PKPD Shiny App Validation

Victor Poon

2023-06-02

Contents

1 Overview

The key simulations performed in the app for validation purpose include the following: Case 1: population simulation based on the simulated covariates (1000 virtual individuals), the simulated dosing regimen and the user-defined sampling times; set seed to for randomization control; Case 2: population simulation based on the actual covariates (sampling with replacement from the 477 individuals to generate 1000 individuals), the simulated dosing regimen and the user-defined sampling time; set seed to for randomization control; Case 3: case 2 with simulation of typical profile (PRED); Case 4: case 2 filtered by cycle 8 and stratified by ALBUbin; Case 5: individual simulation of the actual patients (n = 477) based on the post-hoc parameters, the individual dosing history and individual sampling time augmented with the fine time grid.

2 Read in Data and Model

2.1 Mrgsolve Models and Dose Data

```
mod.pop <- mread(model="Pop",
                 file="../PKcase_Perjeta_valid.cpp")
```

```
## Building Pop ... done.
```

```
mod.indiv <- mread(model="Indiv",
            file="../mrgsolve_ind/Perjeta.ind.cov.cpp")
```

```
## Building Indiv ... done.
```

```
base.dose<-read_csv("Base.csv") %>%
  mutate(time=ifelse(time==0,1.00e-10,time))
```

```
## Rows: 8 Columns: 5
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (5): amt, evid, cmt, time, rate
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

2.2 Read in Seed

```
my.seed<-read_csv("AppData/Case 1/randomSeed.csv")$`Random Seed`
```

```
## Rows: 1 Columns: 1
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (1): Random Seed
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

2.3 Case 1 Data

2.3.1 Covariate

```
case1.covar.app<-read_csv("AppData/Case 1/SimulatedData.csv") %>%
  select(ID,ALBU,LBW) %>%
  distinct()
```

```
## Warning: One or more parsing issues, see `problems()` for details
```

```
## Rows: 194000 Columns: 19
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (3): SIM_TYPE, REC_TYPE, SIM_ID
## dbl (15): USUBJID, ID, time, EVID, AMT, period, ALBU, LBW, CENT, PERIPH, IPR...
## lgl  (1): covar_propo
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
# Script generated results matches exactly with app covariate distribution
set.seed(my.seed)
case1.covar<-tmvtnorm::rtmvnorm(
        n = 1000,
        mean = c(3.9,51.3),
        sigma = diag(c(.5^2,9.5^2)),
        lower = c(2.1,31),
        upper =c(5.3,82)
      ) %>%
  as_tibble() %>%
  rename(ALBU=V1,LBW=V2)
```

```
## Warning: The 'x' argument of 'as_tibble.matrix()' must have unique column names if
## '.name_repair' is omitted as of tibble 2.0.0.
## i Using compatibility '.name_repair'.
## This warning is displayed once every 8 hours.
## Call 'lifecycle::last_lifecycle_warnings()' to see where this warning was
## generated.
```

2.3.1.1 Check Matching of Covariate Distributions

```
summary(case1.covar.app[-1])
```

2.3.1.1.1 App

```
##       ALBU            LBW
##  Min.   :2.490   Min.   :31.12
##  1st Qu.:3.582   1st Qu.:46.07
##  Median :3.910   Median :52.22
##  Mean   :3.902   Mean   :52.38
##  3rd Qu.:4.252   3rd Qu.:58.47
##  Max.   :5.257   Max.   :81.85
```

```
summary(case1.covar)
```

2.3.1.1.2 Validation

```
##       ALBU            LBW
##  Min.   :2.490   Min.   :31.12
##  1st Qu.:3.582   1st Qu.:46.07
##  Median :3.910   Median :52.22
##  Mean   :3.902   Mean   :52.38
##  3rd Qu.:4.252   3rd Qu.:58.47
##  Max.   :5.257   Max.   :81.85
```

2.3.2 Simulation Data from App

```
# Case 1 App Simulations
case1_simdata.app<-read_csv("AppData/Case 1/simulatedData.csv") %>%
  mutate(time=round(time,5))
```

```
## Warning: One or more parsing issues, see `problems()` for details
```

```
## Rows: 194000 Columns: 19
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (3): SIM_TYPE, REC_TYPE, SIM_ID
## dbl (15): USUBJID, ID, time, EVID, AMT, period, ALBU, LBW, CENT, PERIPH, IPR...
## lgl  (1): covar_propo
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
case1_obdata<-case1_simdata.app %>%
  filter(REC_TYPE=="Observation") %>%
  select(time) %>%
  distinct() %>%
  filter(!time %in% seq(0,175,by=1))

# Extra Dosing Data on top of daily sampling
case1_addtime<-case1_obdata$time
```

2.4 Case 2 Data

2.4.1 Covariate

```
actual.population<-read_csv("../mrgsolve_ind/population.data.indpara.csv") %>%
  rename(ID=USUBJID) %>%
  select(-time)
```

```
## Rows: 477 Columns: 8
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (8): USUBJID, time, CLind, V1ind, Qind, V2ind, ALBU, LBW
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
case2.covar.app<-read_csv("AppData/Case 2/SimulatedData.csv") %>%
  select(ID,ALBU,LBW) %>%
  distinct()
```

```
## Warning: One or more parsing issues, see `problems()` for details
```

```
## Rows: 194000 Columns: 21
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (5): SIM_TYPE, REC_TYPE, SIM_ID, ALBU (Bins), LBW (Bins)
## dbl (15): USUBJID, ID, time, EVID, AMT, period, ALBU, LBW, CENT, PERIPH, IPR...
## lgl  (1): covar_propo
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
# Matches exactly
set.seed(my.seed)
case2.covar.R<-actual.population %>%
  sample_n(size=1000,replace=T) %>%
  mutate(ID=cur_group_rows())
#
```

2.4.1.1 Check Matching of Covariate Distributions

```
summary(case2.covar.app[-1])
```

2.4.1.1.1 App

```
##       ALBU           LBW
##  Min.   :2.10   Min.   :33.16
##  1st Qu.:3.50   1st Qu.:44.63
##  Median :3.90   Median :48.77
##  Mean   :3.85   Mean   :51.72
##  3rd Qu.:4.20   3rd Qu.:58.02
##  Max.   :5.30   Max.   :82.25
```

```
summary(case2.covar.R %>% select(ALBU,LBW))
```

2.4.1.1.2 Validation

```
##       ALBU           LBW
##  Min.   :2.10   Min.   :33.16
##  1st Qu.:3.50   1st Qu.:44.63
##  Median :3.90   Median :48.77
##  Mean   :3.85   Mean   :51.72
##  3rd Qu.:4.20   3rd Qu.:58.02
##  Max.   :5.30   Max.   :82.25
```

2.4.2 Simulation Data from App

```r
# Case 2 App Simulations
case2_simdata.app<-read_csv("AppData/Case 2/simulatedData.csv") %>%
  mutate(time=round(time,5))
```

```
## Warning: One or more parsing issues, see ‘problems()‘ for details
```

```
## Rows: 194000 Columns: 21
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (5): SIM_TYPE, REC_TYPE, SIM_ID, ALBU (Bins), LBW (Bins)
## dbl (15): USUBJID, ID, time, EVID, AMT, period, ALBU, LBW, CENT, PERIPH, IPR...
## lgl  (1): covar_propo
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

```r
case2_obdata<-case2_simdata.app %>%
  filter(REC_TYPE=="Observation") %>%
  select(time) %>%
  distinct() %>%
  filter(!time %in% seq(0,175,by=1))

# Extra Dose Data
case2_addtime<-case2_obdata$time
case2_dosedata<-case2_simdata.app %>%
  filter(REC_TYPE=="Dose")
```

2.5 Case 3 Data

2.5.1 Covariate

```r
actual.population<-read_csv("../mrgsolve_ind/population.data.indpara.csv") %>%
  rename(ID=USUBJID) %>%
  select(-time)
```

```
## Rows: 477 Columns: 8
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (8): USUBJID, time, CLind, V1ind, Qind, V2ind, ALBU, LBW
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

```r
case3.covar.app<-read_csv("AppData/Case 3/SimulatedData.csv") %>%
  select(ID,ALBU,LBW) %>%
  distinct()
```

```
## Warning: One or more parsing issues, see ‘problems()‘ for details
```

```
## Rows: 194000 Columns: 21
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (5): SIM_TYPE, REC_TYPE, SIM_ID, ALBU (Bins), LBW (Bins)
## dbl (15): USUBJID, ID, time, EVID, AMT, period, ALBU, LBW, CENT, PERIPH, IPR...
## lgl  (1): covar_propo
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
# Matches exactly
set.seed(my.seed)
case3.covar.R<-actual.population %>%
  sample_n(size=1000,replace=T) %>%
  mutate(ID=cur_group_rows())
```

