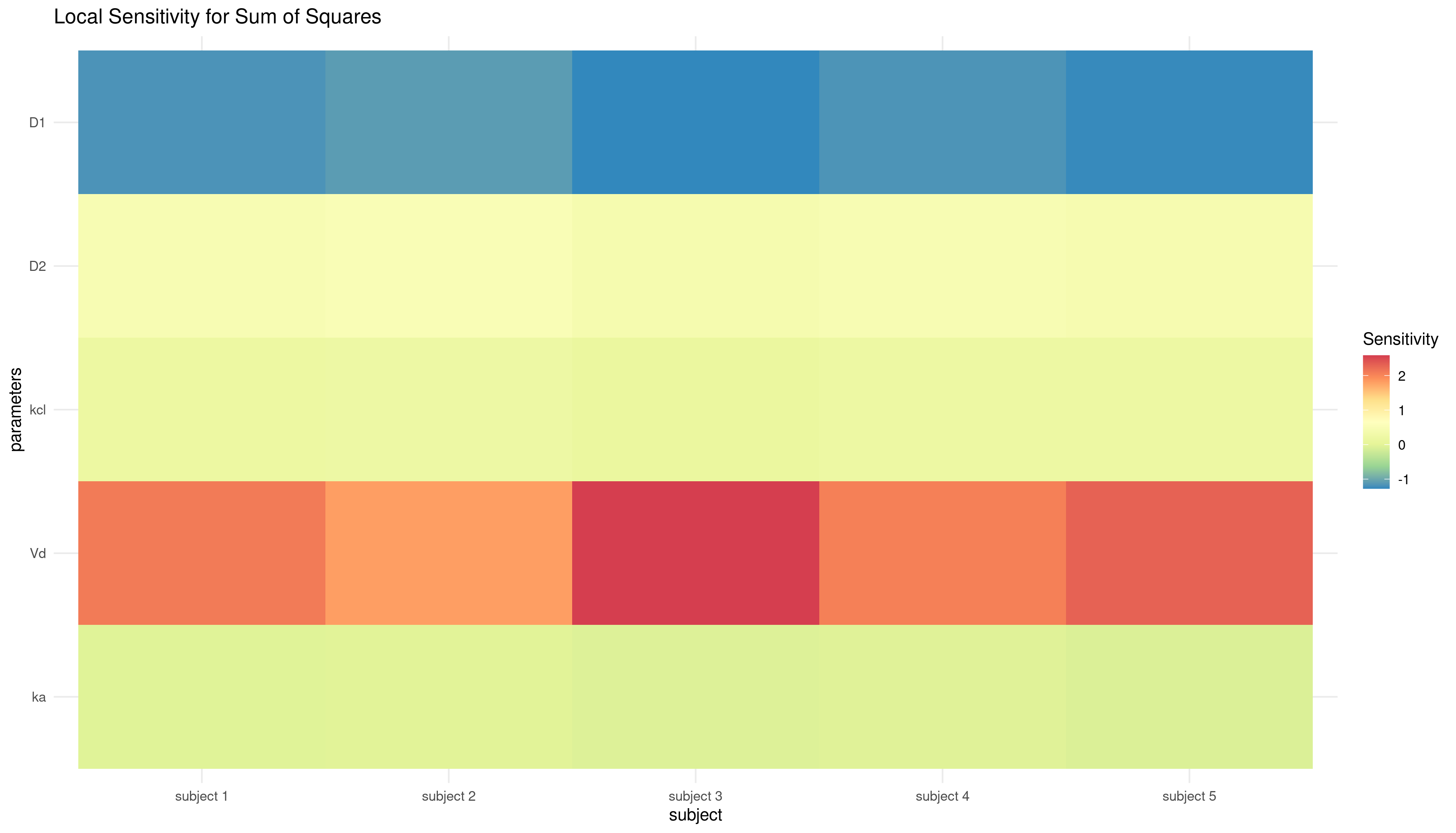
(4) *Vd* (Volume of distribution, in L)

(5) *ka* (absorption rate constant)

**Present the data as 3 heat maps – one for each output listed above**. Remember that since each subject has different values for the outputs, it makes sense to use a normalized sensitivity analysis (e.g. % change in output per % change in parameter).

**Fig 5:** A local sensitivity analysis using normalized sensitivity values (% change in output / % change in parameter), where output was set to the full AUC over the course of 14 hours from 10am of the caffeine blood concentration curve. The parameters from table 1f were used to model each subject. A +10% modification was applied to the 10am dose, 11am dose, clearance rate constant, volume of distribution, and absorption rate constant.

