Infliximab: Time-Weighted Bayesian Estimation

Jessica Wojciechowski

7 July 2016

```
# Set working directory
  project.dir <- "/Volumes/Prosecutor/PhD/InfliximabBayes/infliximab-bayes/Project/"
  n <- 12  # Number of seed individuals that were simulated
  nsim <- 200 # Number of simulations of seed individuals
  sim.output.dir <- paste0("SIM",nsim,"_IND",n)
  work.dir <- paste0(project.dir,sim.output.dir,"/")
  setwd(project.dir)

# Load package libraries
  library(ggplot2)  # Plotting
  library(grid)  # Plotting
  library(plyr)  # ddply function

# Custom ggplot2 theme
    theme_bw2 <- theme_set(theme_bw(base_size = 12))

# Pre-specified functions
  CI95lo <- function(x) quantile(x,probs = 0.025)  # Calculate 2.5th percentile
  CI95hi <- function(x) quantile(x,probs = 0.975)  # Calculate 97.5th percentile

  # Summary function for calculating median and prediction intervals
    summary.function <- function(x) {
      median <- median(x)
      stat.95lo <- CI95lo(x)
      stat.95hi <- CI95hi(x)
      result <- c(median,stat.95lo,stat.95hi)
      names(result)[c(1,2,3)] <- c("median","stat.95lo","stat.95hi")
      result
    }

  # Calculate the number of individuals below the target trough concentration at day 546
    below.target.546 <- function(x) {
      below.target.data <- subset(x,time == 546)
      below.target.data$BTT[below.target.data$IPRE < 3] <- 1
      below.target.data$BTT[below.target.data$IPRE >= 3] <- 0
      result <- data.frame(
                  BTT = sum(below.target.data$BTT),
                  n = length(below.target.data$BTT)
      )
      result
    }

# Pre-specified objects
  trough.target <- 3  # Don't want to be below the trough
  trough.upper <- 5  # Don't want concentrations to be too high - waste of drug
  infusion.times <- c(0,14,42,98,154,210,266,322,378,434,490,546)
  scale.log10.labels <- c(0.01,0.1,1,10,100,1000)
```

```
sample.time1 <- 98  # Sampling time at the end of interval 1
sample.time2 <- 210  # Sampling time at the end of interval 2
sample.time3 <- 378  # Sampling time at the end of interval 3
sample.time4 <- 546  # Sampling time at the end of interval 4
sample.times <- c(sample.time1,sample.time2,sample.time3,sample.time4)
```

Introduction: Time-Weighted Bayesian Estimation

Infliximab Pharmacokinetic Model

Population specifications

All patients weigh 70 kg. All patients have a baseline albumin of 4 g/dL.

There are 12 seed individuals:

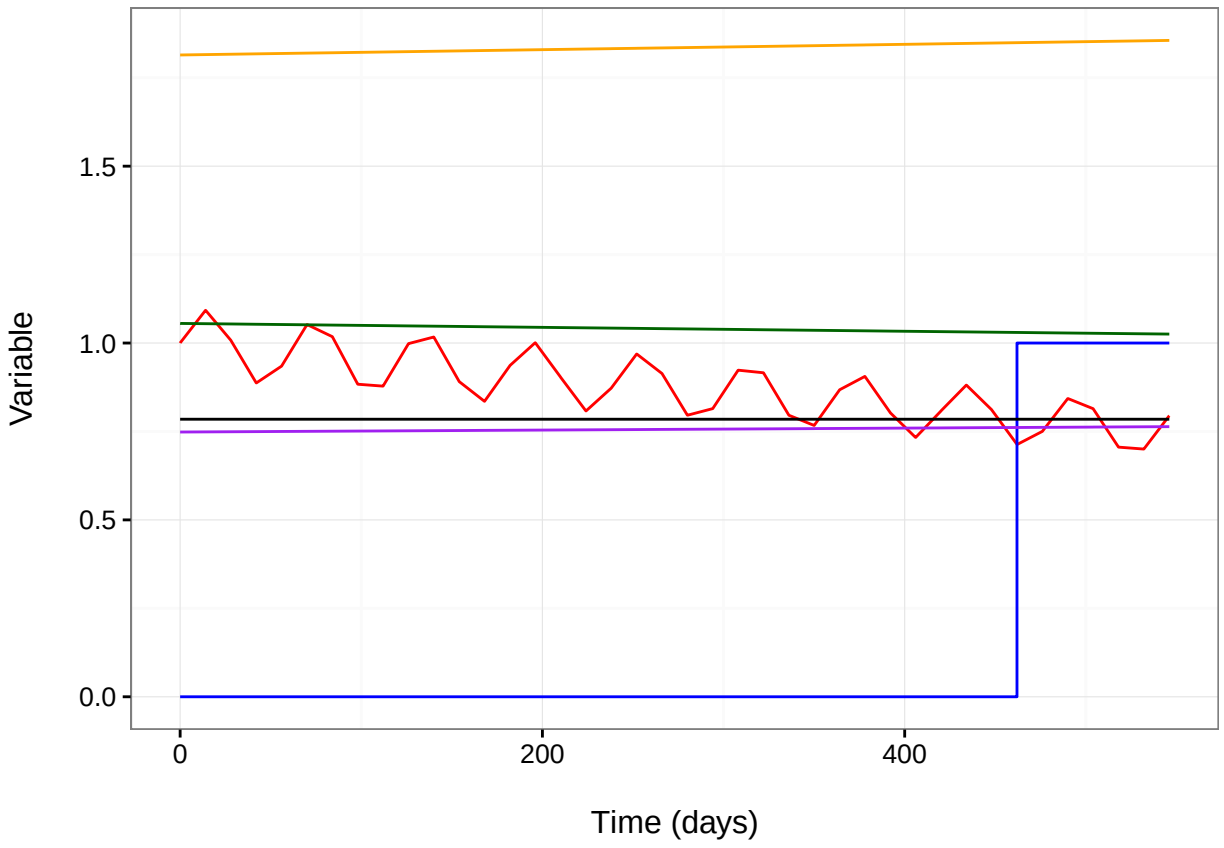
1. ADA onset = 154 days, final albumin = 3 g/dL
2. ADA onset = 154 days, final albumin = 4 g/dL
3. ADA onset = 154 days, final albumin = 5 g/dL
4. ADA onset = 294 days, final albumin = 3 g/dL
5. ADA onset = 294 days, final albumin = 4 g/dL
6. ADA onset = 294 days, final albumin = 5 g/dL
7. ADA onset = 462 days, final albumin = 3 g/dL
8. ADA onset = 462 days, final albumin = 4 g/dL
9. ADA onset = 462 days, final albumin = 5 g/dL
10. ADA onset = 646 days, final albumin = 3 g/dL
11. ADA onset = 646 days, final albumin = 4 g/dL
12. ADA onset = 646 days, final albumin = 5 g/dL

