Visual Predictive Checks

Samuel Callisto, PhD

ECP 8506: Clinical Trial Simulations

8 November 2024

UNIVERSITY OF MINNESOTA

Driven to Discover®

Outline

- Principles of Simulation-Based Diagnostics
- Visual Predictive Checks (VPCs)
  - Simulation structure
  - Implementation
  - Prediction-corrected VPCs
- Landmark VPCs

What are simulation-based diagnostics?

- Standard model diagnostics use residuals and other comparisons between observed and predicted values
- Simulation-based diagnostics simulate from the model, and compare simulated data to observed data
- These approaches tell us if the model can produce data which are similar to that used to fit it
  - "Determine whether model deficiencies have a noticeable effect on substantive inferences" – Yano et al., 2000

Generating simulation-based diagnostics

- Use a simulation model to simulate from the data set used to fit the model
  - mrgsolve/PsN/NONMEM
- The number of simulation replicates depends on the choice of confidence interval (CI)
  - 200 replicates typically used for VPC CIs
  - Precision of estimates at 200 replicates = 0.5%

Visual Predictive Checks (VPCs)

- Several stochastic simulations are generated from the model (can be PK or PK-PD)
- Simulated data are summarized, typically 95% confidence intervals around median, 5^{th}, and 95^{th} percentiles
- Visual representation of the agreement between simulated and observed data

