nlmeVPC: Visual Model Diagnosis for the Nonlinear Mixed Effect Model

by Eun-Hwa Kang, Myungji Ko, and Eun-Kyung Lee

Abstract A nonlinear mixed effects model is useful when the data are repeatedly measured within the same unit or correlated between units. Such models are widely used in medicine, disease mechanics, pharmacology, ecology, social science, psychology, etc. After fitting the nonlinear mixed effect model, model diagnostics are essential for verifying that the results are reliable. The visual predictive check (VPC) has recently been highlighted as a visual diagnostic tool for pharmacometric models. This method can also be applied to general nonlinear mixed effects models. However, functions for VPCs in existing R packages are specialized for pharmacometric model diagnosis, and are not suitable for general nonlinear mixed effect models. In this paper, we propose nlmeVPC, an R package for the visual diagnosis of various nonlinear mixed effect models. The **nlmeVPC** package allows for more diverse model diagnostics, including visual diagnostic tools that extend the concept of VPCs along with the capabilities of existing R packages.

1 Introduction

After fitting a model, diagnosing the fitted model is essential for verifying that the results are reliable (Nguyen et al. 2017). For linear models, the residuals are usually used to determine the goodness of fit of the fitted model. However, due to random effects and nonlinearity, residuals are less useful for diagnosing fit in nonlinear mixed effects models. Therefore, various diagnostic tools for this type of model have been developed. The nonlinear mixed effect model is useful when the data are repeatedly measured within the same unit, or the relationship between the dependent and independent variables is nonlinear. It is widely used in various fields, including medicine, disease mechanics, pharmacology, ecology, pharmacometrics, social science, and psychology (J. C. Pinheiro and Bates 2000; Davidian 2017). Recently, among the various diagnostic tools applicable to nonlinear mixed models, simulation-based diagnostic methods have been developed in the field of pharmacology (M. Karlsson and Savic 2007). The visual predictive check (VPC; M. O. Karlsson and Holford (2008)) is a critical diagnostic tool that visually tests for the adequacy of the fitted model. It allows for the diagnosis of fixed and random effects in mixed models (M. O. Karlsson and Holford 2008) in the original data space. Currently, it is a widely used method for diagnosing population pharmacometric models: Heus et al. (2022) used the VPC method to evaluate their population pharmacokinetic models of vancomycin, Mudde et al. (2022) checked their final population PK model for each regimen of antituberculosis drug using the VPC method, and Otto et al. (2021) compares the predictive performance of parent-metabolite models of (S)-ketamine with the VPC method. This VPC method can be used for general nonlinear mixed effect models, including hierarchical models.

The **psn** (Lindbom, Ribbing, and Jonsson 2004) and **xpose4** (R. J. Keizer, Karlsson, and Hooker 2013) packages provide various diagnostic methods for pharmacometric models in R (R Core Team 2023), including the VPC plot. These packages were developed only for the pharmacometricians, and the users needed to use the NONMEM software (Bauer 2011) for generating inputs of functions in these two packages. However, NONMEM is licensed software, and it is mainly designed towards the analysis of population pharmacometric models. Therefore, it is not easy for nonpharmacometrician to use NONMEM and to draw the VPC plot through **psn** and **xpose4**. Recently, the **vpc** (R. Keizer 2021) package and **nlmixr2** (Fidler et al. 2019) package have been developed to draw VPC plots in R without results from NONMEM. However, **vpc** package provides the function for drawing the original VPC plot only. **nlmixr2** was developed initially for fitting general dynamic models. To check the fitted model, **nlmixr2** provides a function to use graphical diagnostics in **xpose4** with the `nlmixr2` model object. Also, **nlmixr2** uses the VPC plot in the **vpc** package with the `nlmixr2` model object. Therefore, both packages only provide a function to draw the basic VPC plot, and the other newly developed simulation-based methods, including extensions of VPC, are not provided.

We have developed a new R package, **nlmeVPC**, to provide a suite for various visual checking methods for the nonlinear mixed models. This package includes various state-of-the-art model diagnostic methods based on the visual comparison between the original and simulated data. Most methods compare the statistics calculated from the observed data to the statistics from the simulated data. Percentiles, for example, the 10th, 50th, and 90th percentiles, are widely used to summarize relevant statistics from the observed and simulated data. We compare the similarities between the statistics from the observed data and those from the simulated data in two different spaces: the data space and the model space (Hadley Wickham, Cook, and Hofmann 2015). The original data comprise

the data space. Usually, the data space is represented by the independent and dependent variables, such as time and blood concentration, in the pharmacokinetic data. On the other hand, the model space is composed of quantities obtained from the fitted model, for example, residuals and summary values from the fitted model. From this viewpoint, we categorize the well-known visual diagnostic tools into two categories. One method compares the observed and simulated data in the original data space, and the other in the model space. In the method with the data space, we developed functions for the original VPC (`VPCgraph`), the additive quantile regression VPC (`aqrVPC`), and the bootstrap VPC (`bootVPC`). In addition, we proposed a new VPC method: the average shifted VPC (`asVPC`). In the method with the model space, the coverage plot (`coverageplot`) and the quantified VPC (`quantVPC`) are included. For a more detailed diagnosis, we developed a coverage detailed plot (`coverageDetailplot`). In **nlmeVPC**, the **ggplot2** package (H. Wickham 2016) is used to create all plots.

2 Nonlinear mixed effect model

The general nonlinear mixed effect model is defined as follow:

$$y_{ij} = f(x_{ij}, \theta, \eta_i, \epsilon_{ij}) \eta_i \sim N(0, \Omega) \epsilon_{ij} \sim N(0, \Sigma)$$

where y_{ij} is the dependent variable of the jth observation of the ith individual, x_{ij} is the independent variable, f is the nonlinear function of x_{ij}, θ is the population parameter, η_i represents the variability of the individual i, and ϵ_{ij} represents the random error. From the data, θ, Ω, and Σ are estimated. For the simulated data, the fitted model $y_{ij} = f(x_{ij}, \hat{\theta}, \eta_i, \epsilon_{ij})$, $\eta_i \sim N(0, \hat{\Omega})$, $\epsilon_{ij} \sim N(0, \hat{\Sigma})$ are used.

3 Validation in the data space

The VPC is the most popular model validation method in the pharmacometrics area. It was developed to diagnose population pharmacokinetic/pharmacodynamic models visually. The main idea of the VPC is to compare the distribution of observations and the distribution of predicted values, where the distribution of predicted values is obtained from simulated data drawn from the fitted model. If the fitted model explains the observed data well, these two distributions should be similar. Both distributions can be represented in the original data space that consists X axis as the independent variable and Y axis as the dependent variable. It allows us to compare the observed data with the fitted model in the original data space. In **nlmeVPC**, we include an original VPC plot, an additive quantile regression VPC (Jamsen et al. 2018), and a bootstrap VPC (Post et al. 2008). We also proposed a new approach to draw the VPC: the average shifted VPC.

3.1 Visual Predictive Check

The visual predictive check (VPC; M. O. Karlsson and Holford (2008)) is based on the principle that if the fitted model adequately describes the observed data, the distribution of the simulated data from the fitted model should be similar to the distribution of the observed data. There are several ways to compare the similarities between the distributions. In the VPC approach, profiles of quantiles are used. Two profiles are mainly used to compare the distributions of observations and predictions. One profile is from the upper bound of the prediction intervals, and the other is from the lower bound. These prediction intervals are calculated from the simulated data. 90% prediction intervals are usually used. For small and sparse samples, 80% prediction interval is also used. The lower and upper bounds of 80% prediction interval are the 10th and 90th percentiles of the simulated data. Figure 1(A) shows the "scatter" type of the VPC plot. Dots indicate the observed data. Two dashed blue lines represent profiles of the 10th and 90th percentiles of the simulated data, and the solid blue line represents the 50th percentile. If the fitted model represents the observed data well, most observed data should lie between profiles of 10th and 90th percentiles.

Figure 1(B) is the "percentile" type of the VPC plot. In this plot, profiles of percentiles from the observed data are compared to profiles of percentiles from the simulated data. Two dashed red lines represent profiles of the 10th and 90th percentiles of the observed data, and the solid red line represents profiles of the 50th percentile of the observed data. If the fitted model represents the observed data well, two profiles in each percentile - one from the original data and the other from the simulated data - are similar.

Figure 1(C) is the "CI" type of the VPC plot. The solid red line represents the 50th percentile of the observed data, and dashed red lines represent the 10th and 90th percentiles of the observed data.

Light blue areas represent the 95% confidence areas of the 10th and 90th percentiles, and pink areas represent the 95% confidence areas of the 50th percentile. These confidence areas were calculated from the simulated data. After calculating percentiles in each simulated data, we find 95% confidence intervals for each percentile and use this to draw the areas. In this plot, it is necessary to verify that the profiles of the original data are in confidence areas of each profile from the simulated data in each percentile. If each percentile line of the observed data is in the corresponding confidence area, this can be evidence that the fitted model represents the observed data quite well. Otherwise, the fitted model needs to be improved. The "CI" type of the VPC plot is the most widely used type in pharmacometrics.