**Fig 6:** A local sensitivity analysis using normalized sensitivity values (% change in output / % change in parameter), where output was set as the AUC for 1 hr after 11 am for the caffeine blood concentration curve. The parameters from table 1f were used to model each subject and the area under the curve. A +10% modification was applied to the 10am dose, 11am dose, clearance rate constant, volume of distribution, and absorption rate constant.

**Fig 6:** A local sensitivity analysis using normalized sensitivity values (% change in output / % change in parameter), where output was set as the AUC for 1 hr after 11 am for the caffeine blood concentration curve. The parameters from table 1f were used to model each subject and the area under the curve. A +10% modification was applied to the 10am dose, 11am dose, clearance rate constant, volume of distribution, and absorption rate constant.

**Q3. (30 pts)** Use the same caffeine pharmacology model, but now in neonates, who receive the caffeine as an initial **loading dose** of 20mg/kg of caffeine citrate (IV bolus) and then **maintenance doses** of 5 mg/kg caffeine citrate (also IV bolus) every subsequent 24 hours for a total of 1 week.

Perform a two-dimensional **global sensitivity analysis** of this model. For a global sensitivity analysis, we aren’t looking at local small changes, but instead at the impact of changes over larger ranges. **Explore the following ranges: weight 3.0-8.0 lbs and clearance half-life 50-150 hours.** Assume that the **volume of distribution is 0.85 L/kg**.

For each range, use 21 linearly-spaced steps including the ends of the ranges – e.g. linspace(3,8,21) or [3.0:.25:8.0] (think about: is one of these formats better or worse than the other? If so, why? Not a question we’re asking you to answer here). That should give you 21\*21=441 simulations. Plan ahead, your computer might be tied up for a little while running simulations – e.g. test your code with fewer points to work out any problems and then run it for the full number of points.

**Display the results as heatmaps of the values of the following key outputs (one heatmap per output):**

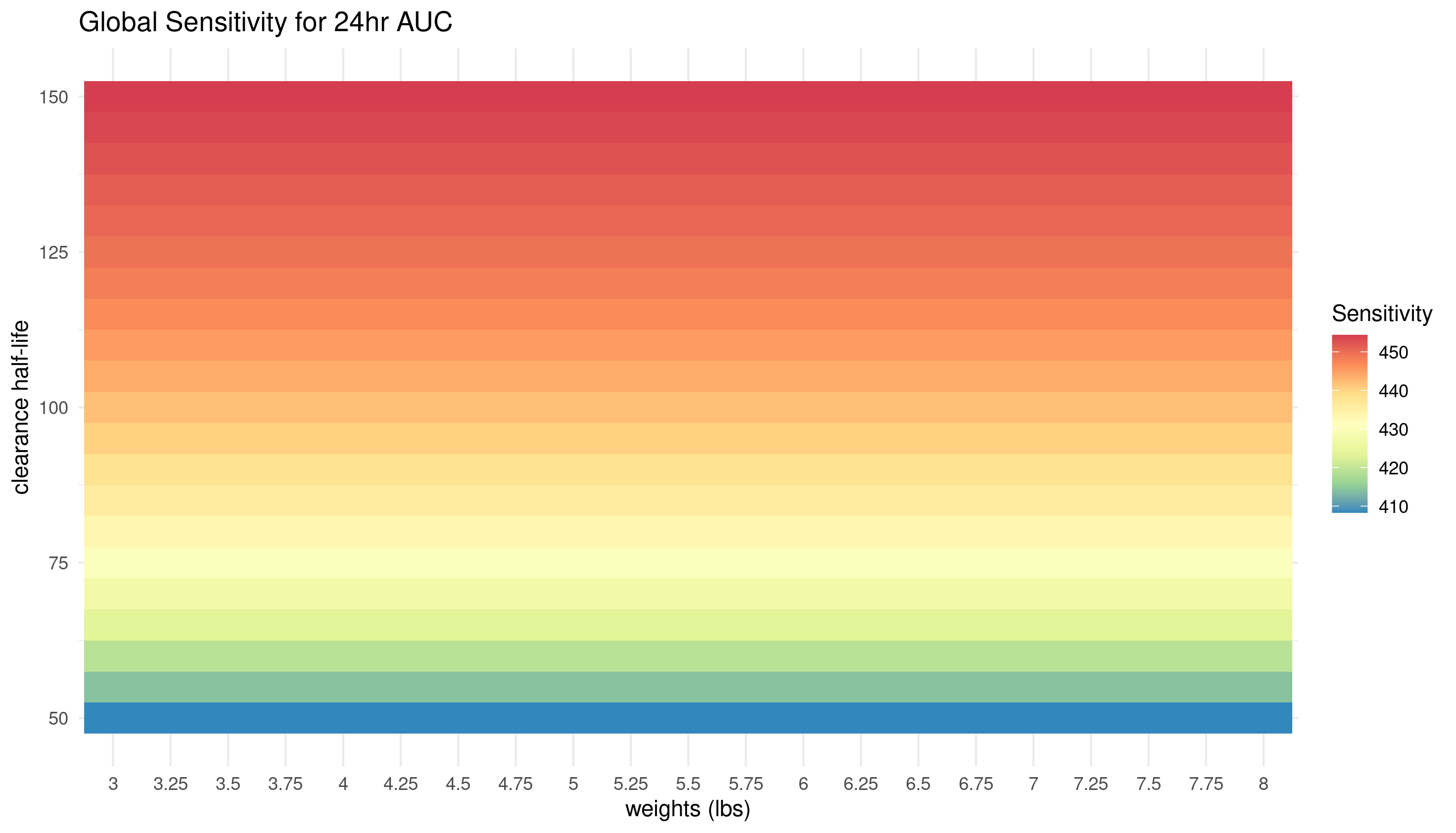
**AUC of caffeine in the blood for the first 24 hours**