Layers of Stochasticity

- IIV (ETAs)
- RUV (EPSs)
- Uncertainty
  - var-cov, simpar, SIR, bootstrap

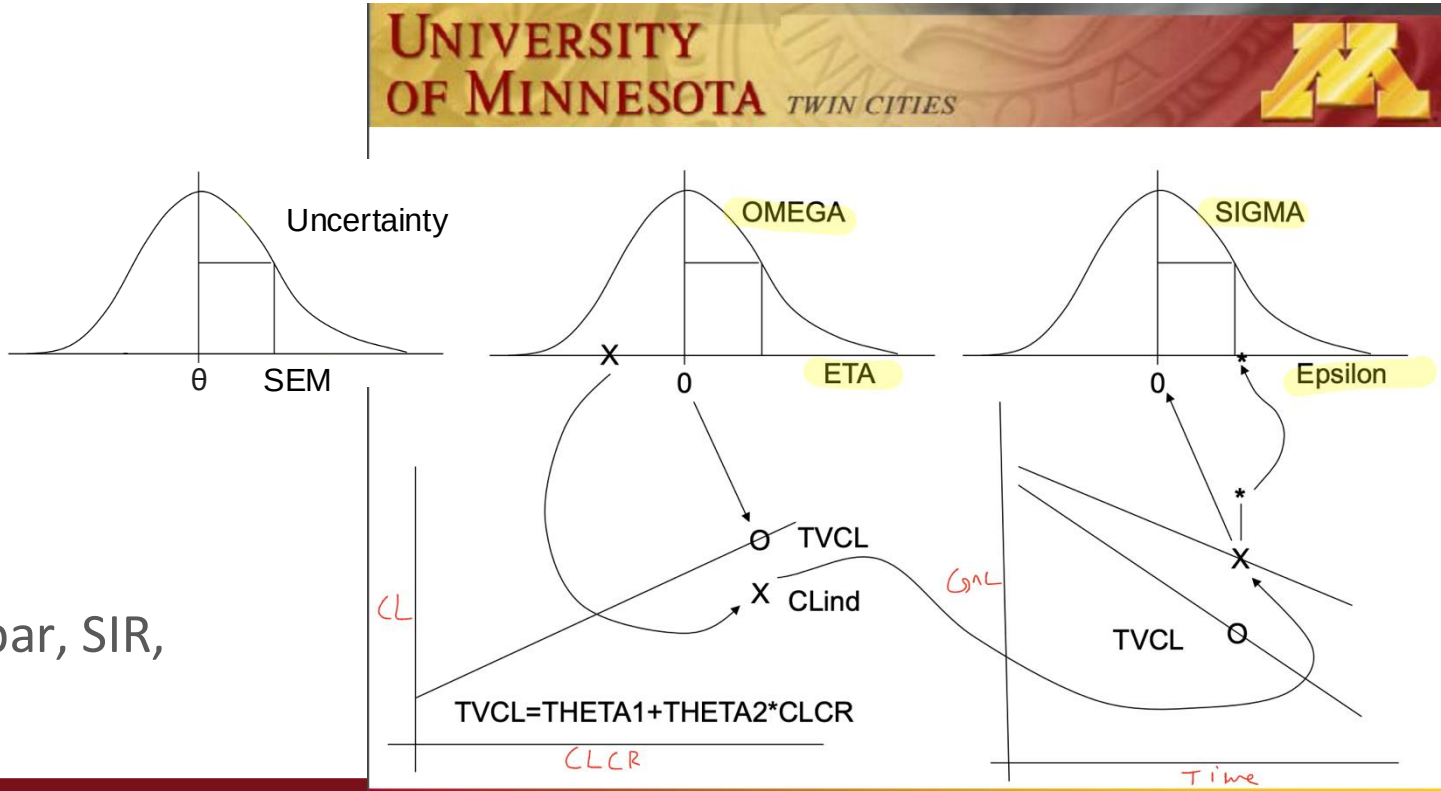


Figure adapted from Dick Brundage PopPK Lecture Slides

VPC Simulation Algorithm

- Create a function which will be called on each iteration which simulates from the study data set

- Use `lapply` or `map` to call the function the set number of times

```
#' Create a function to simulate out one replicate
sim <- function(rep, data, model) {
  mrgsim(
    model,
    data = data,
    carry_out = "EVID,STUDYN,LDOS,DOSE",
    recover = "STUDY,C,USUBJID,ACTARM,RF,Renal,Hepatic",
    Req = "Y",
    output = "df"
  ) %>%  mutate(irep = rep)
}

#' Simluate data

#' 200 replicates
isim <- seq(200)

set.seed(86486)
sims <- lapply(
  isim, sim,
  data = sad_mad,
  mod = mod
) %>% bind_rows()
```

“Should I include RUV?”

- The goal of the VPC is to verify that your model produces data similar to the observed
- If you ever want to compare simulated results to observed data, you need to include RUV
  - This is true in clinical trial simulations as well
  - If you don’t include RUV your confidence intervals will appear artificially narrow

Building a VPC from simulation data

- Summarize observed data at 5^{th}, 50^{th}, and 95^{th} percentiles within time bins (aka bands/intervals)
- Within each replicate, summarize 5^{th}, 50^{th}, and 95^{th} percentiles within time bins
- Across replicates summarize 2.5^{th} and 97.5^{th} percentile for each of the computed within-replicate percentiles
- Plot across-replicate 95% CI as ribbon, observed percentiles as lines, and optionally observed data as points

Interpreting your plot

- A good model will produce data similar to that which was used to fit it (i.e., the observed data)
- If the lines fall within the ribbons, this criteria is met