The percentiles of the dependent variable are calculated in each bin. To estimate the percentiles accurately, enough data points need to lie within each bin. No binning means that the number of bins is equal to the number of different independent variable values. In this case, the VPC plot shows all details of the relationship between the independent and dependent variables. However, the resulting lines become too irregular to show meaningful trends or patterns.

As the number of bins decreases, the lines become smoother and more regular, however this can come at the loss of information if too much smoothing is used. Therefore, the selection of the best number of bins is crucial. The way of determining cutoffs for bin also plays an important role. Lavielle and Bleakley (2011) proposed a procedure for finding the optimal bin configuration automatically, including the optimal number of bins and the optimal bin cutoffs. VPCgraph provides the automatic binning with optK and makeCOVbin; here, optK finds the optimal number of bins, and makeCOVbin finds the optimal cutoffs of bins using Lavielle and Bleakley's method.

3.2 Additive quantile regression VPC

To overcome the difficulties of making bins as well as determining the number of bins, Jamsen et al. (2018) used additive quantile regression to calculate the quantiles of the observed and simulated data. This regression method makes it possible to estimate quantiles without discrete binning, which is especially useful when the data are insufficient, irregular, or inappropriate to configure the bins. To fit the additive quantile regression, we used the rqss function in the **quantreg** (Koenker 2023) package and developed the aqrVPC function to draw the VPC type plot with additive quantile regression. Figure 2 shows the additive quantile regression VPC plot. The solid and dashed lines represent the 10th, 50th, and 90th additive quantile regression lines of the observed data, and the pink and light blue areas represent the confidence areas of the additive quantile regression lines of the simulated data. Lines and areas in the additive quantile regression VPC plot are much smoother than those in the original VPC plot.

3.3 Bootstrap VPC

The bootstrap VPC (Post et al. 2008) compares the distribution of the simulated data to the distribution of the bootstrap samples drawn from the observed data. This plot reflects the uncertainty of the observed data and allows for more objective comparisons with the predicted median.

Figure 3 shows the bootstrap VPC plot using bootVPC. The solid and dashed blue lines represent the 10th, 50th, and 90th percentiles of the simulated data. The solid red line represents the 50th percentile line, and the pink areas represent the 95% confidence areas of the 50th percentile line, calculated from the bootstrap samples of the observed data. If the solid blue line and the solid red line are similar, the solid blue line is in the pink area, and the pink area is located between two dashed blue lines, then this is evidence that the fitted model fit the observed data well.

3.4 Average shifted VPC

Even though binning mitigates the problem with highly irregular data, the VPC plot still has a precision problem with sparse data. In this paper, we propose a new approach to draw the adapted VPC plot from the average shifted histograms (Scott 1985). A histogram is a widely used method for displaying the density of a single continuous variable. However, histograms can look quite different based on different choice of bin width and anchor. This requires computing an optimal bin width. To overcome the problem with the choice of bin width and to obtain smoother estimates, Scott (1985) proposed the averaged shifted histogram (ASH). The idea behind ASH is to make K different histograms with different anchors and to combine them via a weighted average. For each histogram, the starting point is shifted by d/K, where d is the width of the bin. We extend this idea to the VPC graph and propose the average shifted visual predictive check (asVPC) plot.
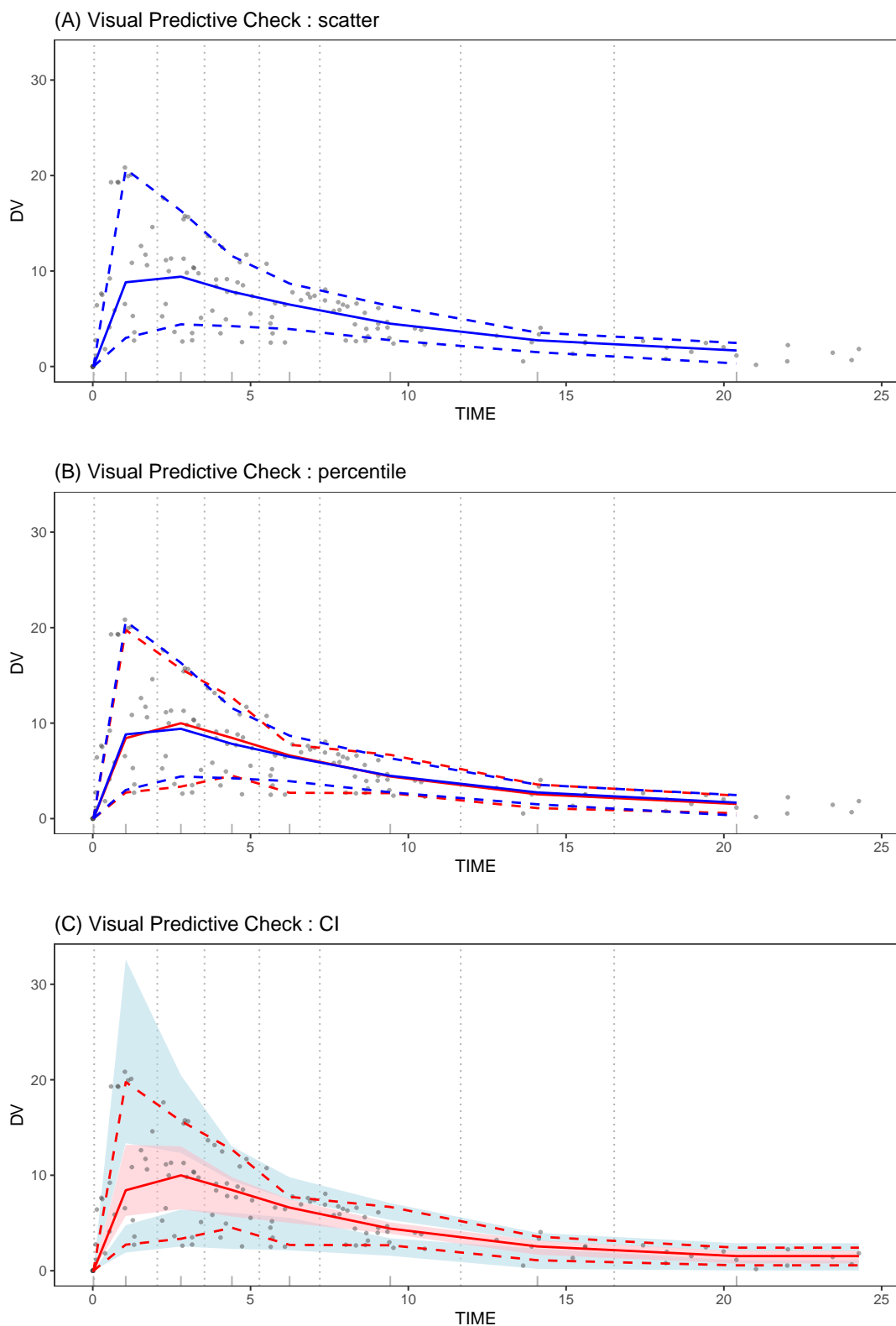
Figure 1: The visual predictive check plot. The solid red line represents the 50th percentile of the observed data, and dashed red lines represent the 10th and 90th percentiles of the observed data. The solid blue line represents the 50th percentile of the simularted data, and dashed blue lines represent the 10th and 90th percentiles of the simulated data. Light blue and pink areas represent the 95% confidence areas of the 10th, 50th and 90th percentile lines.
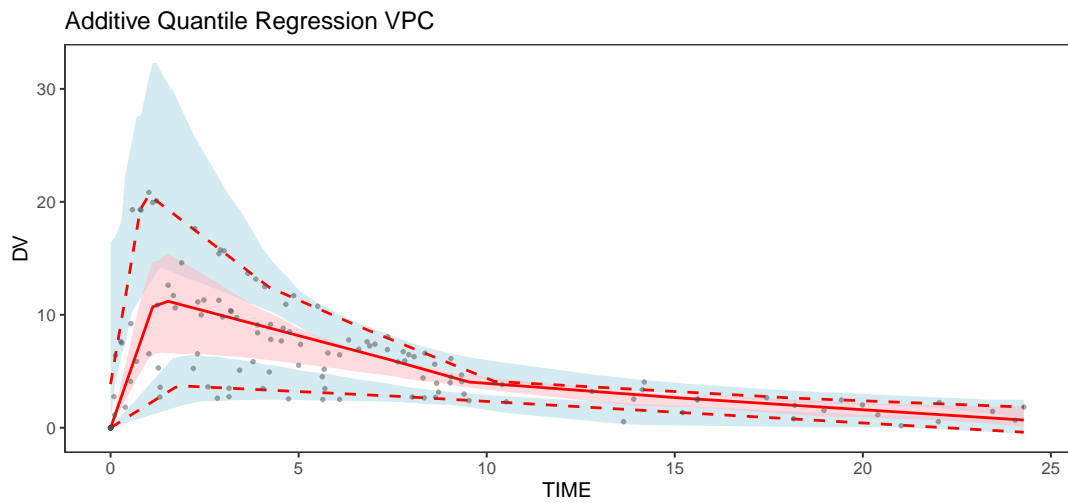
Figure 2: The additive equantile VPC plot. Dots indicate the observed data. The solid and dashed blue lines represent the 10th, 50th, and 90th percentiles of the simulated data. The solid red line represents the 50th percentile line. Light blue and pink areas represent the 95% confidence areas of the 10th, 50th and 90th percentile lines.
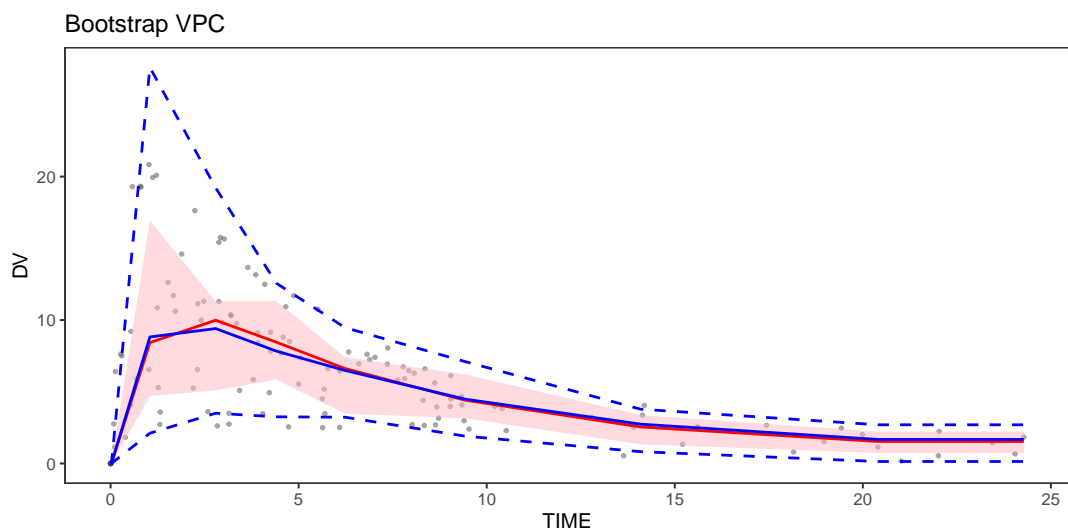