2.5.1.1 Check Matching of Covariate Distributions

```
summary(case3.covar.app[-1])
```

2.5.1.1.1 App

```
##       ALBU           LBW
##  Min.   :2.10   Min.   :33.16
##  1st Qu.:3.50   1st Qu.:44.63
##  Median :3.90   Median :48.77
##  Mean   :3.85   Mean   :51.72
##  3rd Qu.:4.20   3rd Qu.:58.02
##  Max.   :5.30   Max.   :82.25
```

```
summary(case3.covar.R %>% select(ALBU,LBW))
```

2.5.1.1.2 Validation

```
##       ALBU           LBW
##  Min.   :2.10   Min.   :33.16
##  1st Qu.:3.50   1st Qu.:44.63
##  Median :3.90   Median :48.77
##  Mean   :3.85   Mean   :51.72
##  3rd Qu.:4.20   3rd Qu.:58.02
##  Max.   :5.30   Max.   :82.25
```

2.5.2 Simulation Data from App

```
# Case 3 App Simulations
case3_simdata.app<-read_csv("AppData/Case 3/simulatedData.csv") %>%
  mutate(time=round(time,5))
```

```
## Warning: One or more parsing issues, see ‘problems()‘ for details
```

```
## Rows: 194000 Columns: 21
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (5): SIM_TYPE, REC_TYPE, SIM_ID, ALBU (Bins), LBW (Bins)
## dbl (15): USUBJID, ID, time, EVID, AMT, period, ALBU, LBW, CENT, PERIPH, IPR...
## lgl  (1): covar_propo
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

```
case3_obdata<-case3_simdata.app %>%
  filter(REC_TYPE=="Observation") %>%
  select(time) %>%
  distinct() %>%
  filter(!time %in% seq(0,175,by=1))

# Extra Dose Data
case3_addtime<-case3_obdata$time
case3_dosedata<-case3_simdata.app %>%
  filter(REC_TYPE=="Dose")
```

2.6 Case 4 Data

2.6.1 Covariate

```
actual.population<-read_csv("../mrgsolve_ind/population.data.indpara.csv") %>%
  rename(ID=USUBJID) %>%
  select(-time)
```

```
## Rows: 477 Columns: 8
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (8): USUBJID, time, CLind, V1ind, Qind, V2ind, ALBU, LBW
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

```
case4.covar.app<-read_csv("AppData/Case 4/SimulatedData.csv") %>%
  select(ID,ALBU,LBW,`ALBU (Bins)`) %>%
  distinct()
```

```
## Warning: One or more parsing issues, see ‘problems()‘ for details
```

```
## Rows: 194000 Columns: 21
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (5): SIM_TYPE, REC_TYPE, SIM_ID, ALBU (Bins), LBW (Bins)
## dbl (15): USUBJID, ID, time, EVID, AMT, period, ALBU, LBW, CENT, PERIPH, IPR...
## lgl  (1): covar_propo
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
# Matches exactly
set.seed(my.seed)
case4.covar.R<-actual.population %>%
  sample_n(size=1000,replace=T) %>%
  mutate(ID=cur_group_rows()) %>%
  mutate(ALBIN=as.factor(cut(ALBU,
                             breaks=c(2.1,3.5,3.9,4.2,5.3),
                             include.lowest=TRUE))) %>%
  mutate(ALBIN=as.numeric(ALBIN))
summary(case4.covar.R$ALBIN)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.000   1.000   2.000   2.421   3.000   4.000
```

```
# Bins app:
# "(2.1,3.5]" "(3.5,3.9]" "(3.9,4.2]" "(4.2,5.3]"
```

2.6.1.1 Check Matching of Covariate Distributions

```
summary(case4.covar.app[-1])
```

2.6.1.1.1 App

```
##       ALBU           LBW        ALBU (Bins)
##  Min.   :2.10   Min.   :33.16   Length:1000
##  1st Qu.:3.50   1st Qu.:44.63   Class :character
##  Median :3.90   Median :48.77   Mode  :character
##  Mean   :3.85   Mean   :51.72
##  3rd Qu.:4.20   3rd Qu.:58.02
##  Max.   :5.30   Max.   :82.25
```

```
table(case4.covar.app$`ALBU (Bins)`)
```

```
##
## (3.5,3.9] (3.9,4.2] (4.2,5.3] [2.1,3.5]
##       291       235       220       254
```

```
summary(case4.covar.R %>% select(ALBU,LBW))
```

2.6.1.1.2 Validation

```
##       ALBU           LBW
##  Min.   :2.10   Min.   :33.16
##  1st Qu.:3.50   1st Qu.:44.63
##  Median :3.90   Median :48.77
##  Mean   :3.85   Mean   :51.72
##  3rd Qu.:4.20   3rd Qu.:58.02
##  Max.   :5.30   Max.   :82.25
```

```
table(case4.covar.R$ALBIN)
```

```
##
##   1   2   3   4
## 254 291 235 220
```

2.6.2 Simulation Data from App

```
# Case 4 App Simulations
case4_simdata.app<-read_csv("AppData/Case 4/simulatedData.csv") %>%
  mutate(time=round(time,5))
```

```
## Warning: One or more parsing issues, see `problems()` for details
```

```
## Rows: 194000 Columns: 21
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (5): SIM_TYPE, REC_TYPE, SIM_ID, ALBU (Bins), LBW (Bins)
## dbl (15): USUBJID, ID, time, EVID, AMT, period, ALBU, LBW, CENT, PERIPH, IPR...
## lgl  (1): covar_propo
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
case4_obdata<-case4_simdata.app %>%
  filter(REC_TYPE=="Observation") %>%
  select(time) %>%
  distinct() %>%
  filter(!time %in% seq(0,175,by=1))

# Extra Dose Data
case4_addtime<-case4_obdata$time
case4_dosedata<-case4_simdata.app %>%
  filter(REC_TYPE=="Dose")
```

2.7 Case 5 Data

2.7.1 Individual Parameter, Sampling and Dosing Time

```
ind.covar<-read_csv("../mrgsolve_ind/population.data.indpara.csv") %>%
  rename(ID=USUBJID) %>%
  select(-time)
```

```
## Rows: 477 Columns: 8
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (8): USUBJID, time, CLind, V1ind, Qind, V2ind, ALBU, LBW
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

2.7.2 Simulation Data from App

```
# For Mrgsolve
case5_simdata<-read_csv("AppData/Case 5/simulatedData.csv")
```

```
## Rows: 68172 Columns: 21
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (4): SIM_TYPE, REC_TYPE, SIM_ID, covar_propo
## dbl (17): USUBJID, ID, time, EVID, AMT, period, CLind, Qind, V1ind, V2ind, C...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
# Case 5 App Simulations
case5_simdata.app<-read_csv("AppData/Case 5/simulatedData.csv") %>%
  mutate(time=round(time,5))
```

```
## Rows: 68172 Columns: 21
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (4): SIM_TYPE, REC_TYPE, SIM_ID, covar_propo
## dbl (17): USUBJID, ID, time, EVID, AMT, period, CLind, Qind, V1ind, V2ind, C...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