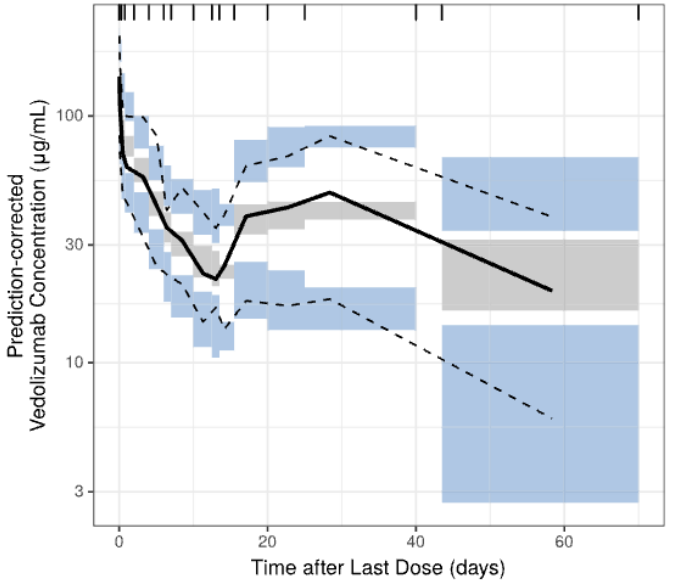
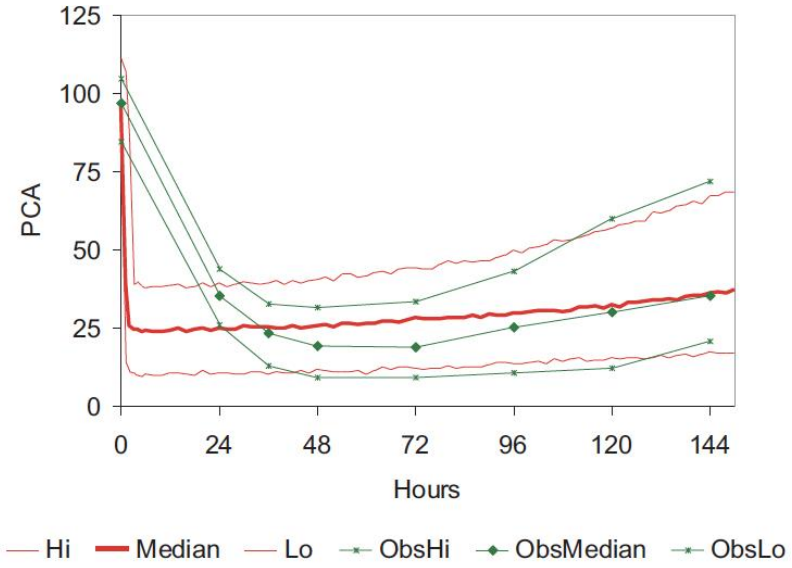


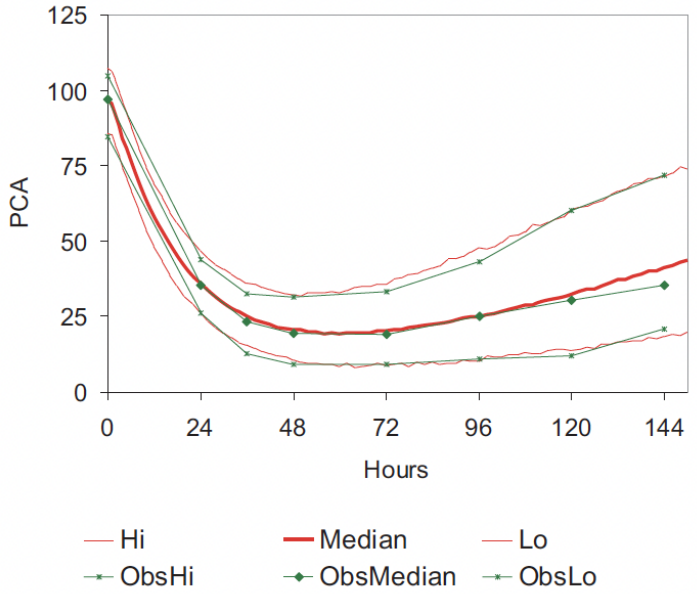
FIGURE 4 Final model: Prediction-corrected visual predictive check of vedolizumab concentration versus time after most recent dose. Black lines represent the median (solid), 5th and 95th percentiles (dashed) of the observed data. Blue and gray-shaded regions represent the 95% confidence intervals of the corresponding (i.e., 5th, 50th, and 95th) percentiles.

Figure: Waterhouse T et al. *Pharmacol Res Perspect*. 2024;12(5):e1257. doi:10.1002/prp2.1257

What does misspecification look like?