Figure 3: The bootstrap VPC plot. Dots indicate the observed data. The solid and dashed blue lines represent the 10th, 50th, and 90th percentiles of the simulated data. The solid red line represents the 50th percentile line, and the pink areas represent the 95% confidence areas of the 50th percentile line, calculated from the bootstrap samples of the observed data.

The binning of the VPC varies from the traditional binning in a histogram. The width of the bins is determined such that each bins contains the same, fixed number of observations, and as such the width of each bin is different. To apply the ASH idea to the VPC, we divide our data into $K * B$ bins along with the independent variable, where each bin has a different width but the same number of observations. Here, B is the number of the original bins in the VPC plot, and K is the number of the histograms averaged in the asVPC plot.

To draw the VPC plots, the following information is needed from each bin:

(A) the median of the independent variable and percentiles of the dependent variable of the observed data.

(B) the median of the independent variable and percentiles of the dependent variable of the simulated data.

(C) 95% confidence interval of percentiles (usually 10th, 50th, and 90th percentiles) of the dependent variable, calculated from the simulated data.

These values are also needed to produce confidence areas and percentile lines in the asVPC plot. Furthermore, additional steps are needed for asVPC. To find the median of the independent variable and percentiles of the dependent variable using the ASH algorithm, the following procedures are needed:

1. Divide the independent variable into $N = K * B$ bins.

2. For $i = 1, \cdots, N$,

   a) If $i < K$, combine $bin_1, \cdots, bin_{i+K-1} \setminus$ else if $K \leq i \leq N - K$, combine $bin_{i-K+1}, \cdots, bin_{i+K-1} \setminus$ else if $N - K < i$, combine $bin_{i-K+1}, \cdots, bin_N$
   b) Calculate the median of the independent variable and the weighted percentiles of the dependent variable in the combined bin

3. Collect the medians of the independent variable and the weighted percentiles of the dependent variable from 2, and connect them to the lines.

We can implement (A) and (B) by applying these procedures separately to the observed and simulated data. Additionally, (C) can be implemented using these procedures for each simulated dataset. First, we find the weighted percentiles, combine the results from each simulated dataset, and then calculate the 95% confidence intervals of each percentile. Using these three quantities (A), (B), and (C), we can draw the VPC type plot with the ASH approach, producing the asVPC plot.

Determining the weights

In the asVPC plot, the observations in each bin are combined using weights. Typically, the data near the center of the integrated bin have higher weights, and the data far from the center have smaller weights. This idea is used in the ASH algorithm as well as the density estimation literature. We suggest two different ways to apply weights for the asVPC calculation.

- Bin-related weight: For each bin in the combined bin, K-1 bins in both directions have weights and the other bins have no weight. Use $\frac{1}{K}, \cdots, \frac{i-1}{K}, \cdots, \frac{K-1}{K}, 1, \frac{K-1}{K}, \cdots, \frac{i-1}{K}, \cdots, \frac{1}{K}$ as weights for 2K-1 bins. This approach is the same as the original ASH approach.

- Distance-related weight: Use the reciprocal of the distance from the center of the independent variable in each combined bin so that the points near the center have higher weights and the points far from the center have lower weights.

Figure 4 shows the results from the asVPC function using bin-related weights and distance-related weights. The solid and dashed lines represent the average shifted quantile lines of the observed data, and the pink and light blue areas represent the confidence areas of the simulated data. The lines in the asVPC plot are smoother than those in the original VPC plot, and the confidence areas in the asVPC plot are thinner than those in the original VPC plot.

4   Validation in the model space

To validate the fitted model in the model space, we need to choose appropriate statistics to visualize and describe them in the model space. In **nlmeVPC**, we use the same statistics, that is quantiles, as the VPC plot in the data space, and compare them in two ways - numerically and visually. The numerical
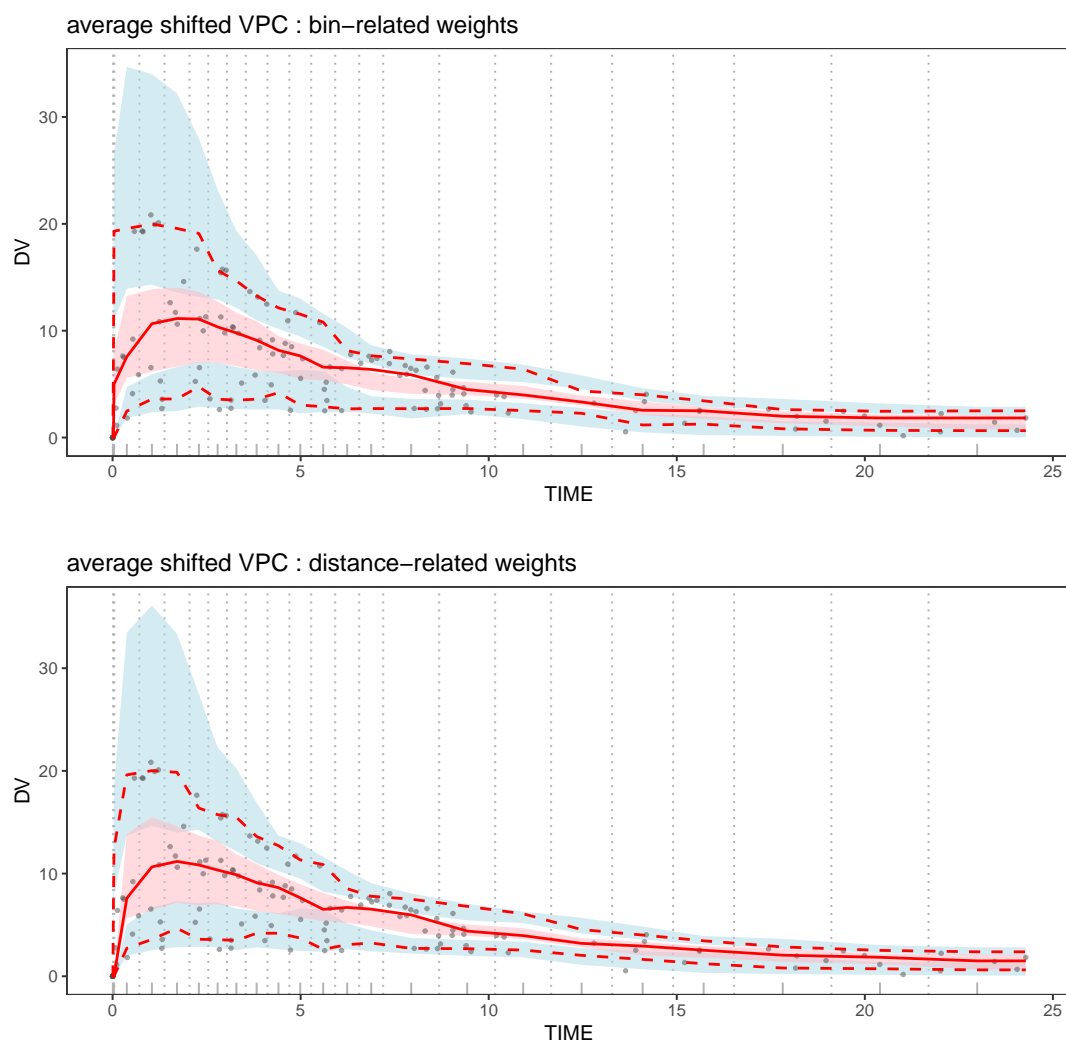
Figure 4: The average shifted VPC plot. Dots indicate the observed data. The solid line represents the 50th quantiles of the observed data, and dashed lines represent the 10th and 90th percentiles of the observed data. Light blue and pink areas represent the 95% confidence areas of the 10th, 50th, and 90th percentiles.

