Diagnostics for nonlinear models with application to population pharmacokinetic modeling

Xiaoyong Sun
Iowa State University
Diagnostics for nonlinear models with application to population pharmacokinetic modeling

by

Xiaoyong Sun

A dissertation submitted to the graduate faculty in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Major: Bioinformatics and Computational Biology

Program of Study Committee:
Dianne Cook, Co-major Professor
Basil Nikolau, Co-major Professor
Heike Hofmann
Peng Liu
Leslie Miller

Iowa State University
Ames, Iowa
2010

Copyright © Xiaoyong Sun, 2010. All rights reserved.
DEDICATION

To my parents and wife
TABLE OF CONTENTS

LIST OF TABLES ......................................................... viii

LIST OF FIGURES ....................................................... x

ACKNOWLEDGEMENTS ..................................................... xxii

CHAPTER 1. OVERVIEW ................................................... 1

1.1 Thesis organization .................................................. 1

1.2 Pharmacokinetic data ................................................. 2

1.2.1 Data set 1 .......................................................... 2

1.2.2 Data set 2 .......................................................... 2

1.3 Pharmacokinetic models .............................................. 4

1.4 Population pharmacokinetic (PopPK) models ....................... 7

1.4.1 General formulation for PopPK models ......................... 8

1.4.2 PopPK model building ............................................ 9

1.5 Model diagnostics .................................................... 9

1.5.1 Checking assumptions and goodness of fit for population pharmacoki-
netic models .......................................................... 10

1.5.2 Diagnosing resampling statistics for population pharmacokinetic modeling 11

1.6 Software ........................................................... 11

1.6.1 NONMEM ...................................................... 12

1.6.2 Monolix .......................................................... 13

1.6.3 NLMIXED(SAS) and S/R nlme package ....................... 13

1.6.4 Other software .................................................. 14

1.7 Research problem and scope ........................................ 15
CHAPTER 2. CASE STUDIES ILLUSTRATING THE USE OF GRAPHICS IN POPULATION PHARMACOKINETIC MODEL BUILDING .

2.1 Introduction ................................................................. 16
2.2 Methods ................................................................. 17
   2.2.1 Exploratory data analysis ................................. 18
   2.2.2 Goodness of fit plots ........................................... 18
   2.2.3 Evaluate parameters and random effects ............ 19
   2.2.4 Diagnose structural models ................................. 19
   2.2.5 Diagnose residual error models ......................... 19
   2.2.6 Diagnose covariate models ................................. 20
2.3 Case study 1 .............................................................. 20
   2.3.1 Introduction ......................................................... 20
   2.3.2 Analysis ............................................................. 21
   2.3.3 Model fitting ......................................................... 22
   2.3.4 Conclusion .......................................................... 28
2.4 Case study 2 .............................................................. 29
   2.4.1 General nonlinear mixed effects model formulation ................................. 29
   2.4.2 One compartment model with zero order absorption and first order elimination ................................. 33
   2.4.3 Data description ..................................................... 37
   2.4.4 Data analysis strategy ........................................... 37
   2.4.5 Parameters .......................................................... 39
   2.4.6 Model fitting, building and selection .................. 40
   2.4.7 Diagnostic plots for model: mixed effect and fixed effect ................................. 40
   2.4.8 Diagnostic plots for model: residuals (RES) and weighted residuals (WRES) ...................... 42
   2.4.9 Covariate relationship ......................................... 42
   2.4.10 Basic diagnostics with interactive graphics ........... 43
   2.4.11 Summary of model fit findings ....................... 44
4.4.2 Configuration ................................................. 68
4.4.3 Data management ........................................... 68
4.5 Diagnostic modules ........................................... 68
  4.5.1 Exploratory data analysis (EDA) ......................... 68
  4.5.2 PK models ............................................... 69
  4.5.3 Model validation ......................................... 70
  4.5.4 Model comparison ....................................... 71
  4.5.5 Interactive graphics ..................................... 74
4.6 Example ......................................................... 74
4.7 Conclusion and future work ................................. 80
4.8 System requirements, availability and installation ....... 80