Misspecified model



Final model

Importance of binning

- If we don't bin our data, we get a "saw-tooth" pattern which makes it difficult to determine the quality of your model
  - This can also occur if you use too many bins
  - Too few bins will miss the trends in the data
- Algorithms exist to determine breakpoints for bins, but manual definition may yield best results

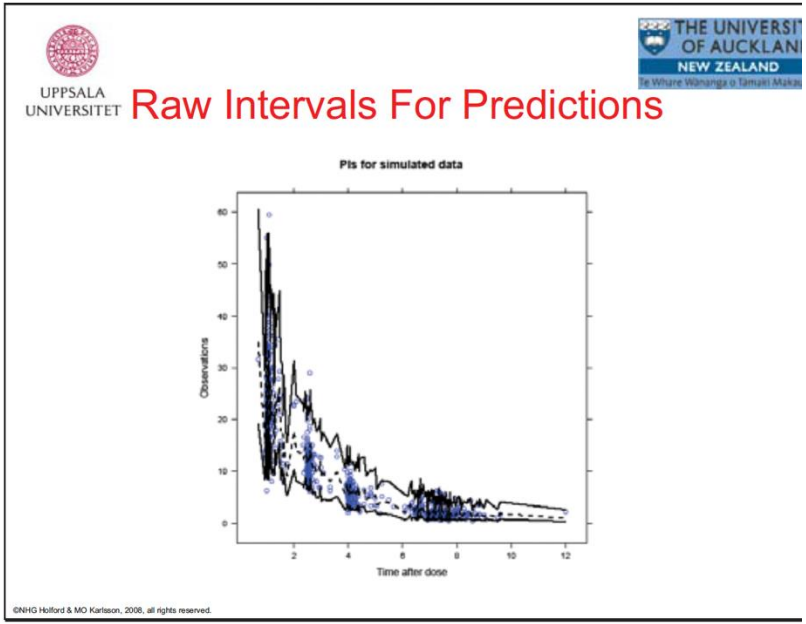
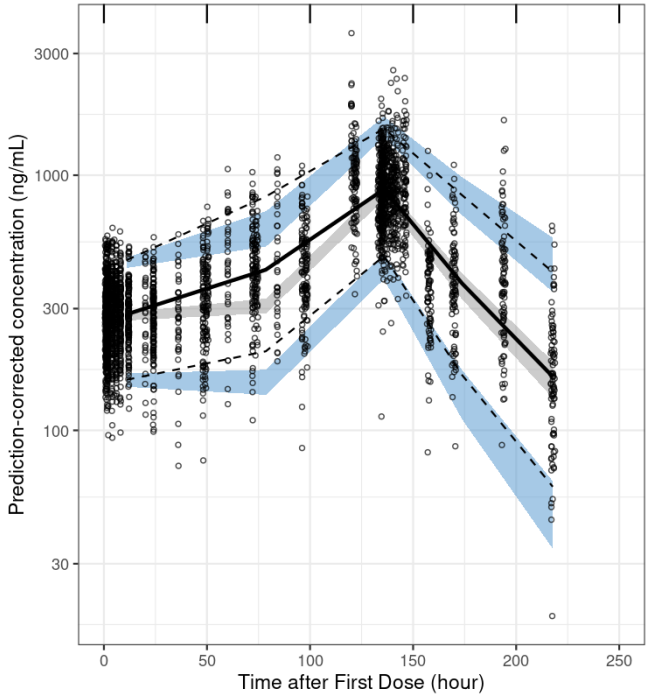