predictive check and the coverage plot are two commonly used methods in pharmacometrics. However, there is a limitation to detecting the illness of the fitted model. These methods combine the results of all ranges of the independent variable, and it is helpful to see the overall fitness. However, it is not easy to detect the detailed discrepancy between the fitted model and the observed data. To overcome this limitation, we developed a new plot, the coverage detailed plot, to compensate for the shortness of the coverage plot.

4.1 Numerical predictive check

The VPC plot visually compares the observed data to the simulated percentiles in the data space. On the other hand, the numerical predictive check (NPC; Wang and Zhang (2012); M. Karlsson and Savic (2007)) numerically compares the observed data to the simulated data. For a given level of prediction (for example, 90%), the predicted interval is calculated using the simulated data, and the number of the observed data points outside of the prediction interval is counted, both below the lower bound and above the upper bound. The expected number of points below the lower bound of the predicted interval (for example, 5% of observations) is also calculated and compared to the observed number. If these two numbers are similar, the suggested model is suitable (Maharaj et al. 2019).

NumericalCheck provides information summarizing the results of the NPC for various prediction levels. PI is the prediction level × 100 and n is the number of observations. "Expected points below PI" and "Expected points above PI" are respectively the expected numbers below and above the PI; "points below PI" and "points above PI" respectively represent the numbers of points below and above the PI; "95%CIBelowFrom(%)" and "95%CIBelowTo(%)" represent the 95% confidence interval of "points below PI(%)"; and "95%CIAboveFrom(%)" and "95%CIAboveTo(%)" represent the 95% confidence interval of "points above PI(%)". If "points below PI(%)" is in the 95% confidence intervals of "points below PI(%)" and is similar to "Expected points below PI", and if "points above PI(%)" is in the 95% confidence intervals of "points above PI(%)" and is similar to "Expected points above PI", this is the evidence that the fitted model explains the data well.

```
NumericalCheck(origdata,simdata,pred.level=c(0,0.2,0.4,0.6,0.8,0.9),N_xbin=8)$NPC

#>   PI   n Expected points below PI points below PI points below PI(%)
#> 1  0 132                     66.0              57           43.18182
#> 2 20 132                     52.8              46           34.84848
#> 3 40 132                     39.6              37           28.03030
#> 4 60 132                     26.4              27           20.45455
#> 5 80 132                     13.2              17           12.87879
#> 6 90 132                      6.6               7            5.30303
#>   95%CIBelowFrom(%) 95%CIBelowTo(%) Expected points above PI points above PI
#> 1        35.6060606        56.81818                     66.0              63
#> 2        26.8750000        48.88258                     52.8              49
#> 3        20.4545455        36.76136                     39.6              35
#> 4        11.3636364        26.15530                     26.4              28
#> 5         3.7878788        16.66667                     13.2              12
#> 6         0.3598485        10.64394                      6.6               5
#>   points above PI(%) 95%CIAboveFrom(%) 95%CIAboveTo(%)
#> 1          47.727273         34.090909       55.303030
#> 2          37.121212         25.359848       46.609848
#> 3          26.515152         16.666667       36.003788
#> 4          21.212121         10.965909       25.000000
#> 5           9.090909          2.632576       15.549242
#> 6           3.787879          1.117424        8.333333
```

4.2 Coverage plot

The result of the NPC is a table with many values, which, while useful, can be difficult to parse visually. The coverage plot (M. O. Karlsson and Holford 2008) was developed to help visually check the fitted model with the NPC result. In each level of the predicted interval, the ratios between the expected number of data points (Exp) outside the prediction interval and the observed number of data points (Obs) outside the prediction interval are calculated. These ratios are calculated separately for the upper and lower bound of the prediction interval. For example, when the prediction level is 90, a 90% prediction interval is used, and 10% of the observations are expected to locate outside this prediction interval. To be more precise, 5% of observations are expected to be above the upper limit, and 5% of observations are expected to be below the lower limit.

The coverage plot with the NPC result can diagnose a model using multiple prediction intervals. The X-axis represents the prediction level, and the Y-axis represents Obs/Exp. The closer Obs/Exp is to 1, the more appropriate the model is. Furthermore, the confidence intervals of Obs/Exp values are obtained using simulated data and then expressed together in the plot for a more objective comparison. This plot can provide more information than the VPC plot, which interprets only a couple of quantiles - usually the 10%, 50%, and 90% percentiles.

The **xpose4** package (R. J. Keizer, Karlsson, and Hooker 2013) provides a coverage plot. However, to draw the coverage plot using **xpose4**, PsN (Lindbom, Ribbing, and Jonsson 2004) software is needed to calculate the NPC result. Therefore, we developed `coverageplot` to draw the coverage plot using the results from `NumericalCheck`.

Figure 5(A) shows the coverage plot. The X-axis shows the level of the prediction interval. The Y-axis show the ratio between the observed number of data and the expected number of data of the lower and upper parts in each level of the prediction interval. The white line is the reference line, and the gray area represents the confidence areas of the ratios in each prediction level. If the solid lines are near the white line or in the gray area, we can conclude that the suggested model is suitable.

4.3 Coverage detailed plot

Unlike the VPC plot, which represents the data space, the information in the observed data space does not come together in the coverage plot, which makes it difficult to determine whether the model is overestimated, underestimated, or adequate in the specific region of the data space. To overcome this limitation of the coverage plot, we propose a new method called the coverage detailed plot. The percentages of observations above the prediction interval are calculated in each bin of the independent variable. Additionally, the percentages of observations below the prediction interval are calculated. The white dots in the plot represent the expected percentages. If the percentages of upper and lower observations are near the white dots, we can conclude that the suggested model is suitable for the specific prediction interval.

Figure 5(B) is the result of `coverageDetailplot` when the prediction level is 80%. The white dots represent the expected percentages of the lower and upper the prediction intervals, 10%, and 90%, respectively. The upper and lower percentages of observation in each time bin are shown in darker gray. The left bin(before 0.045 hours) shows all light gray in the coverage detailed plot, and it is quite different patterns from the expected one. However, it is mainly due to the characteristics of this example data. All observations in this bin are 0. It makes the lower and upper bound of the prediction interval all 0, and the lower and upper percentages become 0.

4.4 Quantified VPC

Post et al. (2008) proposed the quantified VPC (QVPC). It expanded the existing VPC, including information about missing data. The QVPC plot visually represents actual and unavailable observations around the predicted medians through the form of percent, regardless of the observed data's density or shape. Here, "unavailable observations" refer to all kinds of missing values and unobserved data that occur for various reasons, including deviations and unexpected omissions.

If the model is appropriate, observations at each time bin are allocated randomly around the predicted median of the model. In the QVPC plot, white dots represent the model's predicted median in each bin. If the borderlines above and below are close to the white dots, we can conclude that the fitted model describes the observed data well.

Figure 6 shows the result of `quantVPC`. The darker gray areas represent the percentages below the median. The lighter gray areas represent the percentage above. The brightest gray areas represent the percent unavailable in each time bin. The white dots represent the ideal location where the above and the below percentages meet. In this example, there is no missing value.