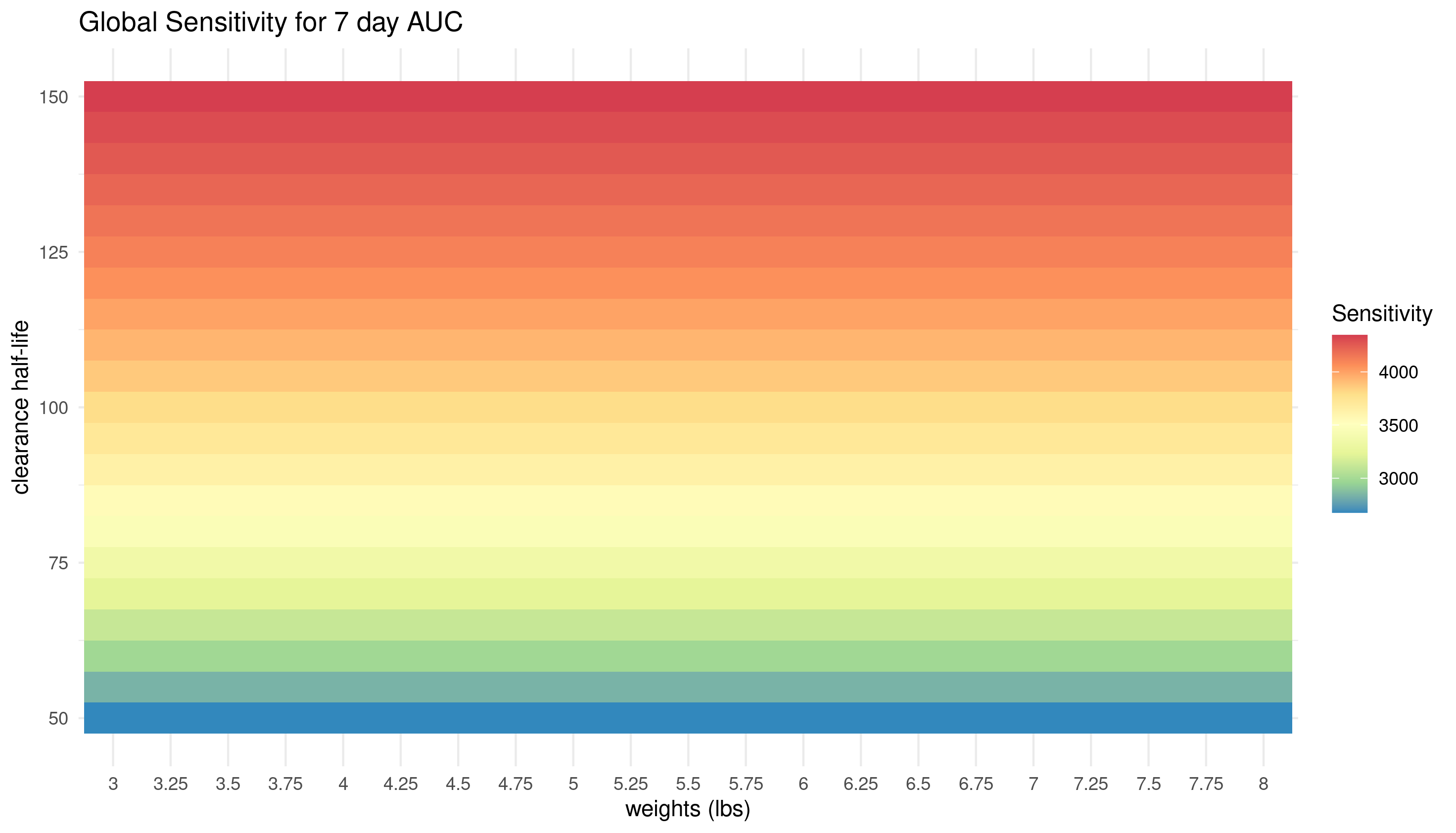
**AUC of caffeine in the blood for 7 days**