CHAPTER 5. PKREPORT: REPORT GENERATION FOR CHECKING
POPULATION PHARMACOKINETIC MODEL ASSUMPTIONS .... 85
  5.1 Introduction ............................................... 85
  5.2 Methods ..................................................... 87
    5.2.1 Exploratory data analysis ............................ 87
    5.2.2 Goodness of fit plot .................................. 88
    5.2.3 Evaluate parameters and random effects .............. 88
    5.2.4 Diagnose structural models .......................... 88
    5.2.5 Diagnose residual error models ....................... 89
    5.2.6 Diagnose covariate models .......................... 89
  5.3 Software implementation ................................... 90
    5.3.1 Metric system ......................................... 90
    5.3.2 Configuration ......................................... 90
    5.3.3 Architecture description and features ............... 91
  5.4 Demonstration ............................................... 92
  5.5 Future ......................................................... 94
  5.6 Availability ................................................ 94
### CHAPTER 6. THE EFFECTS OF COVARIATE CORRELATION ON POPULATION PHARMACOKINETIC COVARIATE MODELING

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.1</td>
<td>Introduction</td>
<td>98</td>
</tr>
<tr>
<td>6.2</td>
<td>Methods</td>
<td>100</td>
</tr>
<tr>
<td>6.2.1</td>
<td>Covariate data generation</td>
<td>100</td>
</tr>
<tr>
<td>6.2.2</td>
<td>Correlation structure</td>
<td>100</td>
</tr>
<tr>
<td>6.2.3</td>
<td>Pharmacokinetic data simulation</td>
<td>101</td>
</tr>
<tr>
<td>6.2.4</td>
<td>Covariate selection method</td>
<td>102</td>
</tr>
<tr>
<td>6.2.5</td>
<td>Performance measures</td>
<td>102</td>
</tr>
<tr>
<td>6.3</td>
<td>Results</td>
<td>103</td>
</tr>
<tr>
<td>6.3.1</td>
<td>Correlation structure of covariates</td>
<td>103</td>
</tr>
<tr>
<td>6.3.2</td>
<td>Comparison of covariate selection methods</td>
<td>106</td>
</tr>
<tr>
<td>6.4</td>
<td>Discussion</td>
<td>107</td>
</tr>
<tr>
<td>6.5</td>
<td>Conclusions</td>
<td>110</td>
</tr>
</tbody>
</table>

### CHAPTER 7. GENERAL CONCLUSION AND FUTURE WORK

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.1</td>
<td>Methodology</td>
<td>117</td>
</tr>
<tr>
<td>7.2</td>
<td>Software</td>
<td>118</td>
</tr>
<tr>
<td>7.2.1</td>
<td>PKgraph</td>
<td>118</td>
</tr>
<tr>
<td>7.2.2</td>
<td>PKreport</td>
<td>118</td>
</tr>
<tr>
<td>7.3</td>
<td>Future work</td>
<td>118</td>
</tr>
</tbody>
</table>

### APPENDIX A. USER GUIDE FOR PKGRAPH PACKAGE

- Page 120

### BIBLIOGRAPHY

- Page 154
LIST OF TABLES

Table 1.1 Variable description for data set 1. . . . . . . . . . . . . . . . . . . . . 2
Table 1.2 Sample data for subject 1 from data set 1. . . . . . . . . . . . . . . . . 3
Table 1.3 Variable description for data set 2 . . . . . . . . . . . . . . . . . . . . . 3
Table 1.4 Sample data for subject 1 from data set 2. . . . . . . . . . . . . . . . . 4
Table 2.1 Dose amount for each subject. The minimum amount is 3.10 for subject
9, and the maximum amount is 5.86 for subject 5. . . . . . . . . . . . . . . . . . 21
Table 2.2 Weight for each subject. The minimum weight is 54.6, and the maxi-
mum weight is 84.6. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 24
Table 2.3 Time summary for each subject. . . . . . . . . . . . . . . . . . . . . . . 27
Table 2.4 Concentration summary for each subject. . . . . . . . . . . . . . . . . . 28
Table 2.5 Summary for nonlinear mixed-effect model fit by maximum likelihood. 28
Table 2.6 AIC, BIC and logLik values. . . . . . . . . . . . . . . . . . . . . . . . . 29
Table 5.1 Package metric system. The first column (Package variable) contains
default variable names. These default variable names match NONMEM
naming system. The second column explains the details of these variables. 91
Table 6.1 Correlation structure 1: one true covariate (WT). A increased by 0.1
from 0 to 1, resulting in 10 values. . . . . . . . . . . . . . . . . . . . . . . . . . . . . 100
Table 6.2 Correlation structure 2: one true covariate (WT). A increased by 0.1
from 0 to 1 and B decreased by 0.1 from 1 to 0 concurrently, resulting
in 10 combinations. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 100
Table 6.3  Correlation structure 3: two true covariates (WT and AGE). $A$ increased by 0.1 from 0 to 1, resulting in 10 values. ........................................ 101