5 The nlmeVPC package: structure and functionality

Table **??** shows the list of functions and Table **??** shows the list of arguments used in the functions of **nlmeVPC**. The following codes are for Figure 1 to Figure 6.

```
library(nlmeVPC)
data(origdata)
data(simdata)
```

(A) Coverage Plot

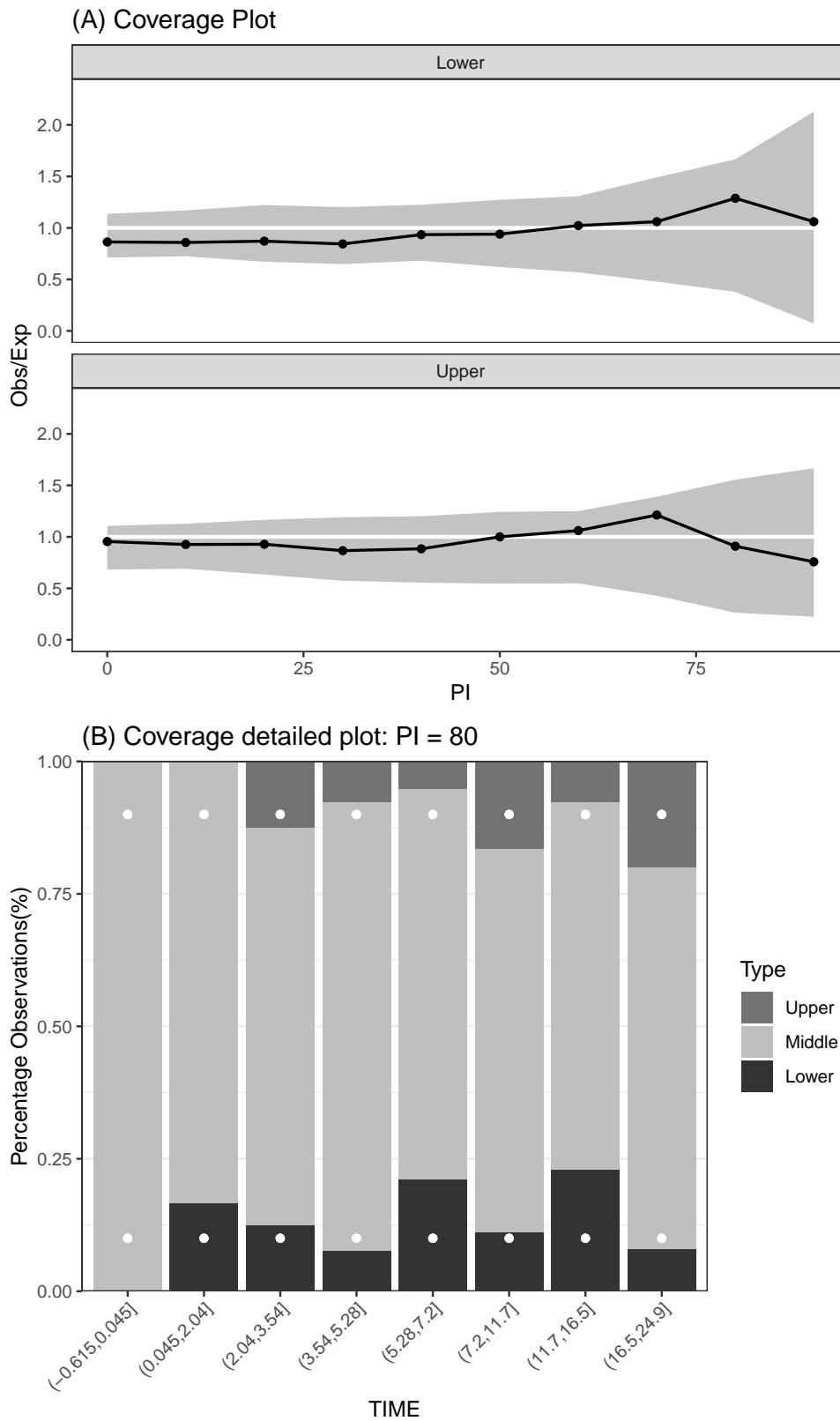## (B) Coverage detailed plot: PI = 80

Figure 5: The coverage plot and the coverage detailed plot for the 80% prediction interval. In the coverage plot, the X-axis is the level of the prediction interval. The Y-axis is the ratio between the number of observed data and the number of expected data of the lower and upper parts in each level of the prediction interval. The white line is the reference line, and the gray area represents the confidence area of the ratios. If the solid lines are near the white line, we can conclude that the suggested model is suitable. In the coverage detailed plot, the white dots represent the expected percentages of lower and upper prediction intervals of, 10%, and 90%, respectively. The upper and lower percentages of observation in each time bin are darker gray.
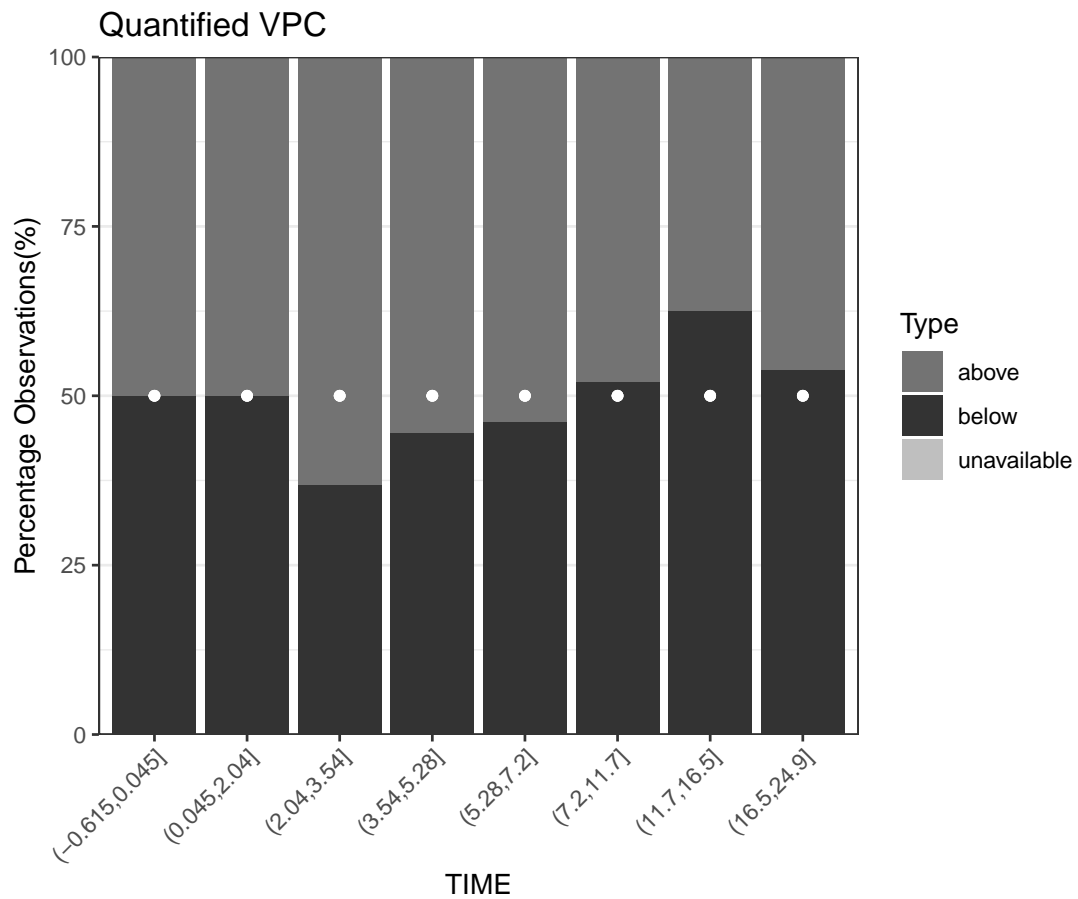
Figure 6: The quantified VPC plot. The darker gray areas represent the percentages below the median, the lighter gray areas represent the percentage above, and the brightest gray areas represent the percent unavailable in each time bin. The white dots represent the ideal location where the above and the below percentages meet.

```
optK(origdata$TIME)$K

#> [1] 8

# Figure 1

VPCgraph(origdata,simdata,N_xbin=8,type="scatter")+
  labs(title="(A) Visual Predictive Check : scatter",caption="")
VPCgraph(origdata,simdata,N_xbin=8,type="percentile")+
  labs(title="(B) Visual Predictive Check : percentile",caption="")
VPCgraph(origdata,simdata,N_xbin=8,type="CI")+
  labs(title="(C) Visual Predictive Check : CI",caption="")

# Figure 2

aqrVPC(origdata,simdata) +labs(caption="")

# Figure 3

bootVPC(origdata,simdata,N_xbin=8)+labs(caption="")

# Figure 4

asVPC(origdata,simdata,type="CI",N_xbin=8,N_hist=3,weight_method="bin")+labs(caption="")
asVPC(origdata,simdata,type="CI",N_xbin=8,N_hist=3,weight_method="distance")+labs(caption="")

# Numerical Predictive Check

NumericalCheck(origdata,simdata,N_xbin=8,pred.level=c(0,0.2,0.4,0.6,0.8,0.9))$NPC

# Figure 5


coverageplot(origdata,simdata,N_xbin=8) +ggtitle("(A) Coverage Plot")
coverageDetailplot(origdata,simdata,N_xbin=8,predL=0.8) +
ggtitle("(B) Coverage Detailed plot: PI = 80")

# Figure 6

quantVPC(origdata,simdata)
```

We use an example to show how to use functions in **nlmeVPC** and how they work.