Between time = 0 (baseline) and time = 546 (final), patient albumin changes in a linear sine wave fashion (amplitude = 0.1, frequency = 60 days, phase = 0)

Random effect values for V1, Q and V2 change linearly between time = 0 and time = 546. CL random effect does not change as this parameter holds ADA and albumin covariate effects.

```
# Plot a single individual's characteristics over time
pop.data <- read.csv(paste0(work.dir,"population_characteristics.csv"))
pop.data1 <- pop.data[pop.data$SIM == 1 & pop.data$ID == 7,]

plotobj11 <- NULL
plotobj11 <- ggplot(pop.data1)
plotobj11 <- plotobj11 + geom_line(aes(x = TIME,y = ALB/4),colour = "red")
plotobj11 <- plotobj11 + geom_step(aes(x = TIME,y = ADA),colour = "blue")
plotobj11 <- plotobj11 + geom_line(aes(x = TIME,y = exp(ETA1)),colour = "black")
plotobj11 <- plotobj11 + geom_line(aes(x = TIME,y = exp(ETA2)),colour = "darkgreen")
plotobj11 <- plotobj11 + geom_line(aes(x = TIME,y = exp(ETA3)),colour = "purple")
plotobj11 <- plotobj11 + geom_line(aes(x = TIME,y = exp(ETA4)),colour = "orange")
plotobj11 <- plotobj11 + scale_x_continuous("\nTime (days)")
plotobj11 <- plotobj11 + scale_y_continuous("Variable\n")
print(plotobj11)
```



Dosing and Sampling Specifications

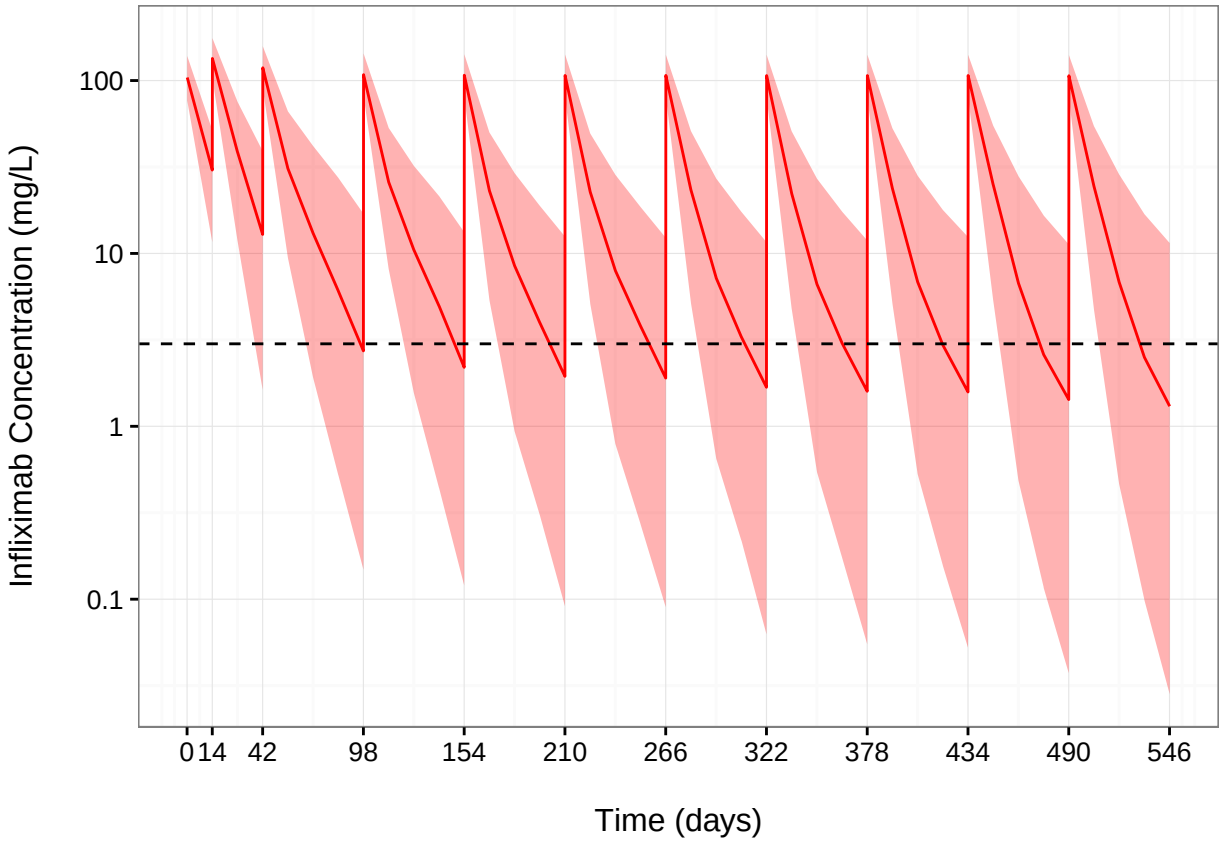
1. Simulations run from time = 0 to time = 546 days (approximately 18 months)
2. There are 4 sampling intervals:
    - Sampling interval 1; 0 to 98 days
    - Sampling interval 2; 98 to 210 days
    - Sampling interval 3; 210 to 378 days
    - Sampling interval 4; 378 to 546 days
3. Trough concentrations are sampled at the end of each sampling interval, i.e., time = 98, 210, and 378 days since first dose
    - No dosing adjustments are made at the conclusion of the study, i.e., day 546, therefore no trough concentration is sampled
4. There are 2 or 3 2-hour infliximab infusions administered in each sampling interval
    - Sampling interval 1; 0, 14 and 42 days
    - Sampling interval 2; 98 and 154 days
    - Sampling interval 3; 210, 266 and 322 days
    - Sampling interval 4; 378, 434 and 490 days