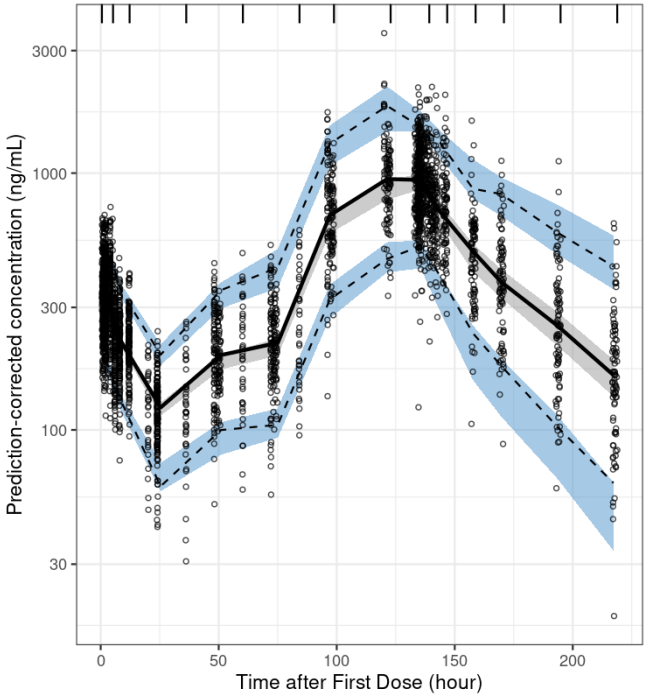
Figure: Holford & Karlsson, 2008 (VPC PAGE Presentation)

Examples of bad binning

Naïve equally-spaced bins



Jenks algorithm bin allocation



Handling BLQs in VPCs

- Several methods to handle BLQs in your model
  - If < 10%, typically we can just ignore all BLQ observations
  - Alternatives are M3 Method, replace with LLOQ/2, etc.
- When constructing the VPC, we need to consider how BLQ values may bias our estimates
- We want to simulate the same number of observations that were collected (“complete data approach”)