5.1 Example

The origdata in **nlmeVPC** is from an experiment on the pharmacokinetics of theophylline. Twelve patients were given oral doses of theophylline, and blood concentrations were measured at 11 time points over the next 25 hours. Each patient had different time points. We consider the following first-order absorption one-compartment model:

$$y_{ij} = \frac{Amt_i * Ke_i * Ka_i}{Cl_i} \left(\exp(-Ke_i * TIME_{ij}) - \exp(-Ka_i * TIME_{ij}) \right) + \varepsilon_{ij}.$$

In this model, y_{ij} is the theophylline concentration at $TIME_{ij}$ after an initial dose of Amt_i. The pharmacokinetic parameters are the absorption rate constant Ka, the elimination rate constant Ke, and the clearance Cl. In this example, two different models are fitted and diagnosed using functions in the **nlmeVPC** package. In Model 1, Ka and Cl are considered as random effects. In Model 2, Ke is considered as random effect, and Ka and Cl are considered only as a fixed effect. The **nlme** (J. Pinheiro et al. 2022) and **nlraa** (Miguez 2022) packages are used to fit the nonlinear mixed models and to generate the simulated data from the fitted model.

```
library(nlme)
library(nlraa)
library(nlmeVPC)
```

```
data(origdata)
origdataT <- groupedData(DV~TIME|ID,origdata)

# Model 1 (True)
T.nlme <- nlme(DV ~ exp(lKe+lKa-lCl)*AMT*
                 (exp(-exp(lKe)*TIME) - exp(-exp(lKa)*TIME))/
                 (exp(lKa)-exp(lKe)), data=origdataT,
             fixed=lKa+lKe+lCl~1,
             random=lKa+lCl~1,
             start=c(lKe=-2,lKa=1.5,lCl=-3))
set.seed(123456)
sim.T <- simulate_nlme(object=T.nlme,nsim=100,psim=3,level=1,value="data.frame",data = NULL)
simdata.T <- matrix(sim.T$sim.y,ncol=100)

# Model 2 (Wrong)
F.nlme <- nlme(DV ~ exp(lKe+lKa-lCl)*AMT*
                 (exp(-exp(lKe)*TIME) - exp(-exp(lKa)*TIME))/
                 (exp(lKa)-exp(lKe)), data=origdataT,
             fixed=lKa+lKe+lCl~1,
             random=lKe~1,
             start=c(lKe=-2,lKa=1.5,lCl=-3))

sim.F <- simulate_nlme(object=F.nlme,nsim=100,psim=3,level=1,value="data.frame",data = NULL)
simdata.F <- matrix(sim.F$sim.y,ncol=100)

# Figure 7

VPCgraph(origdata,simdata.T,type="CI",N_xbin=8)+labs(title="Model 1",caption="")
VPCgraph(origdata,simdata.F,type="CI",N_xbin=8)+labs(title="Model 2",caption="")

# Figure 8

aqrVPC(origdata,simdata.T)+labs(title="Model 1",caption="")
aqrVPC(origdata,simdata.F)+labs(title="Model 2",caption="")

# Figure 9

asVPC(origdata,simdata.T,type="CI",weight_method="distance",N_xbin=8)+labs(title="Model 1",caption="")
asVPC(origdata,simdata.F,type="CI",weight_method="distance",N_xbin=8)+labs(title="Model 2",caption="")

# Figure 10

bootVPC(origdata,simdata.T,N_xbin=8)+labs(title="Model 1",caption="")
bootVPC(origdata,simdata.F,N_xbin=8)+labs(title="Model 2",caption="")

# Figure 11

coverageplot(origdata,simdata.T,conf.level=0.9,N_xbin=8)+labs(title="Model 1")
coverageplot(origdata,simdata.F,conf.level=0.9,N_xbin=8)+labs(title="Model 2")

# Figure 12

coverageDetailplot(origdata,simdata.T,predL=0.5,N_xbin=8)+labs(title="Model 1")
coverageDetailplot(origdata,simdata.F,predL=0.5,N_xbin=8)+labs(title="Model 2")

# Figure 13

coverageDetailplot(origdata,simdata.T,predL=0.8,N_xbin=8)+labs(title="Model 1")
coverageDetailplot(origdata,simdata.F,predL=0.8,N_xbin=8)+labs(title="Model 2")

# Figure 14

quantVPC(origdata,simdata.T,N_xbin=8)+labs(title="Model 1")
quantVPC(origdata,simdata.F,N_xbin=8)+labs(title="Model 2")
```
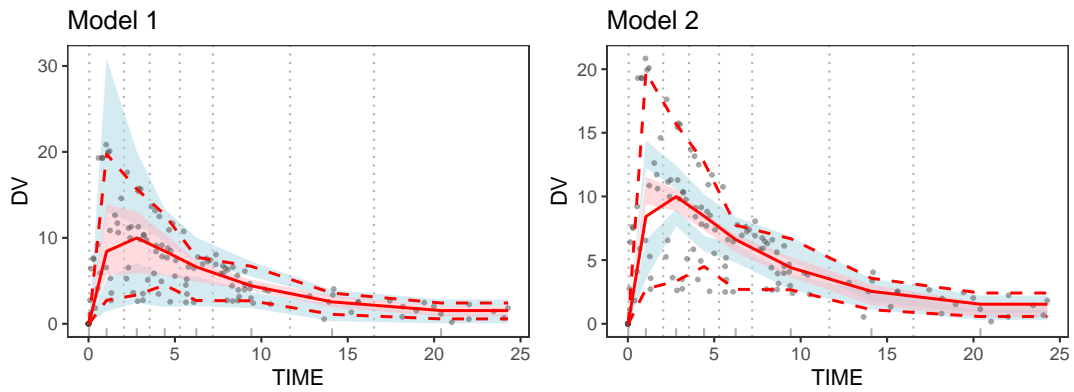
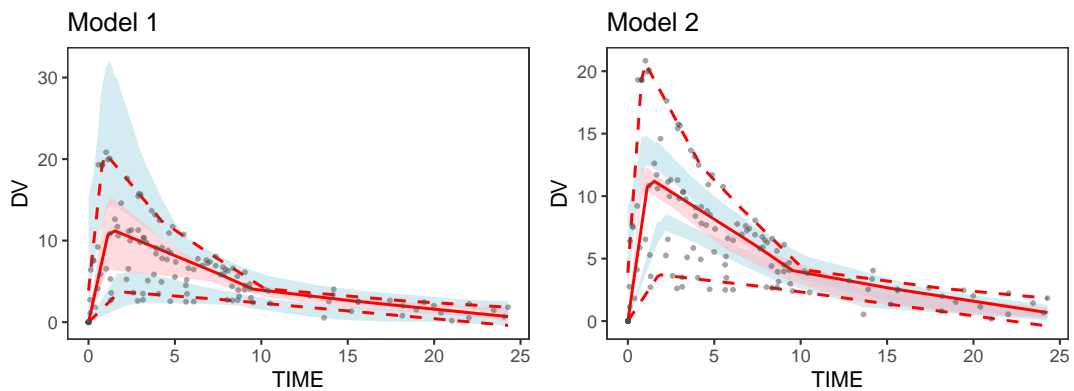**Figure 7:** The VPC plots of Model 1 and Model 2.

Figure 8: The additive quantile regression VPC plots for Model 1 and Model 2.

Figure 7 shows the VPCgraph for Model 1 and Model 2. The solid line represents the 50th percentile of the observed data, and the dashed lines represent the 10th and 90th percentiles of the observed data. In Model 1, all quantile lines (the solid and dashed lines) are in the confidence area. However, the 10th and 90th percentiles in Model 2 are mostly outside the confidence area, especially at 0 to 5 hours.

Figure 8 shows the results of aqrVPC, and Figure 9 shows the results of asVPC for Model 1 and Model 2. The additive quantile regression VPCs and the average shifted VPCs show similar patterns to the original VPCs. Model 1 shows all quantile lines in the confidence area, and the 10th and 90th percentiles in Model 2 are mostly outside the confidence area.