Table 6.4  Correlation structure 4: two true covariates (WT and AGE). $A$ increased by 0.1 from 0 to 1 and $B$ decreased by 0.1 from 1 to 0 concurrently, resulting in 10 combinations. ........................................ 101

A.1  Package metric system ......................................................... 134
LIST OF FIGURES

Figure 1.1 One compartment model with bolus intravenous injection. \( A \): the amount of drug in the body. \( A = C \times V \), where \( C \) is the concentration of drug in body fluids and \( V \) is the drug distribution volume; \( k_{el} \): the elimination rate constant, i.e., the first order rate constant for drug elimination from the body. 5

Figure 1.2 Two compartment model with rapid intravenous injection. \( A_1 \): the amount of drug in the central compartment. \( A_1 = C_1 \times V_1 \), where \( C_1 \) is the concentration of drug in the central compartment and \( V_1 \) is the drug distribution volume in the central compartment; \( A_2 \): the amount of drug in the peripheral compartment; \( k_{el} \): the elimination rate constant, i.e., the first order rate constant for drug elimination from the body; \( k_{12} \): the rate constant for transfer of drug from compartment 1 to compartment 2; \( k_{21} \): the rate constant for transfer of drug from compartment 2 to compartment 1. 6

Figure 2.1 Histogram of dose. Doses range between 3 and 6 mg/kg. 22
Figure 2.2 Histogram of weight in kilograms. 23
Figure 2.3 Scatterplot of Dose and Weight. Dose has mg/kg unit, which comes from total drug amount divided by weight. 25
Figure 2.4 Time histogram. Almost all concentration data were measured between 0 and 15 hours, except that one was measured around 24 hours after single dose. 26
Figure 2.5  Concentration histogram. There are two peaks, one around 0, and the other between 4 and 7. ................................................................. 26

Figure 2.6  Scatterplot for maximum concentration versus weight. There is a negative linear relationship between weight and maximum concentration for each subject. ................................................................. 29

Figure 2.7  Time plot of concentration versus time for each subject. The highest concentration peak is 11.4 for subject 5. ................................. 30

Figure 2.8  Observed values versus time for each subject, ordered by maximum concentration. ................................................................. 31

Figure 2.9  Observed values versus time for each subject, ordered by time reaching maximum concentration. ................................................. 32

Figure 2.10  Left figure: Time plot for concentration with different dose, same parameter values, lke, lka, lCl; Right figure: Time plot for concentration with different lke, same parameter values, dose, lka, lCl. ............... 33

Figure 2.11  Left figure: Time plot for concentration with different lka, same parameter values, dose, lke, lCl; Right figure: Time plot for concentration with different lCl, same parameter values, dose, lke, lka. .......................... 33

Figure 2.12  Mixed effects model fitted values (blue line) and observed concentration (black points) versus Time. Subjects 5 and 9 are substantially underfit. 34

Figure 2.13  Dot plot for maximum fixed effect by each subject. .................. 34

Figure 2.14  Time versus Residual. .......................................................... 35

Figure 2.15  Fixed effects (red line) and observed concentration (black points) versus Time. ................................................................. 35

Figure 2.16  Mixed effects (blue line), fixed effects (red line), and observed values (black points) versus Time. ................................................. 36

Figure 2.17  Time plots for concentration versus time, for patient 1 to patient 10. 37
Figure 2.18  Left figure: A few more male(1) patients than female(0) ones; Right figure: Two different dose amounts, 100mg and 250mg have been given to the equal number of patients. 

Figure 2.19  Left figure: Most patients weigh between 20 to 80kg with several very heavy patients around 100kg; Right figure: Age ranges from 10 to 80.

Figure 2.20  Left figure: Time plot for concentration with different dose, same parameter values, CL, V; Right figure: Time plot for concentration with different KE (elimination rate), with same dose.

Figure 2.21  Left figure: Time plot for concentration with different CL(clearance), same parameter values, DOSE, V; Right figure: Time plot for concentration with different V(volume of distribution), same parameter values, DOSE, CL.

Figure 2.22  Fixed effects versus time with observed value (black points) and fixed effect/population effect (red line).

Figure 2.23  Fixed effects (red) and mixed effects (blue) model versus concentration for each patient.

Figure 2.24  Fixed effect (blue), and mixed effect (black) versus concentration.

Figure 2.25  Time versus random residual. Random residual is not randomly distributed.

Figure 2.26  Left figure: Age versus random effect of clearance; Right figure: Weight versus random effect of clearance. There is some positive association between weight and random effect of clearance.