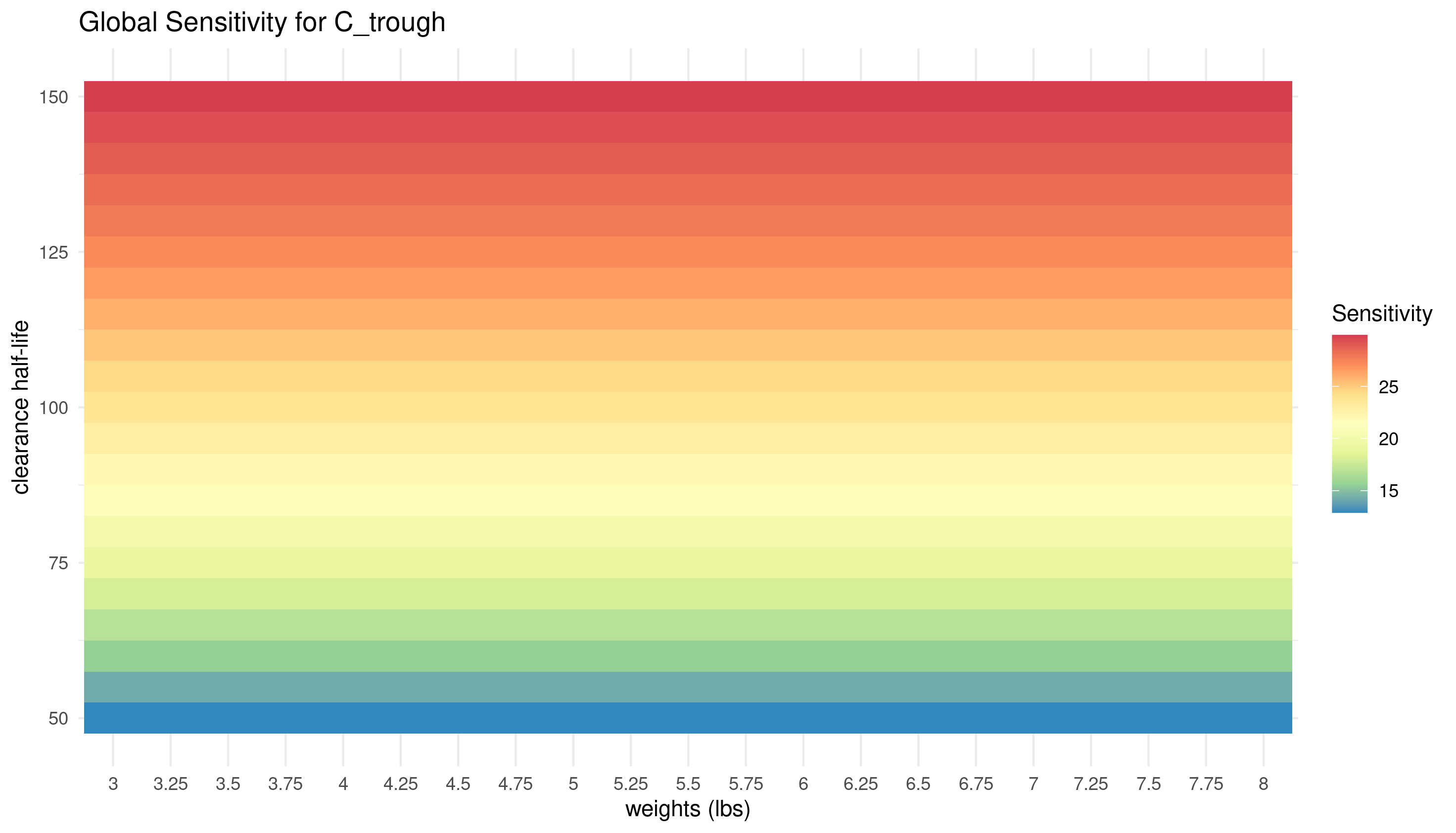
**Ctrough (use the caffeine concentration at 7 days as a surrogate for that)**

Note that in a global sensitivity analysis, we commonly don’t use a normalized sensitivity calculation; we just visualize the actual output itself, and we see (by eye) how much that is changing across the parameter ranges.

*[INSERT RESULTS HERE; PASTE CODE AT END OF DOCUMENT]*

**Fig 7:** A neonatal global sensitivity analysis comparing the effects of clearance half life and weight (in lbs). Weights range from 3 to 8 lbs and are evenly spaced by ¼ lbs , while clearance half-lives range from 50-150 hrs with even spacing every 5 hours. Sensitivity was determined using the AUC of the first 24 hours after introduction of caffeine.