Figure 10 shows the results of bootVPC for Model 1 and Model 2. The solid and dashed blue lines show the 10th, 50th, and 90th percentiles of the simulated data. The solid red line represents the 50th percentile line of the observed data, and the pink areas represent the 95% confidence areas of the 50th percentile line, calculated from the bootstrap samples of the observed data. The solid blue line is in the pink area, and the two solid red and blue lines are almost identical in both models. The dashed blue lines in Model 1 cover most of the observed data. However, the dashed blue lines in Model 2 do not
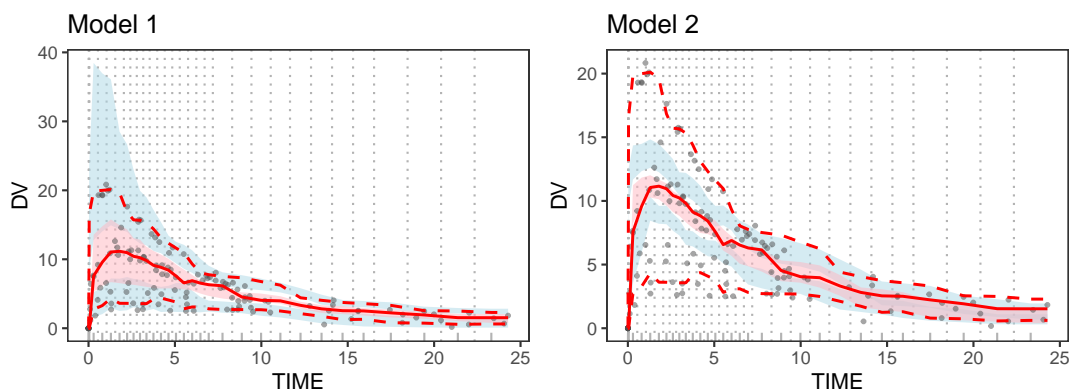

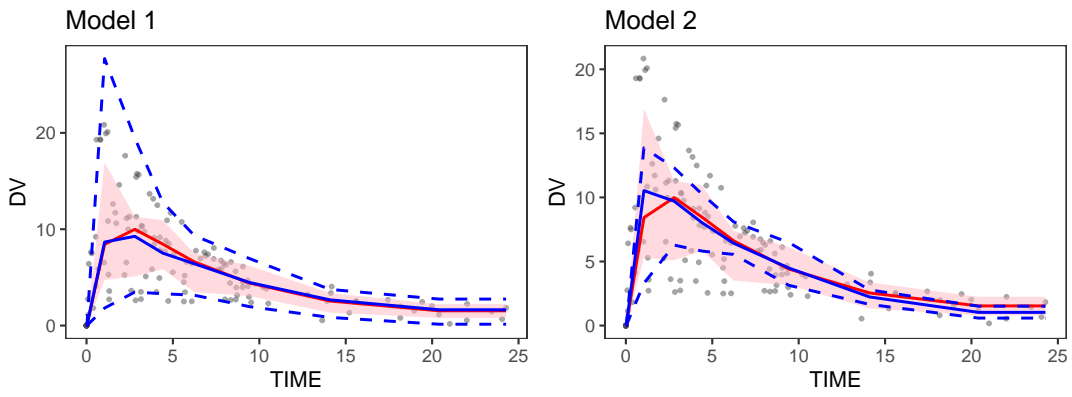**Figure 9:** The average shifted VPC plots for Model 1 and Model 2.

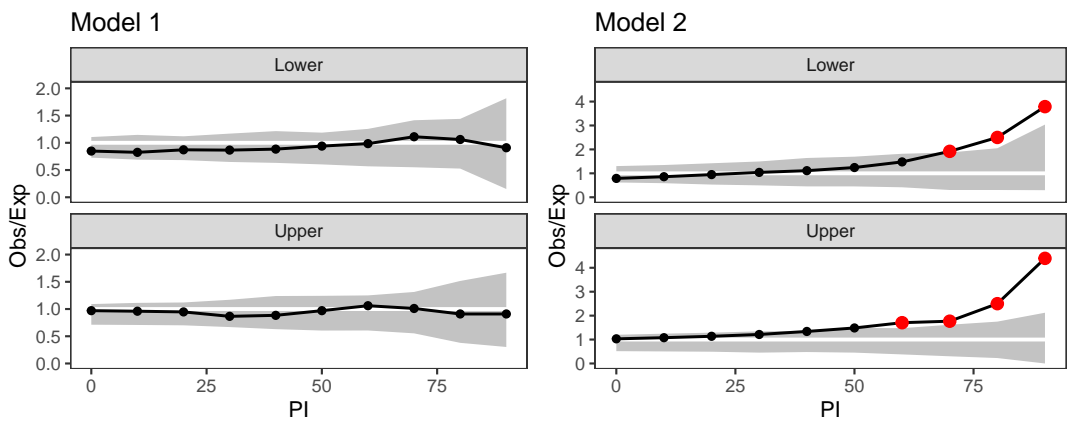**Figure 10:** The bootstrap VPC plots for Model 1 and Model 2.

Figure 11: The coverage plots for Model 1 and Model 2.

cover the observed data. Many observed data points lie outside these dashed blue lines in Model 2, especially in 0 to 10 hours.

Figure 11 shows the coverageplot results for Model 1 and Model 2. The lines are in the gray area and close to the white line in Model 1. However, the lines in Model 2 are not in the gray area, especially when the PI value is large. Figures 12 and 13 show the results of coverageDetailplot for Model 1 and Model 2 when PIs are 50% and 80%. The upper and lower percentages in both figures are close to the white points in Model 1. On the other hand, the upper percentages of the most time bins are far from the white points in Model 2, especially the time bin (3.54,5.28] when PI = 50%. When PI = 80%, most upper and lower percentages are far from the white points.

Figure 14 shows the results of quantVPC for Model 1 and Model 2. In Model 2, the right tail area (after 11 hours) looks quite different from the expected pattern. The above percentages are much larger than the below percentages.
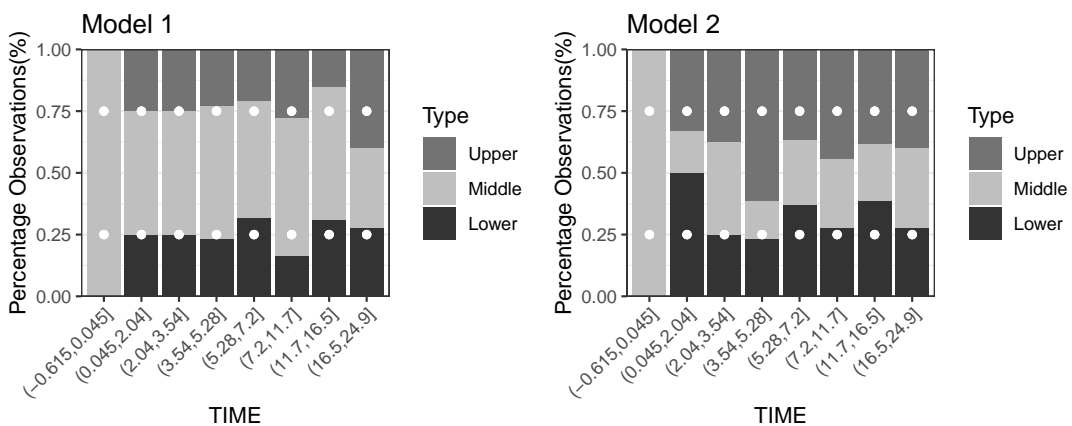

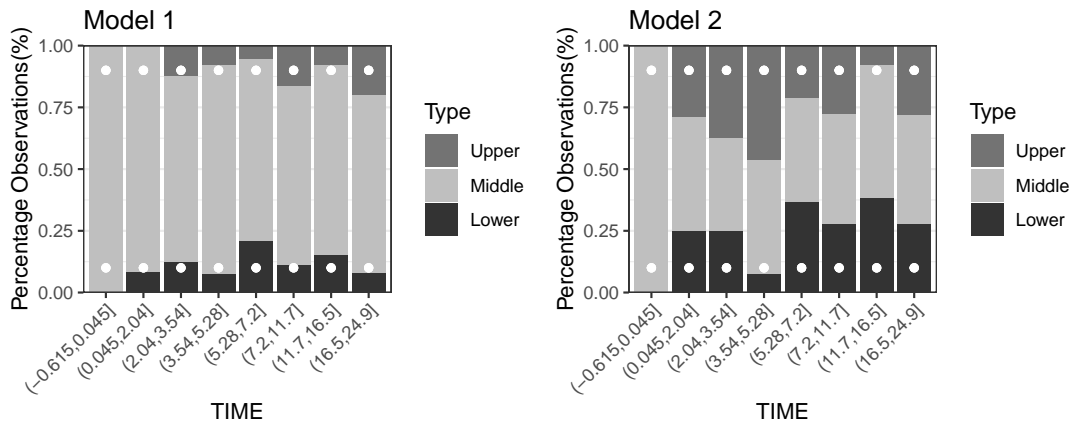**Figure 12:** The coverage detailed plots for Model 1 and Model 2 when PI=50%.

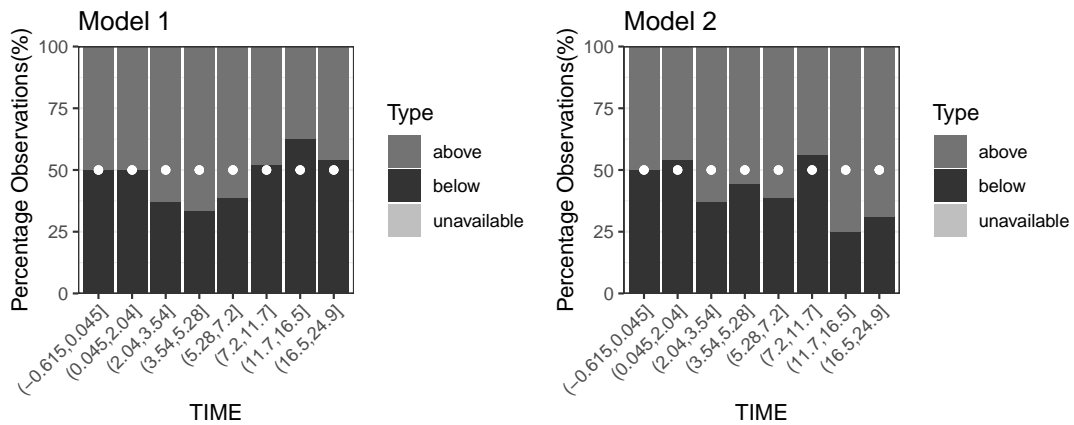**Figure 13:** The coverage detailed plots for Model 1 and Model 2 when PI=80%.