Figure 2.27  Diagnostics plot. Patients with the largest response have low weight.
Figure 3.1  Multiple simulated data sets for resampling statistics. In the first simulation data set (sim1), when the subject is absent from the simulation no parameter is estimated, so a missing value (NA) is generated. One of these tables is generated for each parameter (CL, V and Ke). We can consider this to be a multivariate data set, which we will use to examine the influence of each subject and assess the variability in the parameter estimates.

Figure 3.2  Plots to examine the resampling designs (left) for case deletion, (right) bootstrap. In each case the run id is plotted horizontally, and the vertical axis displays which cases were in or out of the sample. For the case deletion design we plot the id number of the subject deleted for the run. Lines connect sequential runs. If the deletion was done sequentially by PsN then we would see a straight line from (1,1) to (100, 100). This is not the case: in the first run, the first patient was deleted, in the second run, the 10th patient was deleted, and in the third run, the 100th patient was deleted.

Figure 3.3  Distribution of weight for the samples: (left) case deletion statistics (right) bootstrap. Each sample is represented using a different color. The density is similar across the case deletion samples. There is a lot more variation in the distribution of weight in the bootstrap samples, as is to be expected.

Figure 3.4  Distribution of clearance (CL) in (left) case deletion statistics, and (right) bootstrap samples. There is more variability in the bootstrap samples, as expected. In the case deletion samples, one run had a noticeably higher second peak than those in other samples.
Figure 3.5 Parallel coordinate plot for diagnosing case deletion runs for the clearance estimates. Each line in the figure connects values for each subject, across runs. The clearance estimates change substantially for two runs, suggesting there are two patients which have undue influence on the model fit. .......................... 57

Figure 3.6 MDS plot for diagnosing case deletion runs for the clearance estimates. The IDs in the figure match the case deletion run ID, and it means that the patient with this ID was deleted. The plot indicates that subjects 52 and 20 were influential on clearance estimates. .................. 58

Figure 3.7 Interactive graphics for case deletion diagnostics: (left) parallel coordinate plot, (middle) time series plot for all patients, and (right) MDS plot. Two outliers with case deletion ID: 52 and 20 are brushed in the MDS plot, and the corresponding elements of the other two plots are colored accordingly. The two outliers in the MDS plot correspond to the runs in the parallel coordinates plot indicating the same two subjects as being influential. The time series plots would not have suggested that these two subjects are influential. .................. 59

Figure 3.8 Interactive graphics for bootstrap statistics. In the left side plot variance of clearance estimates is plotted against subject ID. There is one subject with very large variance, which is brushed (blue). The plot is linked to the time series plot for all subjects (right). This patient has very low concentration. .................. 60

Figure 4.1 Software architecture of PKgraph. Blue indicates the management interface and basic modules, and green the graphical interface accessing the diagnostic modules. Data can be model fit results from NONMEM, Monolix, SAS or R, and users can manage the data through graphical user interface to perform model diagnostics. Figures can be generated and save based on two R graphic packages: lattice and ggplot2. .... 64
Figure 4.2  Main interface of PKgraph. Area 1 has menu items linking to the basic management modules, and the diagnostic modules. Area 2 shows current files. Area 3 is the data area, showing model fit from NONMEM. Area 4 displays the progress of loading data.

Figure 4.3  Graph interface of PKgraph. It shows the histogram for concentration generated from lattice graphic package. Area 1 is parameter setup area for figures. Area 2 shows tool bar (four buttons) for the following four functions: saving, opening the plot in ggobi for interaction on the plot, synchronizing subset selection from ggobi to the data in R, and closing ggobi. Area 3 is the data area, showing model fit from NONMEM. Area 4 displays the figure, and multiple figures if more than one are created.

Figure 4.4  Graph interface for case deletion diagnostics. The left area is the parameter setup area. The right area is the figure area. In this figure, a multidimensional scaling plot is generated for visualizing case deletion runs. The IDs in the figure match the case deletion run ID, and it means that the patient with this ID was deleted, and model was re-fit with NONMEM.