**Fig 8:** A neonatal global sensitivity analysis comparing the effects of clearance half life and weight (in lbs). Weights range from 3 to 8 lbs and are evenly spaced by ¼ lbs , while clearance half-lives range from 50-150 hrs with even spacing every 5 hours. Sensitivity was determined using the AUC of the first 7 days after introduction of caffeine.

**Fig 9:** A neonatal global sensitivity analysis comparing the effects of clearance half life and weight (in lbs). Weights range from 3 to 8 lbs and are evenly spaced by ¼ lbs , while clearance half-lives range from 50-150 hrs with even spacing every 5 hours. Sensitivity was determined using the minimum serum concentration (C\_trough), where the final concentration value after 7 days was used as a surrogate.

**CODES TO INCLUDE IN THIS DOCUMENT (include these at the end, each one clearly titled/labeled. Remember that you should upload all codes necessary to run your model as separate attachments).**

Q1. Optimization driver code and cost function code; visualization code.

**Q1 OPTIMIZATION DRIVER CODE**

subjects = 5;

abs\_halflife = 7/60; %from question

cl\_halflife = 5; %from prompt

ka = log(2) / abs\_halflife;

kcl = log(2) / cl\_halflife;

q=0;

Vd = 0.6; % L / kg

weights = [111, 128, 143, 197, 222]\*0.45; %convert to kg

V = Vd \* weights;

t\_real =[0.25, 0.5, 0.75, 2, 5, 8]; % measurements in hours since 11am

y0 = [0, 0, 310]; % initial coffee @ 11am

y\_real = [

12.2, 11.1, 10.7, 7.5, 8.0; %11:15

13.5, 12.2, 12.0, 8.4, 9.2; %11:30

13.5, 12.0, 12.2, 8.2, 9.0; %11:45

11.2, 9.2, 11.0, 6.8, 7.5; %02;00

7.0, 5.0, 8.5, 4.0, 4.7; %04;00

4.5, 2.8, 6.5, 2.5, 3.0; %07;00

]; % mg/L

%% 1a

% create subject table from previously calculated parameters

table\_1a = zeros(4, 5);

table\_1a(1, :) = q;

table\_1a(2, :) = kcl;

table\_1a(3, :) = V;

table\_1a(4, :) = ka;

table\_1a(5, :) = Vd;

disp('q1a');

disp(table\_1a);

%% 1b 1c 1e 1f

table = zeros(subjects, subjects, 4);

% solve for 1b and 1c

for i = 1:4

% initialization of caffeineError arguments

restrict = [1, 0, 0, mod(i, 2)]; % if 1, restrict parameter

is10am = (i == 3 || i == 4); % if 1, implement 10am algo

if is10am

y0 = [0, 0, 175];

else

y0 = [0, 0, 310];

end

%iterate through each subject

for subject = 1:numel(V)

initial\_guess\_params = [0, kcl, V(subject), ka]; % initial guess

caff\_err = @(params) caffeineError( ... % anon error function

t\_real, ...

y\_real(:, subject), ...

y0, ...

params, ...

restrict, ...

initial\_guess\_params, ...

is10am ...

);

options = optimoptions( ... % use LM algo and hide displays

'lsqnonlin', ...

'Algorithm', 'levenberg-marquardt', ...

'Display','none' ...

);

% tic

optim\_params = lsqnonlin( ... % solve lsqnonlin

caff\_err, ...

initial\_guess\_params, ...

[], [], ...

options ...

);

% toc

% add Vd (L/kg)

table(:, subject, i) = [optim\_params, optim\_params(3) / weights(subject)];

end

end

% assign respective tables and display

table\_1b = table(:, :, 1);

table\_1c = table(:, :, 2);

table\_1e = table(:, :, 3);

table\_1f = table(:, :, 4);

% takes a hot minute

disp('q1b'); disp(table\_1b);

disp('q1c'); disp(table\_1c);

disp('q1e'); disp(table\_1e);

disp('q1f'); disp(table\_1f);

%% Q1g

for subject = 1:subjects

FILE\_NAME=sprintf('subject%d\_experimental', subject);

writematrix([t\_real' y\_real(:, subject)], FILE\_NAME);

end

% q1abc

question = ['a', 'b', 'c'];

table\_abc = zeros(subjects, subjects, 3);

table\_abc(:, :, 1) = table\_1a;

table\_abc(:, :, 2) = table\_1b;

table\_abc(:, :, 3) = table\_1c;

for i=1:3

for subject = 1:subjects

subject\_param = table\_abc(1:end-1, subject, i);

y0 = [0, 0, 310];