Figure 14: The quantified VPC plots for Model 1 and Model 2.

The results from Figure 7 through Figure 14 show that Model 1 explains the `origdata` quite well. However, Model 2 shows different patterns than Model 1 in most figures. We can conclude that Model 1 is better than Model 2, and treating *Ka* and *Cl* as random effects is better.

6 Summary

This paper introduces the **nlmeVPC** package. The VPC and its extensions are useful for validating the nonlinear mixed effect model. The **nlmeVPC** package provides various visual diagnostic tools for the nonlinear mixed effect model in two different approaches: validation in data space and model space. Both approaches are valuable. Validation in data space can compare the fitted model with the original data, and validation in model space provides detailed comparisons in various ways. In the **nlmeVPC** package, we also provide new approaches - asVPC and coverageDetailplot. Here, asVPC provides a more precise VPC plot, and coverageDetailplot provides the detailed feature of the fitted model that is not captured in the coverage plot. Even though the coverage plot does not show any problem with the fitted model, the coverage detailed plot can reveal the problem in a specific region (Figures 10, 11, and 12).

7 Acknowledgements

References

Bauer, Robert J. 2011. "NONMEM Users Guide Introduction to NONMEM 7.2.0." *ICON Development Solutions Ellicott City, MD*. https://www.iconplc.com/innovation/nonmem/.

Davidian, Marie. 2017. *Nonlinear Models for Repeated Measurement Data*. Routledge.

Fidler, Matthew, Justin J Wilkins, Richard Hooijmaijers, Teun M Post, Rik Schoemaker, Mirjam N Trame, Yuan Xiong, and Wenping Wang. 2019. "Nonlinear Mixed-Effects Model Development and Simulation Using Nlmixr and Related r Open-Source Packages." *CPT: Pharmacometrics & Systems Pharmacology* 8 (9): 621–33. https://doi.org/10.1002/psp4.12445.

Heus, Astrid, David W Uster, Veerle Grootaert, Nele Vermeulen, Annemie Somers, Diana Huis In't Veld, Sebastian G Wicha, and Pieter A De Cock. 2022. "Model-Informed Precision Dosing of Vancomycin via Continuous Infusion: A Clinical Fit-for-Purpose Evaluation of Published PK Models." *International Journal of Antimicrobial Agents* 59 (5): 106579. https://doi.org/10.1016/j.ijantimicag.2022.106579.

Jamsen, Kris M, Kashyap Patel, Keith Nieforth, and Carl MJ Kirkpatrick. 2018. "A Regression Approach to Visual Predictive Checks for Population Pharmacometric Models." *CPT: Pharmacometrics & Systems Pharmacology* 7 (10): 678–86. https://doi.org/10.1002/psp4.12319.

Karlsson, MATS O, and NICK Holford. 2008. "A Tutorial on Visual Predictive Checks." In *17th Meeting of the Population Approach Group in Europe, Marseille, France, Page Abstr*. Vol. 1434. https://www.page-meeting.org/?abstract=1434.

Karlsson, MO, and RM Savic. 2007. "Diagnosing Model Diagnostics." *Clinical Pharmacology & Therapeutics* 82 (1): 17–20. https://doi.org/10.1038/sj.clpt.6100241.

Keizer, R. 2021. "Vpc: Create Visual Predictive Checks. R Package Version 1.2.2." https://cran.r-project.org/web/packages/vpc/index.html.

Keizer, Ron J, MO Karlsson, and A Hooker. 2013. "Modeling and Simulation Workbench for NONMEM: Tutorial on Pirana, PsN, and Xpose." *CPT: Pharmacometrics & Systems Pharmacology* 2 (6): 1–9. https://doi.org/10.1038/psp.2013.24.

Koenker, Roger. 2023. "Quantreg: Quantile Regression. R Package Version 5.95." https://cran.r-project.org/web/packages/quantreg/index.html.

Lavielle, Marc, and Kevin Bleakley. 2011. "Automatic Data Binning for Improved Visual Diagnosis of Pharmacometric Models." *Journal of Pharmacokinetics and Pharmacodynamics* 38 (6): 861–71. https://doi.org/10.1007/s10928-011-9223-3.

Lindbom, Lars, Jakob Ribbing, and E Niclas Jonsson. 2004. "Perl-Speaks-NONMEM (PsN)—a Perl Module for NONMEM Related Programming." *Computer Methods and Programs in Biomedicine* 75 (2): 85–94. https://doi.org/10.1016/j.cmpb.2003.11.003.

Maharaj, Anil R, Huali Wu, Christoph P Hornik, and Michael Cohen-Wolkowiez. 2019. "Pitfalls of Using Numerical Predictive Checks for Population Physiologically-Based Pharmacokinetic Model Evaluation." *Journal of Pharmacokinetics and Pharmacodynamics* 46 (3): 263–72. https://doi.org/10.1007/s10928-019-09636-5.

Miguez, Fernando. 2022. "Nlraa: Nonlinear Regression for Agricultural Applications. R Package Version 1.5." https://cran.r-project.org/web/packages/nlraa/index.html.

Mudde, Saskia E, Rami Ayoun Alsoud, Aart van der Meijden, Anna M Upton, Manisha U Lotlikar, Ulrika SH Simonsson, Hannelore I Bax, and Jurriaan EM de Steenwinkel. 2022. "Predictive Modeling to Study the Treatment-Shortening Potential of Novel Tuberculosis Drug Regimens, Toward Bundling of Preclinical Data." *The Journal of Infectious Diseases* 225 (11): 1876–85. https://doi.org/10.1093/infdis/jiab101.

Nguyen, THT, M-S Mouksassi, Nicholas Holford, N Al-Huniti, I Freedman, Andrew C Hooker, J John, et al. 2017. "Model Evaluation of Continuous Data Pharmacometric Models: Metrics and Graphics." *CPT: Pharmacometrics & Systems Pharmacology* 6 (2): 87–109. https://doi.org/10.1002/psp4.12161.

Otto, ME, KR Bergmann, G Jacobs, and Michiel J van Esdonk. 2021. "Predictive Performance of Parent-Metabolite Population Pharmacokinetic Models of (s)-Ketamine in Healthy Volunteers." *European Journal of Clinical Pharmacology* 77 (8): 1181–92. https://doi.org/10.1007/s00228-021-03104-1.

Pinheiro, José C., and Douglas M. Bates. 2000. *Mixed-Effects Models in s and s-PLUS*. Springer New York. https://doi.org/10.1007/978-1-4419-0318-1.

Pinheiro, José, Douglas Bates, Saikat DebRoy, Deepayan Sarkar, Siem Heisterkamp, Bert Van Willigen, and R Maintainer. 2022. "Nlme: Linear and Nonlinear Mixed Effects Models. R Package Version 3.1-159." https://cran.r-project.org/web/packages/nlme/index.html.

Post, Teun M, Jan I Freijer, Bart A Ploeger, and Meindert Danhof. 2008. "Extensions to the Visual Predictive Check to Facilitate Model Performance Evaluation." *Journal of Pharmacokinetics and Pharmacodynamics* 35 (2): 185–202. https://doi.org/10.1007/s10928-007-9081-1.

R Core Team. 2023. "R: A Language and Environment for Statistical Computing." Vienna, Austria: R Foundation for Statistical Computing. https://www.R-project.org/.

Scott, David W. 1985. "Average Shifted Histograms: Effective Nonparametric Density Estimations

in Several Dimensions." *Annals of Statistics* 13 (3): 1024–40. https://doi.org/10.1214/aos/1176349654.

Wang, Diane D, and Shuzhong Zhang. 2012. "Standardized Visual Predictive Check Versus Visual Predictive Check for Model Evaluation." *The Journal of Clinical Pharmacology* 52 (1): 39–54. https://doi.org/10.1177/0091270010390040.

Wickham, H. 2016. *Ggplot2 : Elegance Graphics for Data Analysis*. Springer.

Wickham, Hadley, Dianne Cook, and Heike Hofmann. 2015. "Visualizing Statistical Models: Removing the Blindfold." *Statistical Analysis and Data Mining: The ASA Data Science Journal* 8 (4): 203–25. https://doi.org/10.1002/sam.11271.

Eun-Hwa Kang
Ewha Womans University
Department of Statistics, Ewha Womans University, Seoul, Korea

Myungji Ko
Ewha Womans University
Department of Statistics, Ewha Womans University, Seoul, Korea

Eun-Kyung Lee
Ewha Womans University
Department of Statistics, Ewha Womans University, Seoul, Korea
ORCiD: 0000-0003-0817-5000
lee.eunk@ewha.ac.kr