2.8 App statistics

```
col_split<-function(dset,preffix,slice=1){
  out<-str_split_fixed(dset,"<br>",8) %>%
```

```
    as_tibble() %>%
    mutate(Mean=str_split(V1, ":",simplify = T)[,2],
           SD=str_split(V2, ":",simplify = T)[,2],
           p50=str_split(V3, ":",simplify = T)[,2],
           CV=str_split(V4, ":",simplify = T)[,2],
           p5=str_split(V5, ":",simplify = T)[,2],
           p95=str_split(V6, ":",simplify = T)[,2],
           GM=str_split(V7, ":",simplify = T)[,2],
           GCV=str_split(V8, ":",simplify = T)[,2]) %>%
    select(c(9:16)) %>%
    mutate(across(.fns=as.numeric)) %>%
    rename_with(~ paste0(preffix,"_", .))

    if(slice==1) {out %>% slice(-1)
    } else out
}

sep_app_data<-function(path,type=1){
  app.stats<-read_csv(path)

  if(type==1){
    AUC<-col_split(app.stats$`AUC Stats`,"AUC")
    CMIN<-col_split(app.stats$`LAST Stats`,"LAST")
    CLAST<-col_split(app.stats$`MAX Stats`,"MAX")
    THRES.app<-app.stats$`Percent Above LAST Threshold (20)`[-1]
  } else {
    AUC<-col_split(app.stats$`AUC Stats`,"AUC",0)
    CMIN<-col_split(app.stats$`LAST Stats`,"LAST",0)
    CLAST<-col_split(app.stats$`MAX Stats`,"MAX",0)
    THRES.app<-app.stats$`Percent Above LAST Threshold (20)`
  }


  out<-AUC %>%
        bind_cols(CMIN) %>%
        bind_cols(CLAST) %>%
        mutate(THRES=THRES.app)

  if(type==1){
    out %>% mutate(Cycle=1:8)%>%
        mutate(THRES=as.numeric(substr(THRES,1,4)))
  } else {
    out %>% mutate(ALBIN=1:4)
  }

}

case1.app.stats<-sep_app_data("AppData/Case 1/summaryStatsAveragesTable.csv")
```

```
## Rows: 9 Columns: 5
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (5): Summarize by, Percent Above LAST Threshold (20), AUC Stats, LAST St...
```

```
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
case2.app.stats<-sep_app_data("AppData/Case 2/summaryStatsAveragesTable.csv")
```

```
## Rows: 9 Columns: 5
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (5): Summarize by, Percent Above LAST Threshold (20), AUC Stats, LAST St...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
case3.app.stats<-sep_app_data("AppData/Case 3/summaryStatsAveragesTable.csv")
```

```
## Rows: 9 Columns: 5
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (5): Summarize by, Percent Above LAST Threshold (20), AUC Stats, LAST St...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.

## Warning in mask$eval_all_mutate(quo): NAs introduced by coercion
```

```
case4.app.stats<-sep_app_data("AppData/Case 4/summaryStatsAveragesTable.csv",
                              type=2) %>%
  mutate(THRES=as.numeric(str_replace(THRES,"%","")))
```

```
## Rows: 4 Columns: 5
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (5): Summarize by, Percent Above LAST Threshold (20), AUC Stats, LAST St...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

3 Perform Simulations

3.1 Case 1

```
dose.dat<- base.dose %>% as.ev()

set.seed(my.seed)
case1.sim.dat<-mod.pop %>%
  idata_set(case1.covar) %>%
  carry_out(evid) %>%
```

```
  mrgsim(events=dose.dat,end=168,add=case1_addtime) %>%
  as_tibble() %>%
  filter(evid==0) %>%
  mutate(
    Cycle=ceil(time/21)
  ) %>%
  filter(time!=0)
```

3.2 Case 2

```
dose.dat<- base.dose %>% as.ev()

set.seed(my.seed)
case2.sim.dat<-mod.pop %>%
  # zero_re() %>%
  idata_set(case2.covar.R) %>%
  mrgsim(events=dose.dat,start=1,end=175,add=case2_addtime) %>%
  as_tibble() %>%
  distinct() %>%
  mutate(
    Cycle=ceil(time/21)
  )
```

3.3 Case 3

```
dose.dat<- base.dose %>% as.ev()

set.seed(my.seed)
case3.sim.dat<-mod.pop %>%
  zero_re() %>%
  idata_set(case3.covar.R) %>%
  mrgsim(events=dose.dat,start=1,end=175,add=case3_addtime) %>%
  as_tibble() %>%
  distinct() %>%
  mutate(
    Cycle=ceil(time/21)
  )
```

3.4 Case 4

```
dose.dat<- base.dose %>% as.ev()

set.seed(my.seed)
case4.sim.dat<-mod.pop %>%
  # zero_re() %>%
  idata_set(case4.covar.R) %>%
  carry_out(ALBU,ALBIN) %>%
```

```
  mrgsim(events=dose.dat,start=1,end=175,add=case4_addtime) %>%
  as_tibble() %>%
  distinct() %>%
  mutate(
    Cycle=ceil(time/21)
  )
```

3.5 Case 5

```
case5.sim.input<-case5_simdata %>%
  select(ID,time,EVID,AMT,CLind,Qind,V1ind,V2ind) %>%
  mutate(cmt=1) %>%
  rename(evid=EVID,amt=AMT)

set.seed(my.seed)
case5.sim.dat<-mod.indiv %>%
  data_set(case5.sim.input) %>%
  carry_out(evid) %>%
  mrgsim() %>%
  as_tibble() %>%
  distinct()  %>%
  mutate(
    Cycle=ceil(time/21)
  )
```

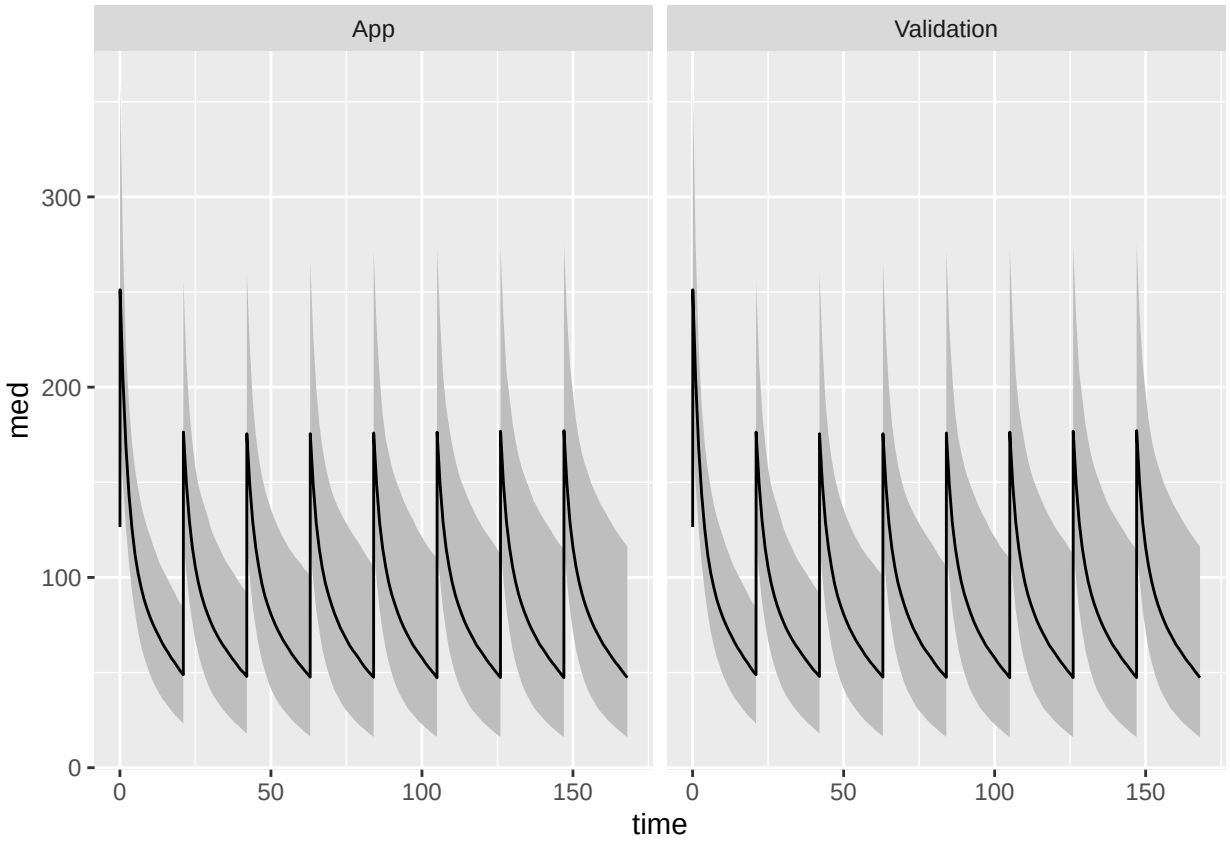
4 Plot (Validation vs App)

```
plot.compare<-function(R.dat,app.dat){
  R.plot<-R.dat %>% select(ID,time,IPREDnormal) %>%
    group_by(time) %>%
    dplyr::summarise(med=median(IPREDnormal),
              p5=quantile(IPREDnormal,p=0.05),
              p95=quantile(IPREDnormal,p=0.95)) %>%
    mutate(type="Validation")

  App.plot<-app.dat %>%
    filter(REC_TYPE=="Observation",time!=0) %>%
    select(ID,time,IPREDnormal) %>%
    group_by(time) %>%
    dplyr::summarise(med=median(IPREDnormal),
              p5=quantile(IPREDnormal,p=0.05),
              p95=quantile(IPREDnormal,p=0.95)) %>%
    mutate(type="App")

  summary_dat<-R.plot %>%
    rbind(App.plot)

  ggplot(summary_dat,aes(time))+
    geom_ribbon(aes(ymin=p5,ymax=p95),fill="grey")+
```