[t, y] = ode45(@(t, y) CaffeineODE(t, y, subject\_param), 1:1/60:14, y0);

FILE\_NAME = sprintf('subject%dtable%s', subject, question(i));

writematrix([t y(:, 1)], FILE\_NAME);

end

end

% q1ef

table\_ef = zeros(subjects, subjects, 2);

table\_ef(:, :, 1) = table\_1e;

table\_ef(:, :, 2) = table\_1f;

question = ['e', 'f'];

for i=1:2

for subject= 1:subjects

subject\_param = table\_ef(1:end-1, subject, i);

y0 = [0, 0, 175];

[t1, y1] = ode45(@(t, y) CaffeineODE(t, y, subject\_param), 0:1/60:1, y0);

y0 = y1(end, :);

y0(end) = y0(end) + 310;

[t2, y2] = ode45(@(t, y) CaffeineODE(t, y, subject\_param), 1:1/60:14, y0);

FILE\_NAME = sprintf('subject%dtable%s', subject, question(i));

t = [t1(:, 1); t2(:, 1)];

y = [y1(:, 1); y2(:, 1)];

writematrix([t, y], FILE\_NAME)

end

end

**Q1 COST FUNCTION CODE**

function error = caffeineError( ...

t\_measured, y\_measured, ...

y0, p, ...

restrict, initial\_guess\_params, is10am ...

)

arguments

t\_measured (1, :) double

y\_measured (1, :) double

y0 (1, 3) double

p (1, 4) double

restrict (1, 4) double

initial\_guess\_params (1, 4) double

is10am logical = 0

end

p = p.\*(1 - restrict) + initial\_guess\_params.\*restrict;

if is10am

[~, y] = ode45(@(t, y) CaffeineODE(t, y, p), 0:1/4:1, y0);

y0 = y(end, :);

y0(end) = y0(end) + 310;

end

error = zeros(size(t\_measured));

[t, y] = ode45(@(t, y) CaffeineODE(t, y, p), 0:1/10:14, y0);

for j=1:length(t\_measured)

teval = abs(t - t\_measured(j));

[~, tindex] = min(teval);

error(j) = y(tindex) - y\_measured(j);

end

end

**Q1 PLOTTING CODE**

setwd(Sys.getenv("PWD"))

library(ggplot2)

library(scales)

hex = hue\_pal()(5) #needed to match point with line

questions = c('a', 'b', 'c', 'e', 'f')

for (j in 1:5) {

data = c() #init data frame

p = ggplot() +

labs(title=paste0('Gut Concentration using q1', questions[j])) +

theme\_bw() +

scale\_x\_continuous(name = 'Time Since 10 am (hrs)', breaks=(seq(0, 14, 2))) +

scale\_y\_continuous(name = 'Gut Concentration (mg/L)', breaks=(seq(0, 14, 1)))

for (i in 1:5) {

subject\_data = read.csv(

paste0('subject', i, 'table', questions[j], '.txt'),

col.names=c('t', 'y'),

header=FALSE

)

subject = rep(paste0('subject', i), times=nrow(subject\_data))

subject\_data = cbind(subject, subject\_data)

data = rbind(data, subject\_data)

real\_data <- read.csv(

paste0("subject", i, "\_experimental.txt"),

col.names = c('t', 'y'),

header=FALSE

)

real\_data$t = real\_data$t + 1 # offset by hour

p = p + geom\_point(data=real\_data, aes(x=t, y=y), color=hex[i])

}

p = p + geom\_line(data=data, aes(x=t, y=y, color=subject))

print(p)

ggsave(paste0("q1", questions[j], ".png"), bg="white")

}

Q2. Sensitivity analysis driver code; visualization code.

**Q2 SENS DRIVER CODE**

%% AUC for 14 hrs

%%% REQUIRES q1\_script to be run

subjects = 5;

delta = 0.1;

multiplier = 1 + delta;

local\_sens\_table = zeros(subjects, subjects, 3);

y0 = [0, 0, 175];

y\_subject=0;

for i = 1:3

for subject = 1:subjects

p = table\_1f(:, subject);

if i == 3

y\_subject = y\_real(:, subject);

end

% modify D1

local\_sens\_table(1, subject, i) = local\_sens( ...

i, ...

175\*multiplier, 310, p, ...

t\_real=t\_real, y\_real=y\_subject ...

);

% modify D2

local\_sens\_table(2, subject, i) = local\_sens( ...

i, ...

175, 310\*multiplier, p, ...

t\_real=t\_real, y\_real=y\_subject ...

);

% modify kcl

p\_kcl = p; p\_kcl(2) = p(2)\*multiplier;

local\_sens\_table(3, subject, i) = local\_sens( ...

i, ...