Figure 4.5  Model comparison. Top figure: “histogram comparison” for comparing distributions of “CL” from two models. The left area is parameter setup area. In the setup, clearance (CL.x, CL.y) from two models are compared. The figure shows that there is not much difference between two models in the distribution of CL. Bottom figure: “scatter plot comparison” for comparing values of “CL” from two models. The figure shows that two models are quite similar for small “CL” values, while there is some difference for high “CL” values.
Figure 4.6 Exploratory data analysis. The left figure is the parameter setup area. 
x, y are set as TIME, CONC; time variable is set as TIME; ID variable 
is set as ID. The right figure is figure area, showing concentration versus 
time (hr). ................................................................. 75

Figure 4.7 Exploratory data analysis. The left figure is the parameter setup area. 
x, y are set as TIME, CONC; time variable is set as TIME; ID variable 
is set as ID. The right figure is figure area, showing concentration versus 
time (hr) conditioned on ISM (0: female, 1: male). ..................... 76

Figure 4.8 The model results are loaded into ggobi for interactive exploration. The 
plot at left shows the concentration versus time for all subjects, and the 
plot at right shows two of the covariates, weight versus age. The two 
plots are linked by subject ID. Points near the peak concentration are 
brushed (blue) and from the covariate plot we can see that these two 
subjects are both light and relatively old. .............................. 77

Figure 4.9 Exploratory data analysis. After clicking the third button in toolbar 
(above figure area), the detailed information for two patients (ID: 55 
and 58) is selected for investigation. ................................. 78

Figure 4.10 Structural model diagnostics. The left area is parameter setup area. It 
has default parameter setup, and users can change them to their specific 
interests. The right area is figure area. In this figure, it shows weighted 
residual (WRES) versus predicted values (PRED). The residuals are 
randomly distributed. ...................................................... 79
Figure 4.11 Influence analysis for case deletion diagnostics. The left area is parameter setup area. The right area is figure area. In this figure, it is parallel coordinate plot for case deletion diagnostics. The y axis is case deletion run. Since there are 100 patients, there are 100 case deletion runs. Each run means one patient ID is deleted, model is re-fit, and parameter (CL here) for this patient is calculated. X axis is the parameter (CL) value. Each line in the figure matches parameters (CL) for one patient across multiple case deletion runs. When two patients were deleted in two case deletion runs (marked with red arrows), the values of parameter CL of all other patients changed clearly compared with those in other runs.

Parameter (CL) value for some patients increases, while other decreases.

Figure 4.12 Influence analysis for case deletion diagnostics: linking results from multidimensional scaling and parallel coordinate plots. The plot at left shows the parallel coordinate plot (x=simulation ID, y=clearance), and the plot at right shows multidimensional scaling plot. The two plots are linked by case deletion ID. One outlier point (case deletion ID: 52) is brushed (blue). When this subject is left out, the estimated clearance values for other subjects are lower.

Figure 4.13 Density plot comparison for comparing distributions of clearance (CL) from two models (blue, pink), separately by gender (ISM: 0=female, 1=male). The distribution of CL for two models is slightly different, for both genders: model 2 has slightly higher density of values at high clearance, with a higher peak in the second mode.

Figure 5.1 Software architecture of PKreport. It has seven main functions: 1) match metrics; 2) configure figures; 3) generate figures; 4) display results; 5) generate R scripts; 6) modify figure; 7) clean archives. The first two functions set up working environment, and the other functions help to generate related figures.
Figure 5.2 Diagnostic results shown in web browser. In this example, the categories in web browser match three diagnostic steps: exploratory data analysis (PKreport.1), goodness of fit plots (PKreport.2) and covariate model diagnostics (PKreport.6). The last one contains all related R scripts.

Figure 5.3 Figure archives. In this example, univar and bivar folders match exploratory data analysis; gof folder matches goodness of fit plots; cov folder matches covariate model diagnostics. These folders contain all related diagnostic figures. PKcode.text has all related R scripts for these figures.

Figure 5.4 Figures for covariate model diagnostics. Left figure: scatterplot of individual prediction (IPRE) versus concentration (CONC) generated with lattice package. Right figure: scatterplot of individual prediction (IPRE) versus concentration (CONC) generated with ggplot2 package.

Figure 5.5 R scripts generated for goodness of fit plots. The figures generated with these R scripts are stored in gof folder. In this example, the R scripts are for two scatterplots (IPRE versus CONC and DOSE versus IPRE) and for each plot, the R scripts for both lattice and ggplot2 packages are generated.

Figure 6.1 Time versus concentration for four simulated data sets. For correlation structure 1 and 3, correlation between WT and HT is 0.1; for correlation structure 2 and 4, correlation between WT and HT is 0.1 and correlation between WT and Age is 0.9. There is no clear difference among these four correlation structures.