Label Scenario

Every patient receives a 5 mg/kg dose of infliximab at every infusion time (i.e., 350 mg infused over 2 hours)

```
# Source the results of label_simulation
  label.data <- read.csv(paste0(work.dir,"label_simulation.csv"))
  label.data <- label.data[label.data$SIM != 0,]  # Ignore the population typical individual simulation

# Plot the median and 95% prediction intervals for individual predicted concentrations (IPRE) over time
  plotobj1 <- NULL
  plotobj1 <- ggplot(label.data)
  plotobj1 <- plotobj1 + stat_summary(aes(x = time,y = IPRE),geom = "line",fun.y = median,colour = "red
  plotobj1 <- plotobj1 + stat_summary(aes(x = time,y = IPRE),geom = "ribbon",fun.ymin = "CI95lo",fun.yma
  plotobj1 <- plotobj1 + geom_hline(aes(yintercept = trough.target),linetype = "dashed")
  plotobj1 <- plotobj1 + scale_x_continuous("\nTime (days)",breaks = infusion.times)
  plotobj1 <- plotobj1 + scale_y_log10("Infliximab Concentration (mg/L)\n",breaks = scale.log10.labels,
  # plotobj1 <- plotobj1 + facet_wrap(~ID)  # Facet for seeds
  print(plotobj1)
```



For each individual, the time spent under the target trough concentration (3 mg/L) for the entire 546 day period was calculated.

```
# Calculate the time spent under the target trough at the end of each sampling interval
  label.data.AUT <- label.data[label.data$time %in% sample.times,]
  label.AUT.summary <- ddply(label.data.AUT, .(time), function(label.data.AUT) summary.function(label.da
  print(label.AUT.summary[label.AUT.summary$time == 546,])
```

```
##   time   median  stat.95lo stat.95hi
## 4  546 99.06022 0.00204839  336.4517
```

Number of individuals below the target trough concentration at day 546:

```
# Label data
  print(below.target.546(label.data))
```

```
##    BTT    n
## 1 1768 2400
```