175, 310, p, p\_kcl, ...

t\_real=t\_real, y\_real=y\_subject ...

);

% modify Volume

p\_vd = p; p\_vd(3) = p(3)\*multiplier;

local\_sens\_table(4, subject, i) = local\_sens( ...

i, ...

175, 310, p, p\_vd, ...

t\_real=t\_real, y\_real=y\_subject ...

);

% modify ka

p\_ka = p; p\_ka(4) = p(4)\*multiplier;

local\_sens\_table(5, subject, i) = local\_sens( ...

i, ...

175, 310, p, p\_ka, ...

t\_real=t\_real, y\_real=y\_subject ...

);

end

end

FILE\_NAME = 'q2a'; writematrix(local\_sens\_table(:, :, 1), FILE\_NAME);

FILE\_NAME = 'q2b'; writematrix(local\_sens\_table(:, :, 2), FILE\_NAME);

FILE\_NAME = 'q2c'; writematrix(local\_sens\_table(:, :, 3), FILE\_NAME);

%%%%%%%%%% FUNCTIONS %%%%%%%%%%%%

function auc = auc14hrs(D1, D2, p)

tspan1 = 0:1/10:1;

tspan2 = 1:1/10:14;

y0 = [0, 0, D1];

[t1, y1] = ode45(@(t, y) CaffeineODE(t, y, p), tspan1, y0);

y0 = y1(end, :);

y0(end) = y0(end) + D2;

[t2, y2] = ode45(@(t, y) CaffeineODE(t, y, p), tspan2, y0);

auc = trapz([t1; t2], [y1(:, 1); y2(:, 1)]);

end

function auc = auc1hrs(D1, D2, p)

tspan1 = 0:1/10:1;

tspan2 = 1:1/10:2;

y0 = [0, 0, D1];

[t1, y1] = ode45(@(t, y) CaffeineODE(t, y, p), tspan1, y0);

y0 = y1(end, :);

y0(end) = y0(end) + D2;

[t2, y2] = ode45(@(t, y) CaffeineODE(t, y, p), tspan2, y0);

auc = trapz([t1; t2], [y1(:, 1); y2(:, 1)]);

end

function error = costfunc(t\_measured, y\_measured, D1, D2, p)

arguments

t\_measured (1, :) double

y\_measured (1, :) double

D1 double

D2 double

p (1, :) double

end

y0 = [0, 0, D1];

[~, y] = ode45(@(t, y) CaffeineODE(t, y, p), 0:1/10:1, y0);

y0 = y(end, :);

y0(end) = y0(end) + D2;

[t, y] = ode45(@(t, y) CaffeineODE(t, y, p), t\_measured, y0);

y\_interp = interp1(t, y(:, 1), t\_measured);

error = sum((y\_interp(:) - y\_measured(:)).^2);

end

function sensitivity = local\_sens(mode, D1, D2, p, pmod, options)

arguments

mode double

D1 double

D2 double

p (1, :) double

pmod (1, :) double = p

options.t\_real = 0

options.y\_real = 0

end

switch mode

case 1

out1 = auc14hrs(175, 310, p);

out2 = auc14hrs(D1, D2, pmod);

case 2

out1 = auc1hrs(175, 310, p);

out2 = auc1hrs(D1, D2, pmod);

case 3

out1 = costfunc(options.t\_real, options.y\_real, 175, 310, p);

out2 = costfunc(options.t\_real, options.y\_real, D1, D2, pmod);

end

numer = (out2 - out1) / out1;

denom = 0.1;

sensitivity = numer / denom;

end

**Q2 PLOTTING CODE**

setwd(Sys.getenv("PWD"))

library(RColorBrewer)

library(tidyr)

library(ggplot2)

sensitivities = c('14hr AUC', 'AUC 1hr after 11', 'Sum of Squares')

colnames = rep(c(paste('subject', 1:5)))

parameters = rep(c('D1', 'D2', 'kcl', 'Vd', 'ka'))

questions = c('a', 'b', 'c')

# make row 1 as col names

header.true <- function(df) {

names(df) <- as.character(unlist(df[1, ]))

df[-1, ]

}

for (i in 1:3) {

sens\_mat = as.data.frame(read.csv(

paste0("q2", questions[i], ".txt"),

header=FALSE

))

sens\_mat = rbind(colnames, sens\_mat)

sens\_mat = header.true(sens\_mat)

sens\_mat = cbind(parameters, sens\_mat)

sens\_mat = pivot\_longer(

sens\_mat,

!parameters,

names\_to='subject',

values\_to='sens'

)

sens\_mat$parameters = factor(sens\_mat$parameters, levels=rev(parameters))

print(sens\_mat)

sens\_mat$sens = as.numeric(sens\_mat$sens)

p = ggplot(data=sens\_mat, aes(x=subject, y=parameters, fill=sens)) +

labs(title=paste('Local Sensitivity for', sensitivities[i])) +

theme\_minimal() + geom\_tile() +

scale\_fill\_distiller(name='Sensitivity', palette='PiYG')

print(p)

ggsave(paste0("q2", questions[i], ".png"), bg="white")

}

Q3. Sensitivity analysis driver code; visualization code.