Figure 6.2 Weight versus concentration for four simulated data sets. For correlation structure 1 and 3, correlation between WT and HT is 0.1; for correlation structure 2 and 4, correlation between WT and HT is 0.1 and correlation between WT and Age is 0.9.
Figure 6.3 Height versus concentration for four simulated data sets. For correlation structure 1 and 3, correlation between WT and HT is 0.1; for correlation structure 2 and 4, correlation between WT and HT is 0.1 and correlation between WT and Age is 0.9. 105

Figure 6.4 Age versus concentration for four simulated data sets. For correlation structure 1 and 3, correlation between WT and HT is 0.1; for correlation structure 2 and 4, correlation between WT and HT is 0.1 and correlation between WT and Age is 0.9. 106

Figure 6.5 Alkaline phosphatase versus concentration for four simulated data sets. For correlation structure 1 and 3, correlation between WT and HT is 0.1; for correlation structure 2 and 4, correlation between WT and HT is 0.1 and correlation between WT and Age is 0.9. 107

Figure 6.6 Scatterplot matrix for correlation structure 1 and 3. Correlation between WT and HT is 0.1. Conc: concentration. WT: weight. HT: height. AP: alkaline phosphatase. 108

Figure 6.7 Scatterplot matrix for correlation structure 2 and 4. Correlation between WT and HT is 0.1 and correlation between WT and Age is 0.9. Conc: concentration. WT: weight. HT: height. AP: alkaline phosphatase. 109

Figure 6.8 Covariate selection for one true covariate and correlation structure 1 and 2. X is the correlation between weight (WT) and height (HT); Y is the times to identify the true covariate with GAM method in 10 data sets. Left figure: the times to identify the true covariate varied between 3 and 6. When the correlation reached 0.9, GAM method identified the true covariate for only two times in 10 data sets. Right figure: the times to identify the true covariate jumped to 8, then dropped to 4 and stayed around 6 when the correlation between WT and HT increased and the correlation between WT and AGE decreased. 111
Figure 6.9  Covariate selection for two true covariates and correlation structure 3 and 4. $X$ is the correlation between weight (WT) and height (HT); $Y$ is the times to identify the true covariate with GAM method in 10 data sets. Left figure: the maximum times to identify correct covariates were 3.5 at correlation 0.1 and the minimum times were 2 at correlation 0.6. Right figure: the maximum times and minimum times were 4.5 at correlation 0.2, and 1.5 at correlation 0.9, respectively.  

Figure 6.10  Comparison of performance of GAM in four correlation structures. $X$ is correlation structure; $Y$ is the times to identify the true covariate with GAM method in 10 data sets. GAM performed best in correlation structure 2. Correlation structure 1 resulted in highest variability.  

Figure 6.11  Performance of three algorithms for covariate selection (correlation structure 1 and 2). There was no obvious difference in three algorithms for both left figure and right figure. $gam$: generalized additive model; $gbm$: gradient boosting; $rf$: random forest.  

Figure 6.12  Performance of three algorithms for covariate selection (correlation structure 3 and 4). Gradient boosting and random forest both performed better than GAM. $gam$: generalized additive model; $gbm$: gradient boosting; $rf$: random forest.  

Figure 6.13  Performance of combination of correlation structures and methods. For correlation structure 1 and 2, all three algorithms performed very similarly with some difference in variability; for correlation structure 3 and 4, both gradient boosting and random forest performed better than GAM. $gam$: generalized additive model; $gbm$: gradient boosting; $rf$: random forest.  

A.1  Main interface of PKgraph  

A.2  Graph interface of PKgraph  

A.3  Menu items in Project
A.4 Menu items in *Configure* ........................................... 127
A.5 Menu items in *Data management* .................................. 128
A.6 Menu items in *Exploratory data analysis* .......................... 129
A.7 Menu items in *PK models* ........................................... 132
A.8 *Configure model result* in *PK models* .............................. 133
A.9 *Parameters in PK models* ............................................ 135
A.10 Menu items in *Model validation* ..................................... 136
A.11 Influence analysis summary (PsN) .................................. 137
A.12 Multiple NONMEM runs for case deletion diagnostics .......... 138
A.13 Parameters and results for case deletion diagnostics ............ 139
A.14 Bootstrap summary (PsN) ............................................ 140
A.15 Multiple NONMEM runs for bootstrap ............................... 141
A.16 Parameters and results for bootstrap visualization ............... 142
A.17 Menu items in *Model comparison* .................................. 143
A.18 *Select datasets in Model comparison* ................................ 144
A.19 *Configure mapping in Model comparison* .......................... 145
A.20 Current data set for *Model comparison* .............................. 146
A.21 *histogram comparison for Model comparison* ..................... 147
A.22 Exploratory data analysis. Peak is identified with brushing. This patient is from light weight and middle age group. .............. 148
A.23 Exploratory data analysis. The detailed information for this patient is selected for investigation. ........................................ 149
A.24 Structural model diagnostics. ........................................ 150
A.25 Influence analysis .................................................... 151
A.26 Influence analysis: linking results from multidimensional scaling and parallel coordinate plots. ..................................... 152
A.27 Histogram comparison for comparing distributions of CL from two models. 153
ACKNOWLEDGEMENTS