```
    geom_line(aes(y=med)) +
    facet_wrap("type")
}
```

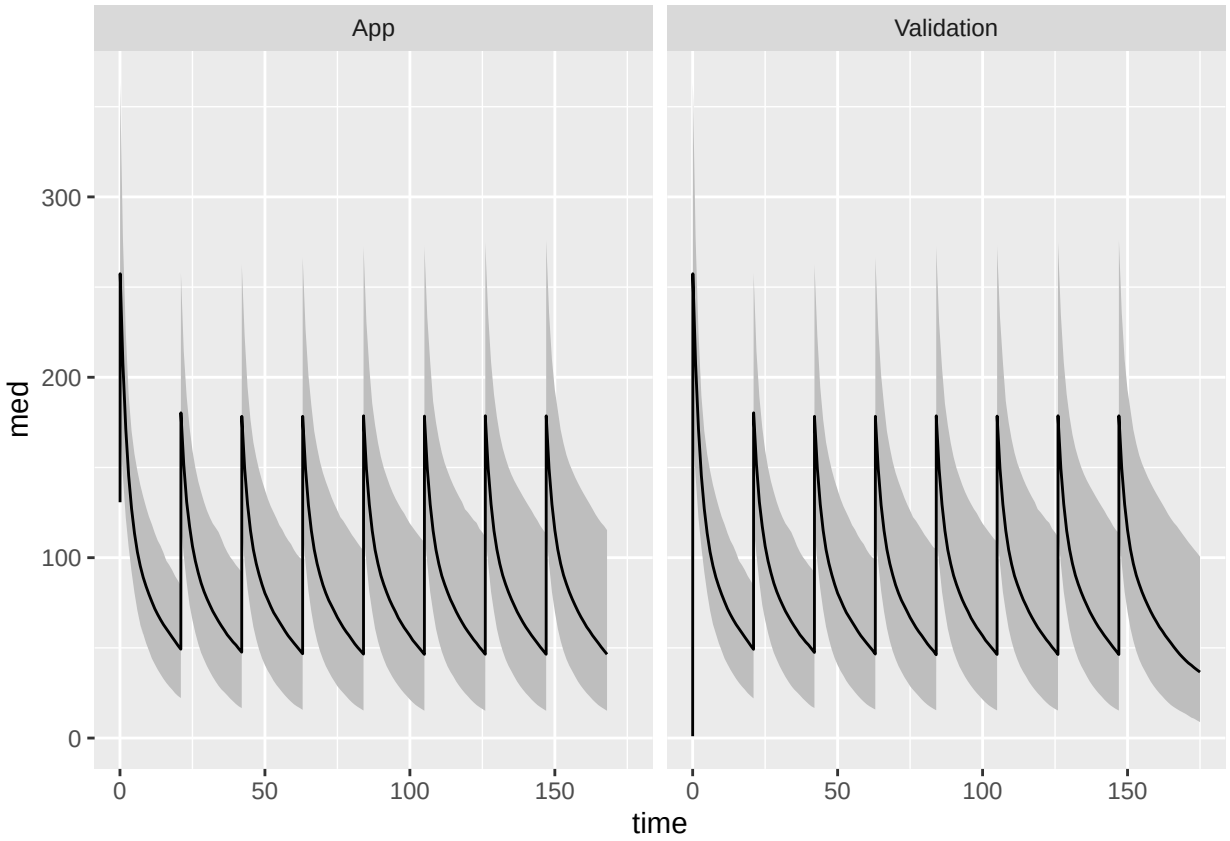
4.1 Case 1

```
plot.compare(case1.sim.dat,case1_simdata.app)
```



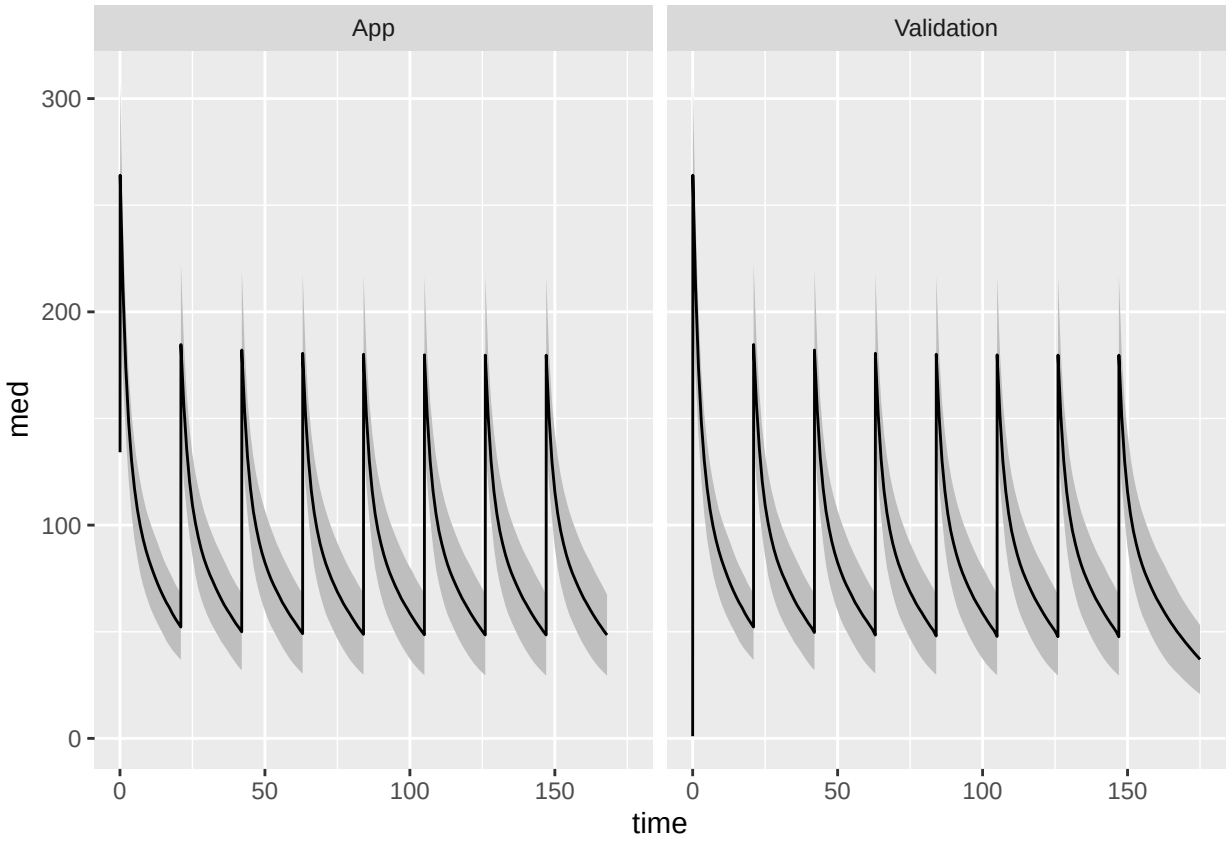
4.2 Case 2

```
plot.compare(case2.sim.dat,case2_simdata.app)
```



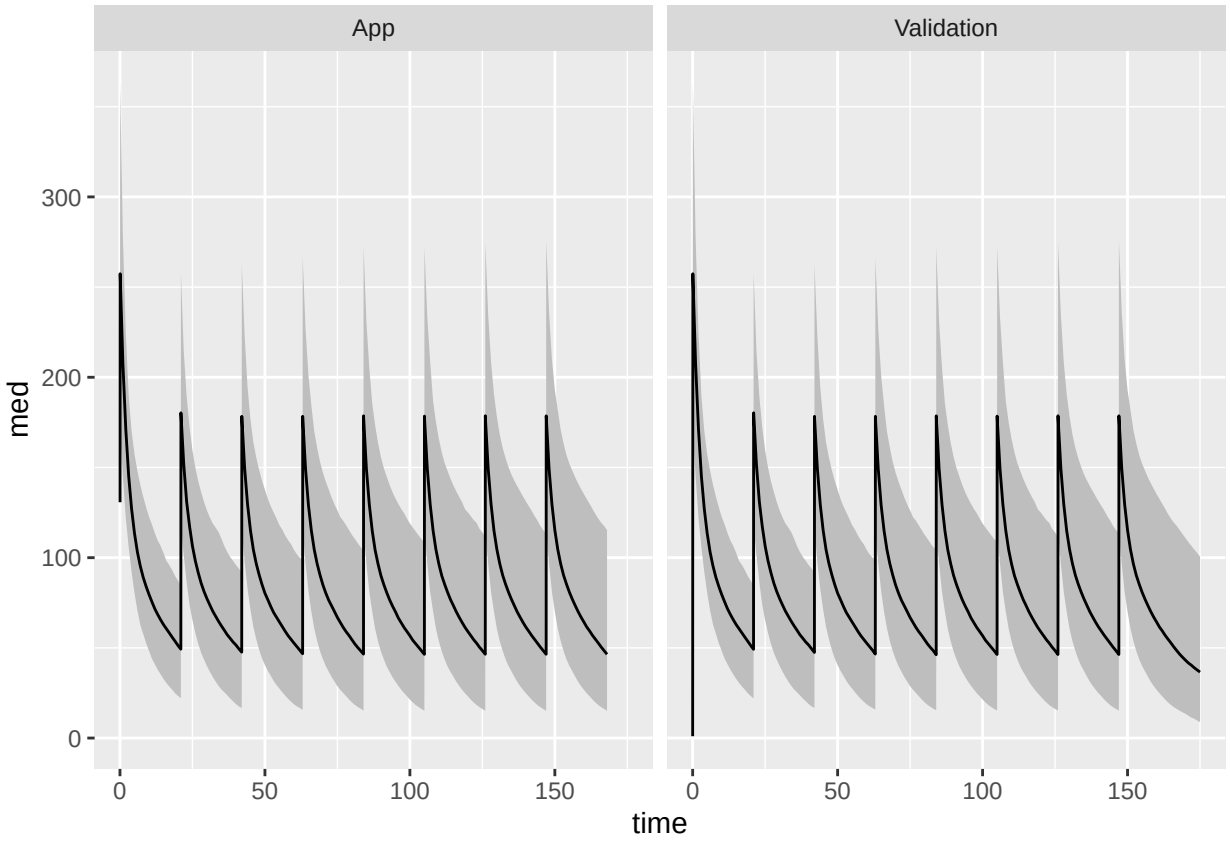
4.3 Case 3