**Q3 SENS DRIVER CODE**

load\_dose = 20;

maintenance\_dose = 5;

Vd = 0.85;

step = 21;

weights = 0.45 \* linspace(3, 8, step);

V = Vd \* weights;

clearances = log(2) ./ linspace(50, 150, step);

ka = log(2) / (7/60);

y0 = [load\_dose, 0, 0];

[vv, cc] = meshgrid(V, clearances); %skip nested for loop

global\_sens1 = zeros(step);

global\_sens2 = zeros(step);

global\_sens3 = zeros(step);

for i = 1:numel(vv)

p = [0, cc(i), vv(i), 0];

global\_sens1(i) = global\_sens(1, y0, p);

global\_sens2(i) = global\_sens(2, y0, p);

global\_sens3(i) = global\_sens(3, y0, p);

end

FILE\_NAME = 'q3a'; writematrix(global\_sens1, FILE\_NAME);

FILE\_NAME = 'q3b'; writematrix(global\_sens2, FILE\_NAME);

FILE\_NAME = 'q3c'; writematrix(global\_sens3, FILE\_NAME);

%%

%%%% FUNCTIONS %%%%

function auc = auc\_first24(y0, parameters)

tspan = 0:1/10:24;

[t, y] = ode45( ...

@(t, y) CaffeineODE(t, y, parameters), ...

tspan, ...

y0...

);

auc = trapz(t, y(:, 1));

end

function auc = auc\_week(y0, parameters)

auc = 0;

for i=0:6 % woah, zero indexing and inclusive end??

[t, y] = ode45( ...

@(t, y) CaffeineODE(t, y, parameters), ...

24\*i:1/24:24\*(i+1), ...

y0 ...

);

auc = auc + trapz(t, y(:, 1));

y0 = y(end, :);

y0(1) = y0(1) + 5;

end

end

function ctrough = ctrough(y0, parameters)

[~, y] = ode45( ...

@(t, y) CaffeineODE(t, y, parameters), ...

0:1/10:24, ...

y0 ...

);

for i = 1:6

y0 = y(end, :);

y0(1) = y0(1) + 5;

[~, y] = ode45( ...

@(t, y) CaffeineODE(t, y, parameters), ...

24\*i:1/10:24\*(i+1), ...

y0 ...

);

end

ctrough = y(end, 1);

end

function sensitivity = global\_sens(mode, y0, parameters)

switch mode

case 1

sensitivity = auc\_first24(y0, parameters);

case 2

sensitivity = auc\_week(y0, parameters);

case 3

sensitivity = ctrough(y0, parameters);

end

end

**Q3 PLOTTING CODE**

setwd(Sys.getenv("PWD"))

library(RColorBrewer)

library(tidyr)

library(ggplot2)

sensitivities = c('24hr AUC', '7 day AUC', 'C\_trough')

colnames = rep(c(paste('subject', 1:5)))

questions = c('a', 'b', 'c')

linspace <- function(a, b) {

# a is array, b is num\_interval

seq(from=min(a), to=max(a), by=(max(a) - min(a))/b)

}

weights = linspace(3:8, 20)

clearances = linspace(50:150, 20)

header.true <- function(df) {

# move row1 to col names

names(df) <- as.character(unlist(df[1, ]))

df[-1, ]

}

for (i in 1:3) {

sens\_mat = as.data.frame(read.csv(

paste0("q3", questions[i], ".txt"),

header=FALSE

))

sens\_mat = rbind(weights, sens\_mat)

sens\_mat = header.true(sens\_mat)

sens\_mat = cbind(clearances, sens\_mat)

sens\_mat = pivot\_longer(

sens\_mat,

!clearances,

names\_to='weights',

values\_to='sens'

)

print(sens\_mat)

sens\_mat$sens = as.numeric(sens\_mat$sens)

p = ggplot(data=sens\_mat, aes(x=weights, y=clearances, fill=sens)) +

labs(title=paste('Global Sensitivity for', sensitivities[i]),

x = "weights (lbs)", y="clearance half-life") +

theme\_minimal() + geom\_tile() +

scale\_fill\_distiller(name='Sensitivity', palette='Spectral')

print(p)

ggsave(paste0("q3", questions[i], ".png"), bg="white")

}