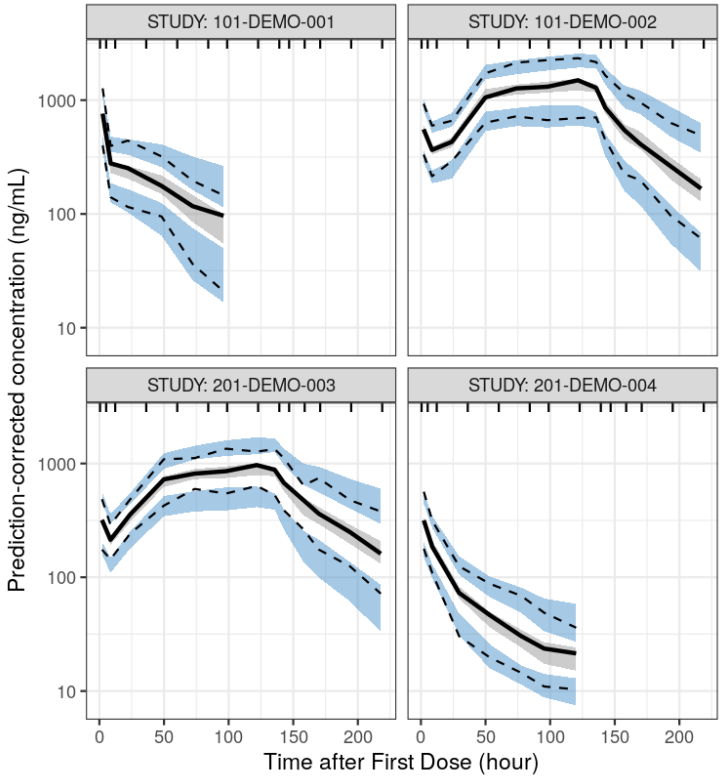
Proposed Option for Ignored BLQs

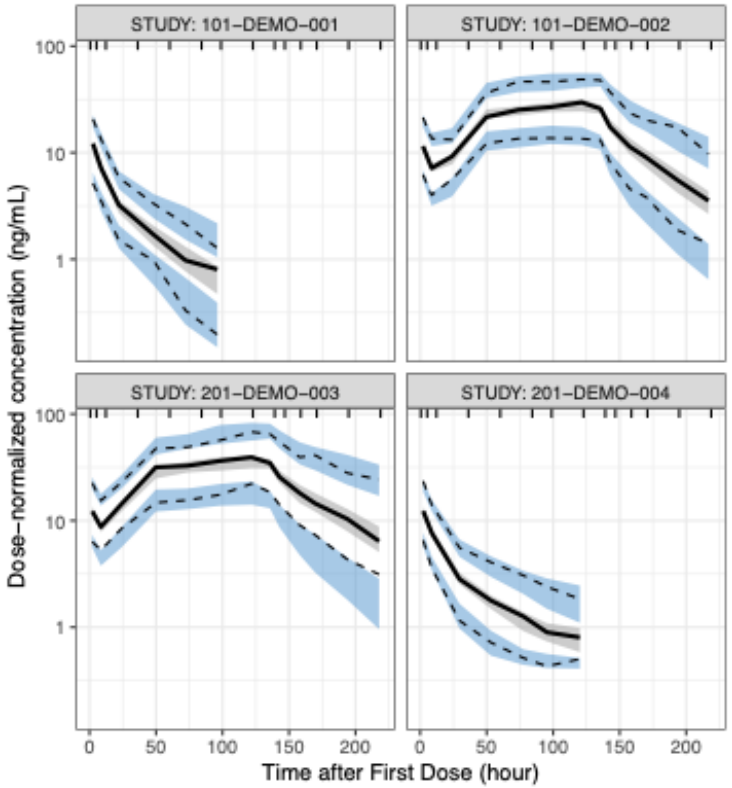
- Reintroduce BLQ observations into the data before you simulate the VPCs
  - This allows us to *simulate the same number of observations that were collected* in the study
- Within each simulation replicate, ignore the simulated values (including RUV) that fall below the LLOQ
  - Replicates will have different number of BLQs, but same number of observations overall

Prediction-Corrected VPCs

- Problem: Predictions within a bin differ largely due to independent variables such as dose
- Solution: Remove the variability coming from the independent variable by normalizing observed and predicted values to the typical prediction (PRED) within each bin
  - Equivalent to dose-normalizing in the case of linear kinetics with a fixed sampling schedule and no covariates included in the model

Prediction-Corrected vs Dose-Normalized VPC





Handling BLQs in pcVPCs

- BLQ values will not have an associated PRED value if they were ignored
- Use your model to simulate PRED values for BLQs and append to dataset

```
#'
#' # Data
#' This returns the complete data set by passing `.superset = TRUE`
#'
data <- nm_join(glue(here("model/pk/{runno}")), .superset = TRUE)
data <- filter(data, is.na(C))

#' Simulate `PRED` for the records that are BLQ, otherwise we'll use the
#' value coming from NONMEM
anyNA(data$PRED) # TRUE
out <- mrgsim(zero_re(mod), data, digits = 5)
data <- mutate(data, PRED = ifelse(BLQ > 0, out$Y, PRED))
anyNA(data$PRED) # FALSE
```

Stratifying VPCs

- Common to stratify VPCs by dose when not prediction-corrected
- Good practice to stratify on key variates (whether included as covariates or not)
- May also consider stratifying on treatment arm, study, routes of administration, formulations, occasions
- WARNING: as you stratify, your N will decrease