I would like to thank my major professor, Dr. Di Cook for her guidance and support. She trained me to learn independently, think critically and work professionally. By providing great opportunities to teach in the school as a teaching assistant and work in Novartis as a summer intern, she helped me to plan my career and guided me to understand the teaching and research. The flexible research environment, refreshing tea break, inspiring discussion and lovely statistical graphics group will always be part of beautiful memories of Iowa State University. I would like to thank Dr. Heike Hofmann for providing constructive suggestions in the research and helpful advices as a friend during the study. I would like to thank Dr. David James for teaching me to write elegant and efficient R code, training me as a professional statistician and serving as a wonderful mentor in Novartis. I would like to thank Dr. Zhijun Wu for showing me the whole picture of research in a short and fruitful rotation. I am grateful to my committee members for their support and guidance. I would also like to thank Bioinformatics and Computational Biology program, Plant Science Institute, and Department of Statistics for financial support.

I own a special thanks to Dr. Karin Dorman for serving as my mentor in Preparing Future Faculty program. She taught me how to teach efficiently and write clearly. During regular discussion every week, she explained to me the details of how to hunt for the job, how to balance life and how to apply for the grant. These experiences help me build a solid foundation for my career. I would also like to thank Dr. Volker Brendel for teaching me how to organize an advanced bioinformatics course for graduate students and Dr. Peng Liu for showing me how to manage an introductory statistics class for undergraduates.

I would express my gratitude to Dr. Kai Wu who collaborates with me on several chapters
of this thesis and provides his expertise and advices on pharmacology. I am very grateful to
my lab mates and classmates, Michael Lawrence, Hadley Wickham, Michael Zimmermann and
Li Xue for their help, advices and support. I am indebted to my friends, Zhaoyuan Hu and
Feng Fang for drawing my attention to Bioinformatics, the field that I enjoy so much.

Finally, I would like to thank my parents and my wife. The love and support from my
parents motivates me to run happily and energetically. The encouragement and assistance
from my wife accompanies me to school in numberless days and nights.
CHAPTER 1. OVERVIEW

1.1 Thesis organization

The main theme of this thesis is to develop better diagnostics for pharmacokinetic models. It consists of two components: methodology and software. The methodology component includes chapter 2, 3, and 6, focusing on diagnostic methods for population pharmacokinetic models. The software component consists of chapter 4 and 5 and emphasizes two related software: PKgraph and PKreport derived from methodology component.

Chapter 1 provides a general overview and research scope for this thesis. It covers explanations of pharmacokinetic (PK) data, PK models, Population PK (PopPK) models, model diagnostics, related software, and data used for examples in the thesis. Chapter 2 describes methods to check model assumptions and goodness of fit during model building. Two case studies were utilized to illustrate these methods for exploration data analysis, goodness of fit, parameter and random effects evaluation, structural model diagnostics, residual model diagnostics and covariate model diagnostics. Chapter 3 develops methods for diagnosing PopPK models by visualizing resampling statistics. In this work, we adapt visual methods from multivariate analysis, parallel coordinate plots and multidimensional scaling. Chapter 4 describes an R package, PKgraph, that was developed as part of this thesis. This software provides a graphical user interface for PopPK model diagnosis. It also provides an integrated and comprehensive platform for exploration of PK data. Chapter 5 describes an R package, PKreport, which is for automatic report generation, for checking model assumptions, visualizing data and diagnosing models. Chapter 6 presents some preliminary work on covariate selection. We explore the effects of covariate correlation on covariate model building and compare performance of three algorithms for covariate selection, including generalized additive model (GAM),
gradient boosting and random forest. Finally, Chapter 7 gives conclusions and plans for future work.

1.2 Pharmacokinetic data

Two data sets were used in this thesis are as follows.

1.2.1 Data set 1

This example, Theoph in NONMEM (Boeckmann et al., 1994), is from a study of the pharmacokinetics of drug theophylline, which was used to treat asthma. This is a short-duration study (about 24 hours) and only one dose per subject is given. There were 12 subjects in this research, and each subject was measured 11 times post-dose. Table 1.1 describes the variables in the data. Sample data for one subject are shown in Table 1.2.