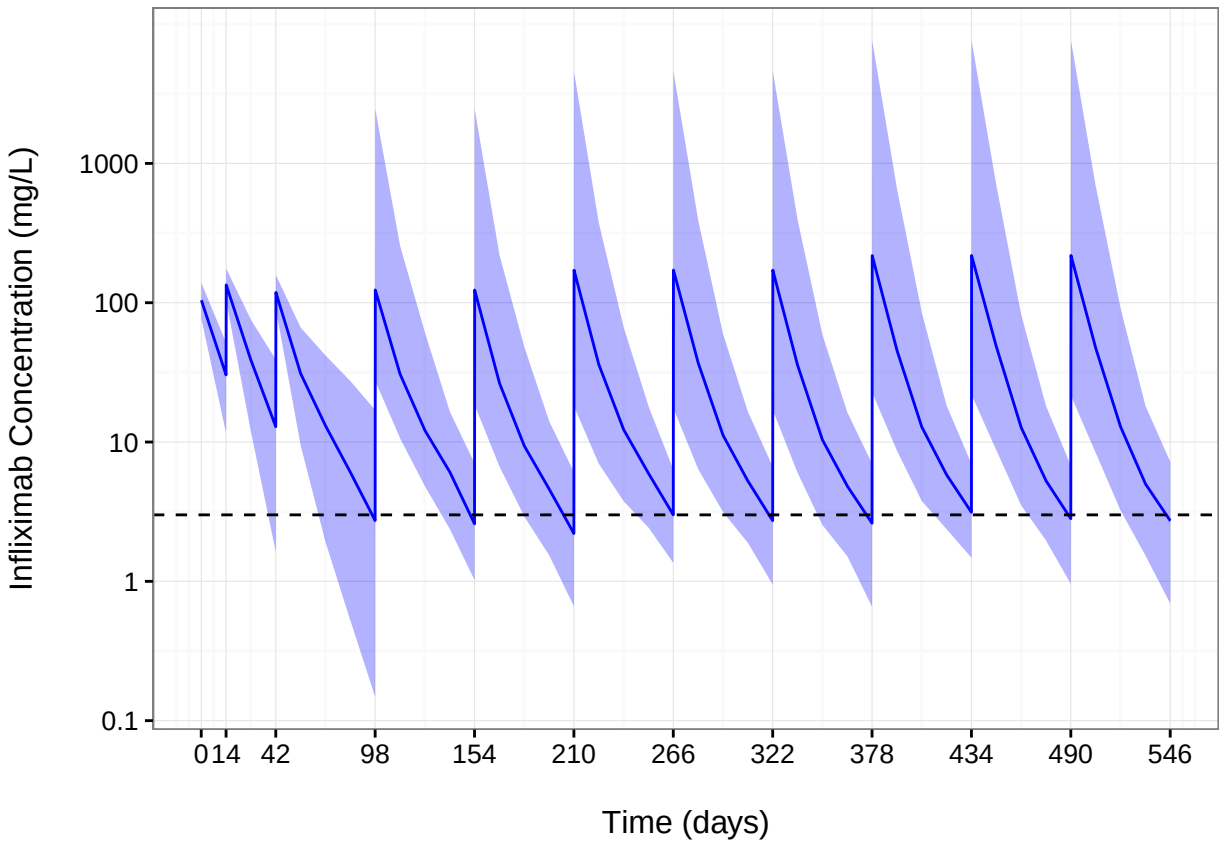
Clinical Scenario

Scenario where every patient receives a 5 mg/kg dose of infliximab for every infusion time in the first sampling interval (i.e., at 0, 14 and 42 days). Then:

- Trough concentration at time = 98 days is sampled (DV)
- Dose for the next sampling interval is determined as:
  - If a sample < trough target OR sample >= trough upper (5 mg/L) then the new dose will be equal to (trough target/sample)*previous dose
  - If a sample is within the pre-specified range, then the patient continues with the previous dose
- Cycle is repeated at sampling times = 210 and 378 days

```
# Source the results of clinical_simulation
  clinical.data <- read.csv(paste0(work.dir,"clinical_simulation.csv"))
  clinical.data <- clinical.data[clinical.data$SIM != 0,]  # Ignore the population typical individual s

# Plot the median and 95% prediction intervals for individual predicted concentrations (IPRE) over time
  plotobj2 <- NULL
  plotobj2 <- ggplot(clinical.data)
  plotobj2 <- plotobj2 + stat_summary(aes(x = time,y = IPRE),geom = "line",fun.y = median,colour = "blu
  plotobj2 <- plotobj2 + stat_summary(aes(x = time,y = IPRE),geom = "ribbon",fun.ymin = "CI95lo",fun.yma
  plotobj2 <- plotobj2 + geom_hline(aes(yintercept = trough.target),linetype = "dashed")
  plotobj2 <- plotobj2 + scale_x_continuous("\nTime (days)",breaks = infusion.times)
  plotobj2 <- plotobj2 + scale_y_log10("Infliximab Concentration (mg/L)\n",breaks = scale.log10.labels,l
  # plotobj2 <- plotobj2 + facet_wrap(~ID)  # Facet for seeds
  print(plotobj2)
```



We can see a significant improvement in the proportion of people who are achieving constant concentrations above the target trough (median line is predominantly above the line in the clinical scenario compared to the label scenario).

Calculate time spent under the target trough:

```
# Calculate the time spent under the target trough at the end of each sampling interval
  clinical.data.AUT <- clinical.data[clinical.data$time %in% sample.times,]
  clinical.AUT.summary <- ddply(clinical.data.AUT, .(time), function(clinical.data.AUT) summary.functio
  print(clinical.AUT.summary[clinical.AUT.summary$time == 546,])
```

```
##   time   median   stat.95lo stat.95hi
## 4  546 46.22288 0.002640362  157.7478
```

The median time spent under the target trough at time = 546 days in the clinical scenario is half compared to the label scenario.

Number of individuals below the target trough concentration at day 546:

```
# Clinical data
  print(below.target.546(clinical.data))
```

```
##    BTT    n
## 1 1377 2400
```

Bayesian Estimation and Dose Optimisation Scenarios

Scenario where every patient receives a 5 mg/kg dose of infliximab for every infusion time in the first sampling interval (i.e., at 0, 14 and 42 days). Then:

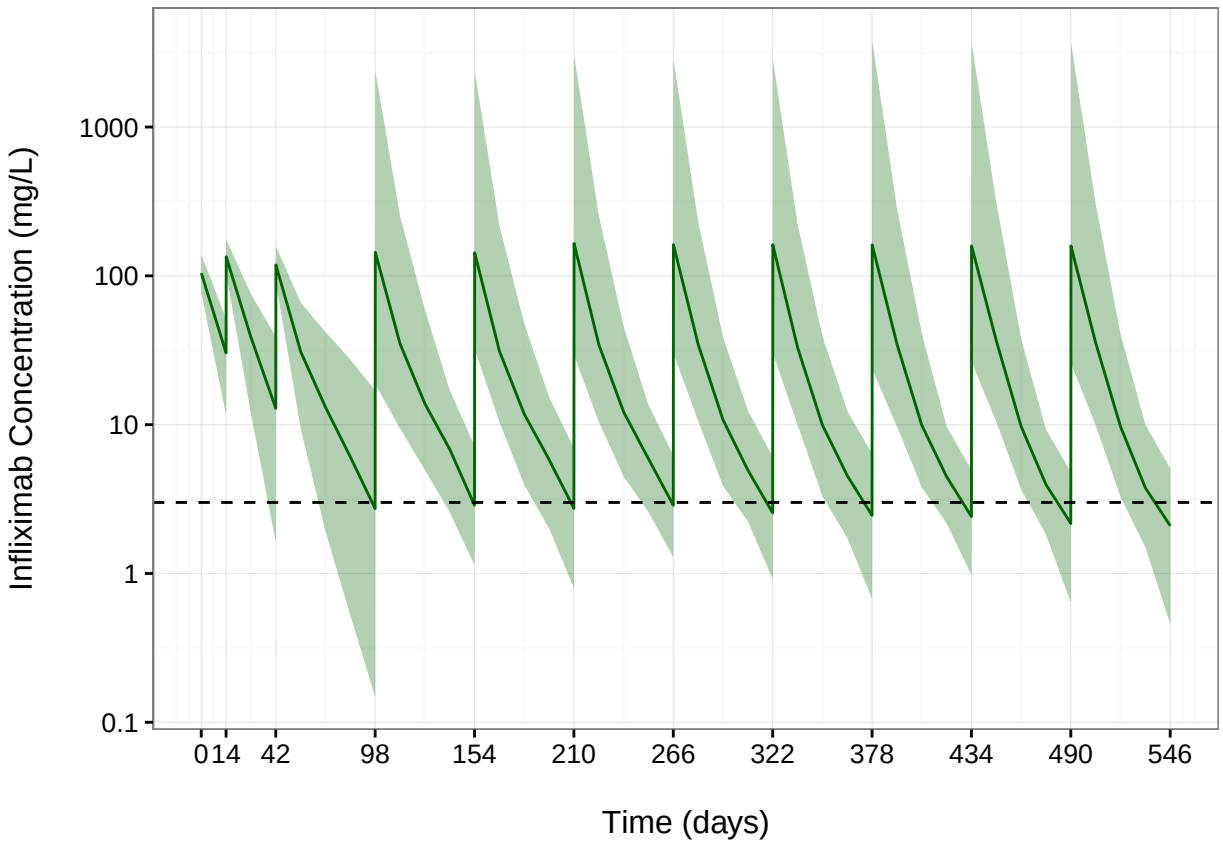
- Trough concentration at time = 98 days is sampled (DV)
- Individual values for pharmacokinetic parameters are obtained from empirical Bayes estimates using known administered doses, the sample, and covariate information at time = 0 and the sampling time
- Doses for the next sampling interval are determined by maximum likelihood estimation (minimise the error between trough concentration given optimised doses and the target trough concentration)
  - Optimised doses for the next sampling interval are allowed to be different
  - Initial estimates for doses are obtained by using the method described earlier in "Clinical Scenario"
  - If an initial estimate for a dose was greater than 1,000,000 mg - then doses were not optimised for the patient and they were just administered 1,000,000 mg
- Cycle is repeated at sampling times = 210 and 378 days
  - Bayes estimates obtained from subsequent sampling intervals are based on all previous samples, not just the most recent one
- When simulating the next interval given the individual parameters estimated from the previous interval, covariate values from the last sample are carried forward
  - We can predict the concentrations, however we cannot predict changes in their covariate profile

No Time-Weighting, All Covariate Information is available

All observations are weighted equally when maximising the posterior likelihood. Covariate information at the time of sampling as available.

```
# Source the results of bayes_simulation
  method1 <- "NTimeWeight"
  covariate1 <- "AllCov"
  bayes.method1 <- paste0(method1,covariate1)
  bayes.data1 <- read.csv(paste0(work.dir,"/",bayes.method1,"/",bayes.method1,"_optimise_bayes_simulati
  bayes.data1 <- bayes.data1[bayes.data1$SIM != 0,]  # Ignore the population typical individual simulat

# Plot the median and 95% prediction intervals for individual predicted concentrations (IPRE) over time
  plotobj3 <- NULL
  plotobj3 <- ggplot()
  plotobj3 <- plotobj3 + stat_summary(aes(x = time,y = IPRE),data = bayes.data1,geom = "line",fun.y = m
  plotobj3 <- plotobj3 + stat_summary(aes(x = time,y = IPRE),data = bayes.data1,geom = "ribbon",fun.ymi
  plotobj3 <- plotobj3 + geom_hline(aes(yintercept = trough.target),linetype = "dashed")
  plotobj3 <- plotobj3 + scale_x_continuous("\nTime (days)",breaks = infusion.times)
  plotobj3 <- plotobj3 + scale_y_log10("Infliximab Concentration (mg/L)\n",breaks = scale.log10.labels,
  print(plotobj3)
```