```
plot.compare(case3.sim.dat,case3_simdata.app)
```

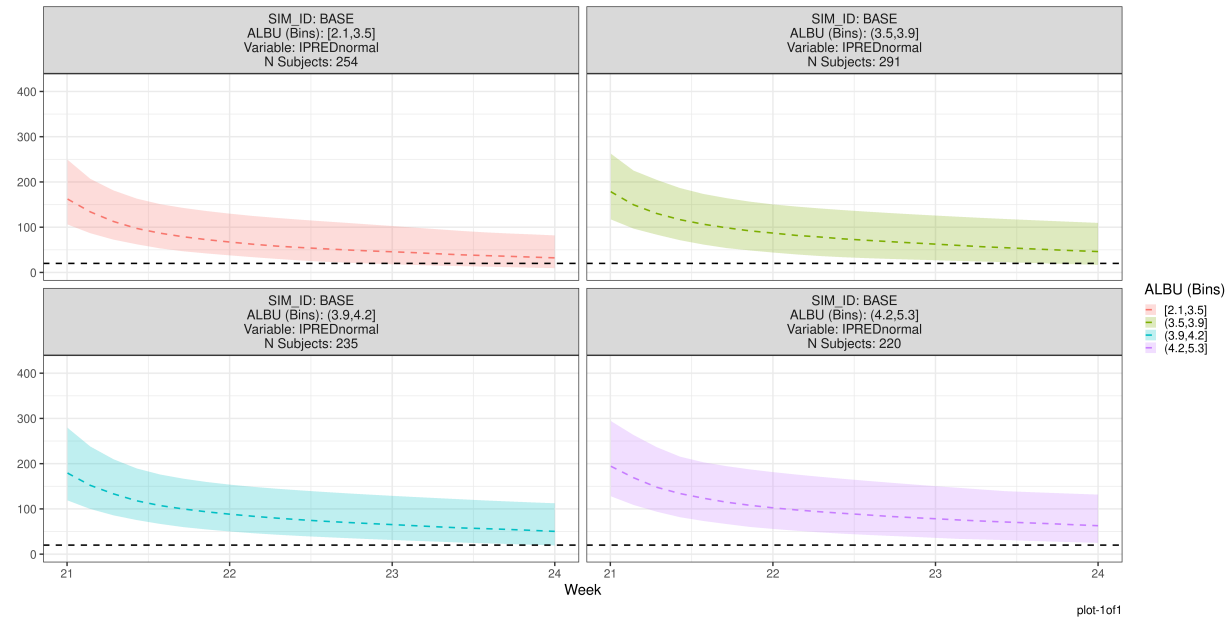


4.4 Case 4

```
plot.compare(case4.sim.dat,case4_simdata.app)
```



4.4.1 By ALBU Bin

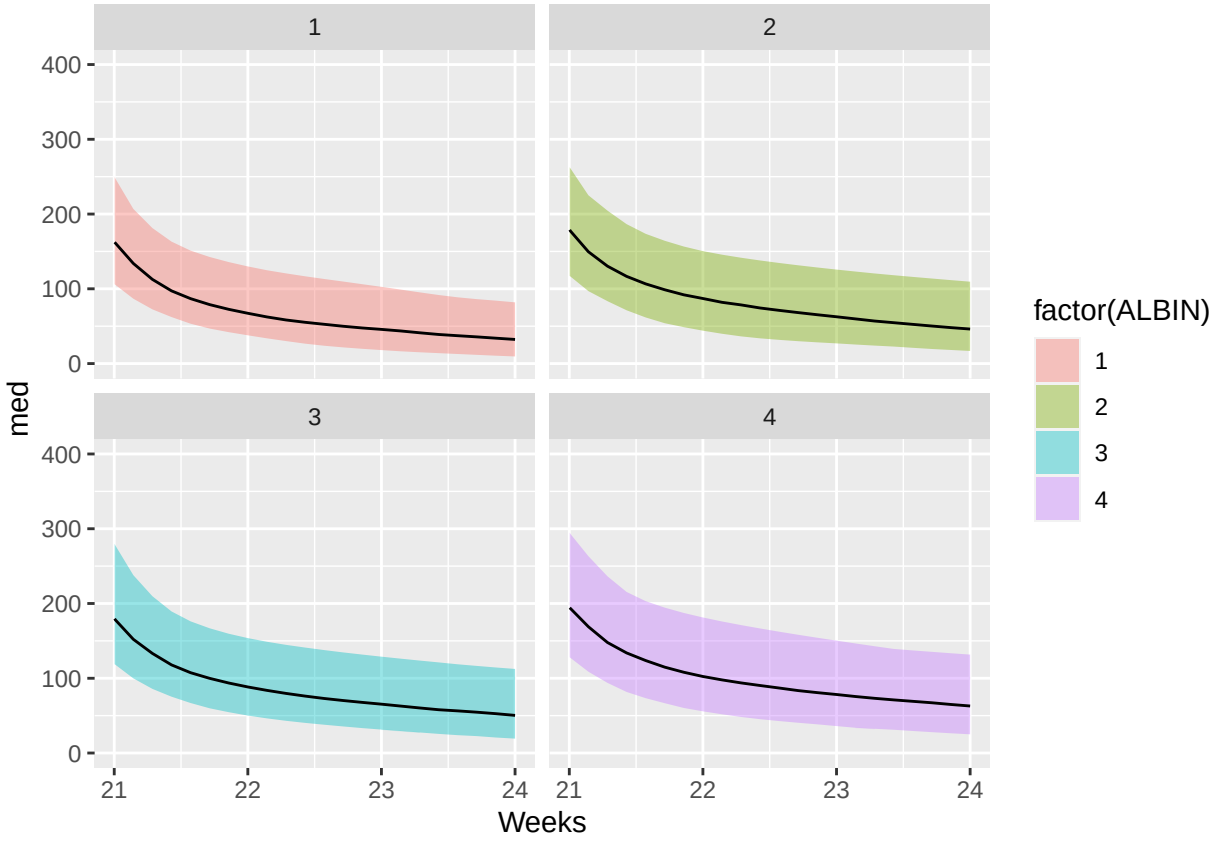



```
R.plot<-case4.sim.dat %>% filter(Cycle==8) %>%
  select(ID,time,IPREDnormal,ALBIN) %>%
```

```
  group_by(time,ALBIN) %>%
  dplyr::summarise(med=median(IPREDnormal),
            p5=quantile(IPREDnormal,p=0.05),
            p95=quantile(IPREDnormal,p=0.95)) %>%
  mutate(type="Validation")
```