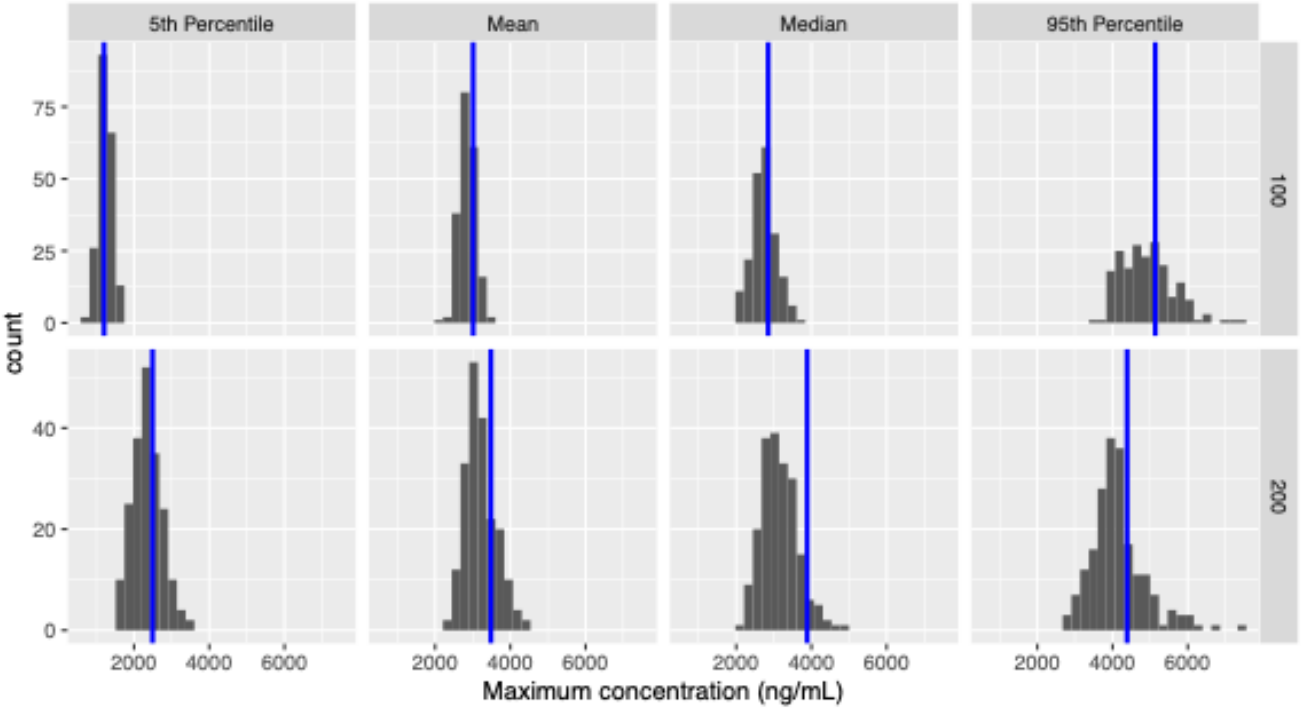
Landmark VPCs

- Whereas longitudinal VPCs show the model fit over some time period, landmark looks at model performance at a specific point in time
- This can be helpful for both Pop PK and PK-PD models
  - Determining if PK model predicts Cmax
  - Determining if PK-PD model predicts response at Week 52

Constructing a Landmark VPC Plot

- Use the same simulations which were used to develop the longitudinal VPC
- Filter down to the endpoint of interest
- Summarize simulation as appropriate (e.g., mean, median, 5^{th} and 95^{th} percentiles) within each replicate
- Overlay the summary of the observed data on the distribution of matching summary from simulations

Interpretation of Landmark VPCs



In a good model the observed data (blue line) will fall within the simulated distribution

Summary

- VPCs are a simulation-based diagnostic tool used to determine if your model generates data similar to observed
- VPC are generated by summarizing across several replicates of stochastic simulations including IIV and RUV
- Landmark VPCs are an alternative to longitudinal VPCs which looks at values from a single timepoint

References

- MeRGE: Visual Predictive Checks. https://merge.metrumrg.com/expo/expo1-nonmem-foce/posts/pk-vpc-final.html
- Yano, Y., Beal, S. L., & Sheiner, L. B. (2001). Evaluating pharmacokinetic/pharmacodynamic models using the posterior predictive check. *Journal of pharmacokinetics and pharmacodynamics*, 28, 171-192.
- Bergstrand, M., Hooker, A. C., Wallin, J. E., & Karlsson, M. O. (2011). Prediction-corrected visual predictive checks for diagnosing nonlinear mixed-effects models. *The AAPS journal*, 13, 143-151.
- Holford & Karlsson (2008). Model Evaluation: Visual Predictive Checks. Presented at PAGE 2008. https://www.page-meeting.org/?abstract=1434

University of Minnesota

Driven to Discover®

Crookston Duluth Morris Rochester Twin Cities