Calculate time spent under the target trough:

```
# Calculate the time spent under the target trough at the end of each sampling interval
  bayes.data.AUT1 <- bayes.data1[bayes.data1$time %in% sample.times,]
  bayes.AUT.summary1 <- ddply(bayes.data.AUT1, .(time), function(bayes.data.AUT1) summary.function(baye
  print(bayes.AUT.summary1[bayes.AUT.summary1$time == 546,])
```

```
##   time   median   stat.95lo stat.95hi
## 4  546 47.92876 0.002269087  174.6173
```

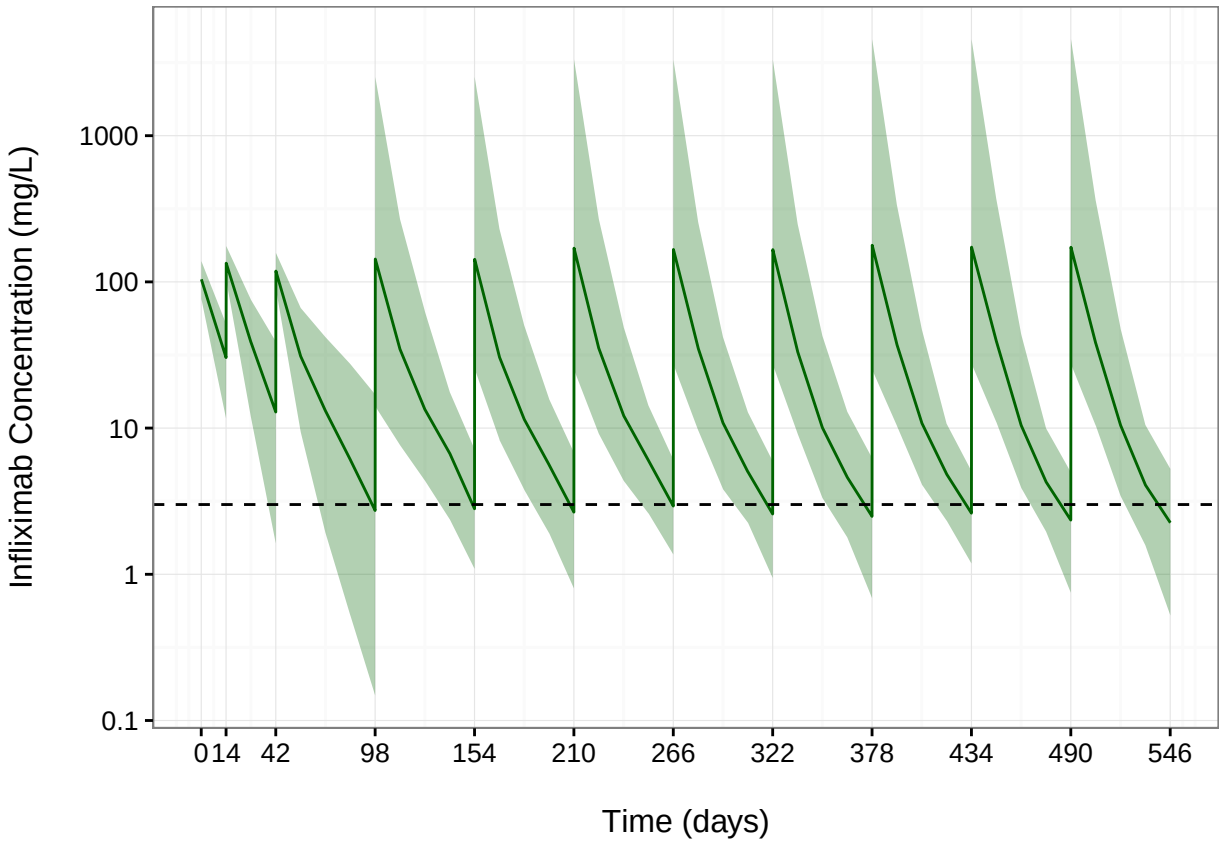
Peck 1.005 Time-Weighting, All Covariate Information is available

All observations are time-weighted using the Peck method with Q = 1.005 when maximising the posterior likelihood. Covariate information at the time of sampling as available.

```
# Source the results of bayes_simulation
  method2 <- "Peck1.005"
  covariate2 <- "AllCov"
  bayes.method2 <- paste0(method2,covariate2)
  bayes.data2 <- read.csv(paste0(work.dir,"/",bayes.method2,"/",bayes.method2,"_optimise_bayes_simulati
  bayes.data2 <- bayes.data2[bayes.data2$SIM != 0,]  # Ignore the population typical individual simulat

# Plot the median and 95% prediction intervals for individual predicted concentrations (IPRE) over time
  plotobj4 <- NULL
  plotobj4 <- ggplot()
  plotobj4 <- plotobj4 + stat_summary(aes(x = time,y = IPRE),data = bayes.data2,geom = "line",fun.y = me
```

```
  plotobj4 <- plotobj4 + stat_summary(aes(x = time,y = IPRE),data = bayes.data2,geom = "ribbon",fun.ymi
  plotobj4 <- plotobj4 + geom_hline(aes(yintercept = trough.target),linetype = "dashed")
  plotobj4 <- plotobj4 + scale_x_continuous("\nTime (days)",breaks = infusion.times)
  plotobj4 <- plotobj4 + scale_y_log10("Infliximab Concentration (mg/L)\n",breaks = scale.log10.labels,
  print(plotobj4)
```