```
## `summarise()` has grouped output by 'time'. You can override using the
## `.groups` argument.
```

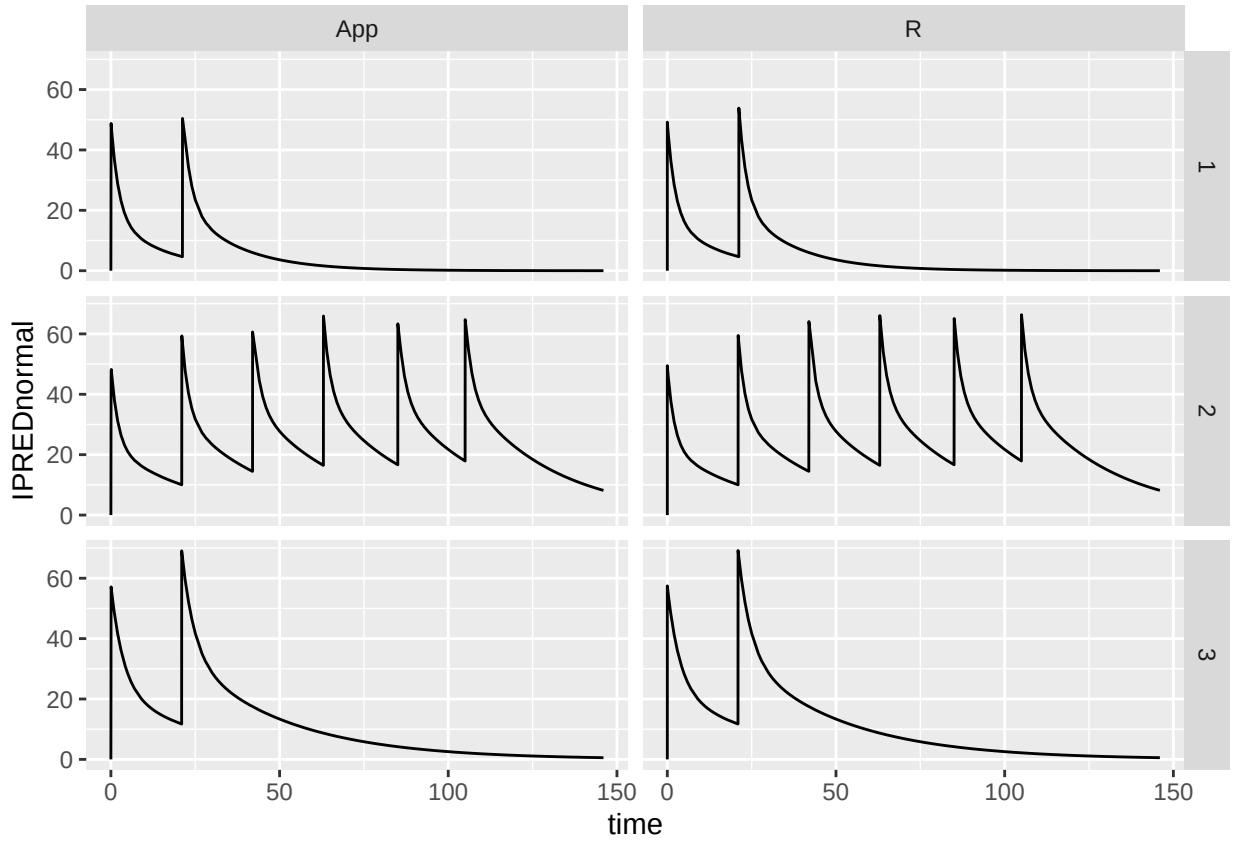
```
ggplot(R.plot,aes(time))+
  geom_ribbon(aes(ymin=p5,ymax=p95,fill=factor(ALBIN)),alpha=0.4)+
  geom_line(aes(y=med)) +
  facet_wrap("ALBIN")+ylim(c(0,400))+ scale_x_continuous(breaks=seq(21,24,by=1)*7,
                                                         labels=21:24)+
  labs(x="Weeks")
```



4.5 Case 5

```
case5.plot<-case5.sim.dat %>%
  filter(ID %in% 1:3) %>%
  mutate(type="R") %>%
  bind_rows(
```

```
    case5_simdata.app %>%
      filter(ID %in% 1:3) %>%
      mutate(type="App")
  )

ggplot(case5.plot,aes(time,IPREDnormal))+
  geom_line()+
  facet_grid(vars(ID),vars(type))
```



5 Summary Stats

- check AUC function in App

5.1 Function

```
# PKPDmisc::auc_partial(idv = time, dv = IPREDnormal)
# DescTools::AUC(time,IPREDnormal)
sumstats.func<-function(x,type=1){
  x %>%
  {if(type==1) group_by(.,ID,Cycle) else group_by(.,ID,Cycle,ALBIN)} %>%
  summarise(
```

```
    AUC=DescTools::AUC(time,IPREDnormal),
    LAST=min(IPREDnormal),
    MAX=max(IPREDnormal)
  ) %>% ungroup() %>%
  filter(!is.na(AUC)) %>%
  mutate(
    THRES=ifelse(LAST>20,1,0)
  ) %>%
   {if(type==1) group_by(.,Cycle) else group_by(.,Cycle,ALBIN)}  %>%
  summarise(
    THRES=100*sum(THRES)/n(),
    across(
      c(AUC,LAST,MAX),
      .fns=list(CV=function(x){100*sd(x)/mean(x)},
                GM=function(x){exp(mean(log(x)))},
                GCV=function(x){sqrt(exp(sd(log(x))^2)-1)*100},
                Mean=mean,
                SD=sd,
                p50=median,
                p5=function(x){quantile(x,p=0.05)},
                p95=function(x){quantile(x,p=0.95)})
    )
  )
}
```

5.2 Case 1

5.2.1 Validation

```
case1.sum.stats.R<-case1.sim.dat %>%
  filter(Cycle<=8,IPRED>0) %>%
  sumstats.func()
```

```
## `summarise()` has grouped output by 'ID'. You can override using the `.groups`
## argument.
```

5.2.2 Calculate % Difference

- With Same set of covariates, essentially same values

```
# Calculate Difference
abs_difference.case1 <- case1.app.stats %>%
  mutate(across(-Cycle,
                ~100-(100*abs(. - case1.sum.stats.R[[cur_column()]])/.),
                .names = "abs_diff_{.col}")) %>%
  select(Cycle,starts_with("abs"))
print(abs_difference.case1 %>%
        select(1,2:5))
```

```
## # A tibble: 8 x 5
```

```
##   Cycle abs_diff_AUC_Mean abs_diff_AUC_SD abs_diff_AUC_p50 abs_diff_AUC_CV
##   <int>             <dbl>           <dbl>            <dbl>           <dbl>
## 1     1              99.8            99.9             99.9           100.
## 2     2              99.9           100.              99.8            99.9
## 3     3              99.8           100.              99.8            99.9
## 4     4             100.            100.             100.            100.
## 5     5              99.8           100.              99.9           100.
## 6     6              99.9           100.              99.9           100.
## 7     7              99.8           100.              99.7            99.9
## 8     8              99.8           100.              99.8            99.9
```

```
print(abs_difference.case1 %>%
        select(1,6:9))
```

```
## # A tibble: 8 x 5
##   Cycle abs_diff_AUC_p5 abs_diff_AUC_p95 abs_diff_AUC_GM abs_diff_AUC_GCV
##   <int>           <dbl>            <dbl>           <dbl>            <dbl>
## 1     1            99.8            100.             99.8             99.8
## 2     2           100.             99.8            99.7             99.9
## 3     3           100.             99.9            99.9             99.9
## 4     4           100.             99.8            99.9             99.9
## 5     5           100.            100.             99.8             99.9
## 6     6           100.             99.9            99.8            100.
## 7     7           100.            100.             99.9             99.9
## 8     8           100.             99.9            99.8             99.9
```

```
print(abs_difference.case1 %>%
        select(1,10:13))
```

```
## # A tibble: 8 x 5
##   Cycle abs_diff_LAST_Mean abs_diff_LAST_SD abs_diff_LAST_p50 abs_diff_LAST_CV
##   <int>              <dbl>            <dbl>             <dbl>            <dbl>
## 1     1              100.              99.9              99.9            100.
## 2     2              100.              99.9              99.9            100.
## 3     3              100.              99.9             100.              99.9
## 4     4              100.             100.              100.              99.9
## 5     5               99.9             99.9             100.              99.9
## 6     6               99.9             99.9              99.9            100.
## 7     7              100.             100.               99.9            100.
## 8     8              100.             100.              100.              99.9
```

```
print(abs_difference.case1 %>%
        select(1,14:17))
```

```
## # A tibble: 8 x 5
##   Cycle abs_diff_LAST_p5 abs_diff_LAST_p95 abs_diff_LAST_GM abs_diff_LAST_GCV
##   <int>           <dbl>             <dbl>            <dbl>             <dbl>
## 1     1            99.8             100.              99.9              99.9
## 2     2           100.              100.             100.               99.9
## 3     3            99.9              99.6             99.9              99.9
## 4     4           100.               99.7             99.9             100.
## 5     5            99.7              99.7            100.              100.
```

```
## 6     6             99.7             99.7             99.9             100.
## 7     7             99.7             99.7            100.              100.
## 8     8             99.7             99.8            100.              100.
```