Calculate time spent under the target trough:

```
# Calculate the time spent under the target trough at the end of each sampling interval
  bayes.data.AUT2 <- bayes.data2[bayes.data2$time %in% sample.times,]
  bayes.AUT.summary2 <- ddply(bayes.data.AUT2, .(time), function(bayes.data.AUT2) summary.function(baye
  print(bayes.AUT.summary2[bayes.AUT.summary2$time == 546,])
```

```
##   time   median   stat.95lo stat.95hi
## 4  546 45.23204 0.002254981   167.949
```

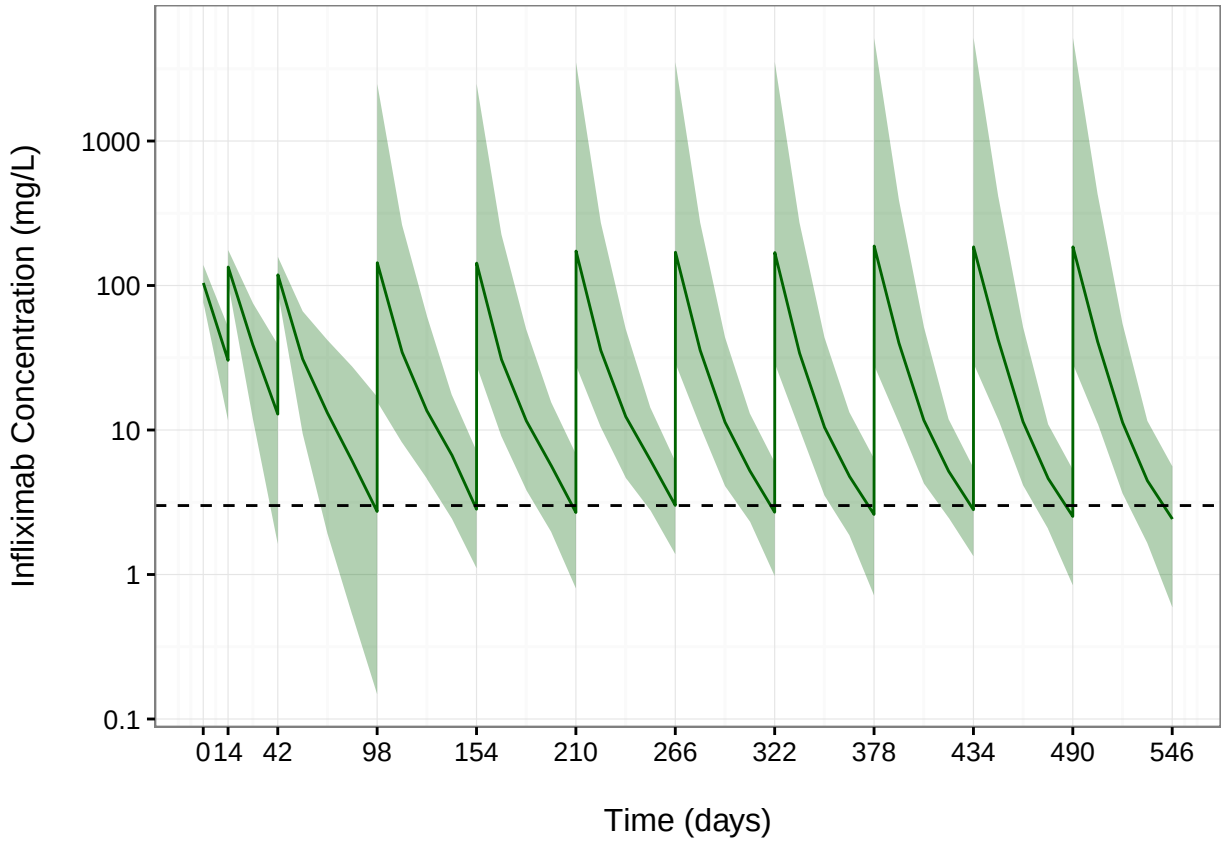
Peck 1.01 Time-Weighting, All Covariate Information is available

All observations are time-weighted using the Peck method with Q = 1.01 when maximising the posterior likelihood. Covariate information at the time of sampling as available.

```
# Source the results of bayes_simulation
  method3 <- "Peck1.01"
  covariate3 <- "AllCov"
  bayes.method3 <- paste0(method3,covariate3)
```

```
  bayes.data3 <- read.csv(paste0(work.dir,"/",bayes.method3,"/",bayes.method3,"_optimise_bayes_simulati
  bayes.data3 <- bayes.data3[bayes.data3$SIM != 0,]  # Ignore the population typical individual simulat

# Plot the median and 95% prediction intervals for individual predicted concentrations (IPRE) over time
  plotobj5 <- NULL
  plotobj5 <- ggplot()
  plotobj5 <- plotobj5 + stat_summary(aes(x = time,y = IPRE),data = bayes.data3,geom = "line",fun.y = me
  plotobj5 <- plotobj5 + stat_summary(aes(x = time,y = IPRE),data = bayes.data3,geom = "ribbon",fun.ymi
  plotobj5 <- plotobj5 + geom_hline(aes(yintercept = trough.target),linetype = "dashed")
  plotobj5 <- plotobj5 + scale_x_continuous("\nTime (days)",breaks = infusion.times)
  plotobj5 <- plotobj5 + scale_y_log10("Infliximab Concentration (mg/L)\n",breaks = scale.log10.labels,
  print(plotobj5)
```



Calculate time spent under the target trough:

```
# Calculate the time spent under the target trough at the end of each sampling interval
  bayes.data.AUT3 <- bayes.data3[bayes.data3$time %in% sample.times,]
  bayes.AUT.summary3 <- ddply(bayes.data.AUT3, .(time), function(bayes.data.AUT3) summary.function(baye
  print(bayes.AUT.summary3[bayes.AUT.summary3$time == 546,])
```

```
##   time  median   stat.95lo stat.95hi
## 4  546 40.6542 0.002246327  156.8893
```

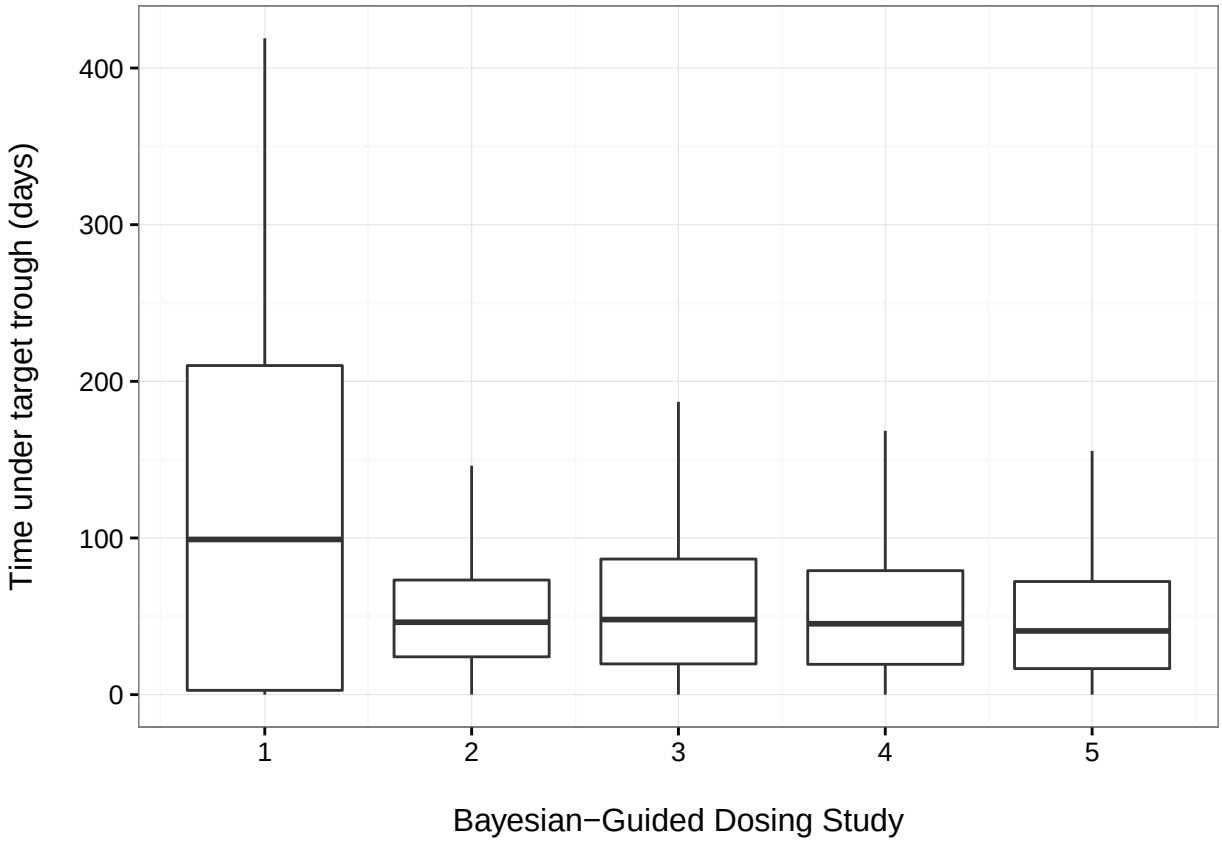
No Time-Weighting versus Time-Weight (Peck, Q = 1.005 and 1.01), All Covariate Information Available

Time under target trough concentration

```
label.data.AUT$SCEN <- 1
clinical.data.AUT$SCEN <- 2
bayes.data.AUT1$SCEN <- 3
bayes.data.AUT2$SCEN <- 4
bayes.data.AUT3$SCEN <- 5

AUT.data1 <- rbind(label.data.AUT,clinical.data.AUT[,-5],bayes.data.AUT1[,-5],bayes.data.AUT2[,-5],ba

plotobj6 <- NULL
plotobj6 <- ggplot(AUT.data1[AUT.data1$time == 546,])
plotobj6 <- plotobj6 + geom_boxplot(aes(x = SCEN,y = AUT,group = SCEN),outlier.shape = NA)
plotobj6 <- plotobj6 + scale_y_continuous("Time under target trough (days)\n",lim = c(NA,NA))
plotobj6 <- plotobj6 + scale_x_continuous("\nBayesian-Guided Dosing Study")
print(plotobj6)
```



Number of people below target trough concentration

```
# Bayes data - no time-weighting
  print(below.target.546(bayes.data1))
```

```
##    BTT    n
## 1 1860 2400
```

```
# Bayes data - Peck, 1.005
  print(below.target.546(bayes.data2))
```

```
##    BTT    n
## 1 1749 2400
```

```
# Bayes data - Peck, 1.01
  print(below.target.546(bayes.data3))
```

```
##    BTT    n
## 1 1604 2400
```