```
print(abs_difference.case1 %>%
        select(1,18:21))
```

```
## # A tibble: 8 x 5
##   Cycle abs_diff_MAX_Mean abs_diff_MAX_SD abs_diff_MAX_p50 abs_diff_MAX_CV
##   <int>             <dbl>           <dbl>            <dbl>           <dbl>
## 1     1              99.9            99.9             99.9            100.
## 2     2              99.8            99.9             99.8             99.9
## 3     3              99.8           100.              99.7            100.
## 4     4              99.8           100.              99.7             99.9
## 5     5              99.9            99.9             99.9            100.
## 6     6              99.9            99.9             99.8             99.9
## 7     7              99.8            99.9             99.9             99.9
## 8     8              99.9            99.9             99.9             99.9
```

```
print(abs_difference.case1 %>%
        select(1,22:25))
```

```
## # A tibble: 8 x 5
##   Cycle abs_diff_MAX_p5 abs_diff_MAX_p95 abs_diff_MAX_GM abs_diff_MAX_GCV
##   <int>           <dbl>            <dbl>           <dbl>            <dbl>
## 1     1            99.9             99.9            100.             100.
## 2     2            99.6             99.8             99.8             99.9
## 3     3            99.9             99.9             99.8            100.
## 4     4            99.7             99.9            100.              99.9
## 5     5            99.7             99.8             99.8             99.9
## 6     6            99.7             99.8             99.9             99.9
## 7     7            99.7            100.              99.7             99.8
## 8     8            99.7             99.8             99.9            100.
```

```
print(abs_difference.case1 %>%
        select(1,26))
```

```
## # A tibble: 8 x 2
##   Cycle abs_diff_THRES
##   <int>          <dbl>
## 1     1            100
## 2     2            100
## 3     3            100
## 4     4            100
## 5     5            100
## 6     6            100
## 7     7            100
## 8     8            100
```

5.3 Case 2

5.3.1 Validation data summary stats calculation

```
case2.sum.stats.R<-case2.sim.dat %>%
  filter(Cycle<=8,IPRED>0) %>%
  sumstats.func()
```

```
## `summarise()` has grouped output by 'ID'. You can override using the `.groups`
## argument.
```

5.3.2 Calculate Difference

```
# Calculate the absolute difference between the same variables in two datasets for sanity check
abs_difference.case2 <- case2.app.stats %>%
  mutate(across(-Cycle,
                ~100-(100*abs(. - case2.sum.stats.R[[cur_column()]])/.),
                .names = "abs_diff_{.col}")) %>%
  select(Cycle,starts_with("abs"))
print(abs_difference.case2 %>%
        select(1,2:5))
```

```
## # A tibble: 8 x 5
##   Cycle abs_diff_AUC_Mean abs_diff_AUC_SD abs_diff_AUC_p50 abs_diff_AUC_CV
##   <int>             <dbl>           <dbl>            <dbl>           <dbl>
## 1     1              99.8           100.              99.8            99.9
## 2     2              99.8            99.9             99.9            99.9
## 3     3              99.9           100.              99.8           100.
## 4     4              99.8           100.             100.             99.9
## 5     5              99.9            99.9             99.9           100.
## 6     6             100.             99.9            100.             99.9
## 7     7              99.9           100.              99.8            99.9
## 8     8              99.7           100.              99.9           100.
```

```
print(abs_difference.case2 %>%
        select(1,6:9))
```

```
## # A tibble: 8 x 5
##   Cycle abs_diff_AUC_p5 abs_diff_AUC_p95 abs_diff_AUC_GM abs_diff_AUC_GCV
##   <int>           <dbl>            <dbl>           <dbl>            <dbl>
## 1     1            99.9            100.             99.8             99.8
## 2     2           100.              99.9            99.7             99.9
## 3     3           100.             100.            100.             100.
## 4     4           100.              99.9            99.7             99.9
## 5     5           100.              99.9            99.8            100.
## 6     6           100.              99.9           100.             100.
## 7     7           100.              99.9            99.7             99.9
## 8     8           100.              99.9            99.9             99.9
```

```
print(abs_difference.case2 %>%
        select(1,10:13))
```

```
## # A tibble: 8 x 5
##   Cycle abs_diff_LAST_Mean abs_diff_LAST_SD abs_diff_LAST_p50 abs_diff_LAST_CV
##   <int>              <dbl>            <dbl>              <dbl>            <dbl>
## 1     1               99.9             99.8               99.9             99.9
## 2     2               99.9            100.               100.             100.
## 3     3              100.             100.               99.9             99.9
## 4     4              100.              99.9              99.9             99.9
## 5     5              100.              99.9             100.              99.9
## 6     6              100.              99.8              99.9             99.9
## 7     7               99.9            100.               99.9            100.
## 8     8              100.             100.               99.9             99.9
```

```
print(abs_difference.case2 %>%
        select(1,14:17))
```

```
## # A tibble: 8 x 5
##   Cycle abs_diff_LAST_p5 abs_diff_LAST_p95 abs_diff_LAST_GM abs_diff_LAST_GCV
##   <int>            <dbl>             <dbl>            <dbl>             <dbl>
## 1     1             99.9             100.              99.9              99.9
## 2     2             99.9             100.             100.              100.
## 3     3             99.9             100.             100.              99.9
## 4     4             99.7             100.              99.9              99.9
## 5     5             99.7              99.6            100.              100.
## 6     6             99.8              99.6            100.              99.9
## 7     7             99.8              99.8             99.9             100.
## 8     8             99.9              99.8             99.9             100.
```

```
print(abs_difference.case2 %>%
        select(1,18:21))
```

```
## # A tibble: 8 x 5
##   Cycle abs_diff_MAX_Mean abs_diff_MAX_SD abs_diff_MAX_p50 abs_diff_MAX_CV
##   <int>             <dbl>           <dbl>            <dbl>           <dbl>
## 1     1              99.9            99.9             99.8            100.
## 2     2              99.9           100.              99.8             99.8
## 3     3              99.9            99.9             99.8             99.9
## 4     4              99.7            99.9             99.8             99.9
## 5     5              99.9            99.9             99.8             99.9
## 6     6              99.9           100.              99.7             99.9
## 7     7              99.8           100.              99.8            100.
## 8     8              99.9            99.9             99.8             99.9
```

```
print(abs_difference.case2 %>%
        select(1,22:25))
```

```
## # A tibble: 8 x 5
##   Cycle abs_diff_MAX_p5 abs_diff_MAX_p95 abs_diff_MAX_GM abs_diff_MAX_GCV
##   <int>           <dbl>            <dbl>           <dbl>            <dbl>
```

```
## 1     1            99.8           100.            99.9            99.8
## 2     2            99.6            99.9           99.8            99.9
## 3     3           100.             99.9           99.8            99.8
## 4     4            99.7            99.9           99.8           100.
## 5     5            99.9            99.8           99.9           100.
## 6     6            99.8            99.9           99.9            99.9
## 7     7            99.8           100.            99.9            99.9
## 8     8            99.7            99.9           99.8            99.9
```

```
print(abs_difference.case2 %>%
        select(1,26))
```

```
## # A tibble: 8 x 2
##   Cycle abs_diff_THRES
##   <int>          <dbl>
## 1     1            100
## 2     2            100
## 3     3            100
## 4     4            100
## 5     5            100
## 6     6            100
## 7     7            100
## 8     8            100
```

5.4 Case 3

5.4.1 Validation data summary stats calculation

```
case3.sum.stats.R<-case3.sim.dat %>%
  filter(Cycle<=8,IPRED>0) %>%
  sumstats.func()
```

```
## `summarise()` has grouped output by 'ID'. You can override using the `.groups`
## argument.
```

5.4.2 Calculate Difference

```
# Calculate the absolute difference between the same variables in two datasets for sanity check
abs_difference.case3 <- case3.app.stats %>%
  mutate(across(-Cycle,
                ~100-(100*abs(. - case3.sum.stats.R[[cur_column()]])/.),
                .names = "abs_diff_{.col}")) %>%
  select(Cycle,starts_with("abs"))

print(abs_difference.case3 %>%
        select(1,2:5))
```

```
## # A tibble: 8 x 5
##   Cycle abs_diff_AUC_Mean abs_diff_AUC_SD abs_diff_AUC_p50 abs_diff_AUC_CV
```

```
##   <int>             <dbl>           <dbl>            <dbl>           <dbl>
## 1     1              99.8            99.9             99.8           100.
## 2     2             100.            100.              99.8            99.7
## 3     3              99.8            99.9             99.7            99.8
## 4     4             100.             99.9             99.9            99.7
## 5     5              99.8            99.9             99.9            99.8
## 6     6             100.            100.              99.9           100.
## 7     7             100.             99.9             99.8            99.9
## 8     8             100.             99.9             99.7            99.8
```

```
print(abs_difference.case3 %>%
        select(1,6:9))
```

```
## # A tibble: 8 x 5
##   Cycle abs_diff_AUC_p5 abs_diff_AUC_p95 abs_diff_AUC_GM abs_diff_AUC_GCV
##   <int>           <dbl>            <dbl>           <dbl>            <dbl>
## 1     1            99.9            100.             99.9             99.9
## 2     2            99.6             99.8            99.8             99.8
## 3     3            99.9            100.             99.9             99.9
## 4     4            99.6             99.9            99.8             99.7
## 5     5            99.8             99.9            99.8             99.8
## 6     6            99.6             99.9            99.7            100.
## 7     7            99.7            100.             99.8             99.9
## 8     8            99.7            100.             99.8             99.9
```

```
print(abs_difference.case3 %>%
        select(1,10:13))
```

```
## # A tibble: 8 x 5
##   Cycle abs_diff_LAST_Mean abs_diff_LAST_SD abs_diff_LAST_p50 abs_diff_LAST_CV
##   <int>              <dbl>            <dbl>             <dbl>            <dbl>
## 1     1               99.9            100.               99.9             99.8
## 2     2               99.9             99.6              99.9             99.9
## 3     3              100.              99.6             100.              99.8
## 4     4               99.9             99.9              99.9             99.8
## 5     5              100.              99.6              99.9             99.8
## 6     6              100.              99.7             100.              99.9
## 7     7              100.              99.8             100.              99.8
## 8     8               99.9             99.8             100.              99.8
```

```
print(abs_difference.case3 %>%
        select(1,14:17))
```

```
## # A tibble: 8 x 5
##   Cycle abs_diff_LAST_p5 abs_diff_LAST_p95 abs_diff_LAST_GM abs_diff_LAST_GCV
##   <int>            <dbl>             <dbl>            <dbl>             <dbl>
## 1     1             99.9             100.             100.               99.8
## 2     2            100.               99.9            100.               99.8
## 3     3             99.9             100.             100.               99.8
## 4     4            100.              100.              99.9              99.9
## 5     5             99.9              99.9             99.9              99.9
## 6     6            100.              100.              99.9             100.
```

```
## 7     7             99.9             100.              99.9             99.9
## 8     8             99.8             100.             100.              99.9
```

```
print(abs_difference.case3 %>%
        select(1,18:21))
```

```
## # A tibble: 8 x 5
##   Cycle abs_diff_MAX_Mean abs_diff_MAX_SD abs_diff_MAX_p50 abs_diff_MAX_CV
##   <int>             <dbl>           <dbl>            <dbl>           <dbl>
## 1     1              99.9           100.             100.              99.8
## 2     2              99.8            99.8             99.8             99.8
## 3     3             100.             99.9             99.9            100.
## 4     4              99.9            99.9             99.8             99.7
## 5     5              99.8            99.9             99.9             99.6
## 6     6              99.9            99.8             99.9             99.7
## 7     7              99.9            99.9             99.8             99.7
## 8     8              99.9            99.9             99.8             99.7
```

```
print(abs_difference.case3 %>%
        select(1,22:25))
```

```
## # A tibble: 8 x 5
##   Cycle abs_diff_MAX_p5 abs_diff_MAX_p95 abs_diff_MAX_GM abs_diff_MAX_GCV
##   <int>           <dbl>            <dbl>           <dbl>            <dbl>
## 1     1           100.              99.9           100.              99.6
## 2     2            99.9             99.8           100.              99.6
## 3     3            99.8             99.9            99.8             99.9
## 4     4            99.9             99.9            99.8             99.7
## 5     5            99.8             99.9            99.9             99.9
## 6     6            99.7             99.9            99.8            100.
## 7     7            99.7             99.9            99.8            100.
## 8     8            99.7            100.             99.8            100.
```

```
print(abs_difference.case3 %>%
        select(1,26))
```

```
## # A tibble: 8 x 2
##   Cycle abs_diff_THRES
##   <int>          <dbl>
## 1     1             NA
## 2     2            100
## 3     3            100
## 4     4            100
## 5     5            100
## 6     6            100
## 7     7            100
## 8     8            100
```

5.5 Case 4

5.5.1 Validation data summary stats calculation

```
case4.sum.stats.R<-case4.sim.dat %>%
  filter(Cycle<=8,IPRED>0) %>%
  sumstats.func(type=2) %>%
  filter(Cycle==8) %>%
  select(-Cycle)
```

```
## `summarise()` has grouped output by 'ID', 'Cycle'. You can override using the
## `.groups` argument.
## `summarise()` has grouped output by 'Cycle'. You can override using the
## `.groups` argument.
## Adding missing grouping variables: `Cycle`
```

5.5.2 Calculate Difference

```
# Calculate the absolute difference between the same variables in two datasets for sanity check
abs_difference.case4 <- case4.app.stats %>%
  mutate(across(-ALBIN,
                ~100-(100*abs(. - case4.sum.stats.R[[cur_column()]])/.),
                .names = "abs_diff_{.col}")) %>%
  select(ALBIN,starts_with("abs"))

print(abs_difference.case4 %>%
        select(1,2:5))
```

```
## # A tibble: 4 x 5
##   ALBIN abs_diff_AUC_Mean abs_diff_AUC_SD abs_diff_AUC_p50 abs_diff_AUC_CV
##   <int>             <dbl>           <dbl>            <dbl>           <dbl>
## 1     1              99.9            99.9             99.9            99.9
## 2     2              99.8            99.9             99.8            99.9
## 3     3              99.9           100.             100.            100.
## 4     4              99.8           100.              99.8            99.9
```

```
print(abs_difference.case4 %>%
        select(1,6:9))
```

```
## # A tibble: 4 x 5
##   ALBIN abs_diff_AUC_p5 abs_diff_AUC_p95 abs_diff_AUC_GM abs_diff_AUC_GCV
##   <int>           <dbl>            <dbl>           <dbl>            <dbl>
## 1     1           100.             100.             99.9             99.9
## 2     2           100.             100.             99.9             99.9
## 3     3           100.              99.9            99.8            100.
## 4     4            99.7            100.             99.9             99.9
```

```
print(abs_difference.case4 %>%
        select(1,10:13))
```

```
## # A tibble: 4 x 5
##   ALBIN abs_diff_LAST_Mean abs_diff_LAST_SD abs_diff_LAST_p50 abs_diff_LAST_CV
##   <int>              <dbl>            <dbl>             <dbl>            <dbl>
## 1     1              100.              99.8              99.9             99.9
## 2     2               99.9            100.               99.9            100.
## 3     3               99.9             99.9              99.9             99.9
## 4     4              100.              99.9             100.              99.9
```

```
print(abs_difference.case4 %>%
        select(1,14:17))
```

```
## # A tibble: 4 x 5
##   ALBIN abs_diff_LAST_p5 abs_diff_LAST_p95 abs_diff_LAST_GM abs_diff_LAST_GCV
##   <int>            <dbl>             <dbl>            <dbl>             <dbl>
## 1     1            100.              100.             100.               99.9
## 2     2             99.7              99.6             99.9             100.
## 3     3             99.8              99.6             99.9              99.9
## 4     4            100.               99.7            100.              100.
```

```
print(abs_difference.case4 %>%
        select(1,18:21))
```

```
## # A tibble: 4 x 5
##   ALBIN abs_diff_MAX_Mean abs_diff_MAX_SD abs_diff_MAX_p50 abs_diff_MAX_CV
##   <int>             <dbl>           <dbl>            <dbl>           <dbl>
## 1     1             100.             99.9             99.8           100.
## 2     2              99.9            99.9             99.9            99.9
## 3     3              99.7           100.              99.8            99.9
## 4     4              99.9            99.9             99.9            99.9
```

```
print(abs_difference.case4 %>%
        select(1,22:25))
```

```
## # A tibble: 4 x 5
##   ALBIN abs_diff_MAX_p5 abs_diff_MAX_p95 abs_diff_MAX_GM abs_diff_MAX_GCV
##   <int>           <dbl>            <dbl>           <dbl>            <dbl>
## 1     1            99.8             99.8            99.8            100.
## 2     2            99.8             99.8            99.8             99.9
## 3     3            99.8             99.9            99.9             99.9
## 4     4            99.9            100.             99.8             99.8
```

```
print(abs_difference.case4 %>%
        select(1,26))
```

```
## # A tibble: 4 x 2
##   ALBIN abs_diff_THRES
##   <int>          <dbl>
## 1     1           100.
## 2     2           100.
## 3     3           100.
## 4     4           100.
```