<table>
<thead>
<tr>
<th>Data variables</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subject</td>
<td>unique ID for each subject</td>
</tr>
<tr>
<td>Time</td>
<td>clock time (hour)</td>
</tr>
<tr>
<td>Wt</td>
<td>weight (kilogram)</td>
</tr>
<tr>
<td>Dose</td>
<td>dose of theophylline administered orally to the subject (mg/kg)</td>
</tr>
<tr>
<td>conc</td>
<td>theophylline concentration in the sample (mg/L)</td>
</tr>
</tbody>
</table>

Table 1.1 Variable description for data set 1.

1.2.2 Data set 2

This is an example used to illustrate PK/PD modeling using NONMEM downloaded at https://www.accp1.org/pharmacometrics/Datafile/CS1_IV1EST_PAR.csv. There were 100 subjects, and each individual was sampled with 0, 0.25, 0.5, 0.75, 1, 1.5, 2, 2.5, 3, 4, 6, 8, 12, 16, 20, 24 hours post-dose. The variables are described in Table 1.3. Sample data for one subject are shown in Table 1.4.
<table>
<thead>
<tr>
<th>Subject</th>
<th>Wt</th>
<th>Dose</th>
<th>Time</th>
<th>conc</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>79.6</td>
<td>4.02</td>
<td>0</td>
<td>0.74</td>
</tr>
<tr>
<td>1</td>
<td>79.6</td>
<td>4.02</td>
<td>0.25</td>
<td>2.84</td>
</tr>
<tr>
<td>1</td>
<td>79.6</td>
<td>4.02</td>
<td>0.57</td>
<td>6.57</td>
</tr>
<tr>
<td>1</td>
<td>79.6</td>
<td>4.02</td>
<td>1.12</td>
<td>10.5</td>
</tr>
<tr>
<td>1</td>
<td>79.6</td>
<td>4.02</td>
<td>2.02</td>
<td>9.66</td>
</tr>
<tr>
<td>1</td>
<td>79.6</td>
<td>4.02</td>
<td>3.82</td>
<td>8.58</td>
</tr>
<tr>
<td>1</td>
<td>79.6</td>
<td>4.02</td>
<td>5.1</td>
<td>8.36</td>
</tr>
<tr>
<td>1</td>
<td>79.6</td>
<td>4.02</td>
<td>7.03</td>
<td>7.47</td>
</tr>
<tr>
<td>1</td>
<td>79.6</td>
<td>4.02</td>
<td>9.05</td>
<td>6.89</td>
</tr>
<tr>
<td>1</td>
<td>79.6</td>
<td>4.02</td>
<td>12.12</td>
<td>5.94</td>
</tr>
<tr>
<td>1</td>
<td>79.6</td>
<td>4.02</td>
<td>24.37</td>
<td>3.28</td>
</tr>
</tbody>
</table>

Table 1.2 Sample data for subject 1 from data set 1.

<table>
<thead>
<tr>
<th>Data variables</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TIME</td>
<td>Time (hour)</td>
</tr>
<tr>
<td>DV/CONC</td>
<td>Dependant variable (concentration in plasma, ug/ml)</td>
</tr>
<tr>
<td>AMT</td>
<td>Dose (mg)</td>
</tr>
<tr>
<td>MDV</td>
<td>Missing dependant variable (MDV = 1 when DV = 0 or missing(.))</td>
</tr>
<tr>
<td>SCR</td>
<td>Serum creatinine</td>
</tr>
<tr>
<td>CLCR</td>
<td>Creatinine clearance (ml/min)</td>
</tr>
<tr>
<td>AGE</td>
<td>Age (year)</td>
</tr>
<tr>
<td>WT</td>
<td>Body weight (kilogram)</td>
</tr>
<tr>
<td>ISM</td>
<td>Gender (0 female, 1 male)</td>
</tr>
</tbody>
</table>

Table 1.3 Variable description for data set 2
<table>
<thead>
<tr>
<th>CID</th>
<th>TIME</th>
<th>CONC</th>
<th>AMT</th>
<th>DOSE</th>
<th>MDV</th>
<th>AGE</th>
<th>WT</th>
<th>SCR</th>
<th>ISM</th>
<th>CLCR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>100</td>
<td>100</td>
<td>1</td>
<td>34.823</td>
<td>38.212</td>
<td>1.1129</td>
<td>0</td>
<td>42.635</td>
</tr>
<tr>
<td>1</td>
<td>0.25</td>
<td>13.026</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>34.823</td>
<td>38.212</td>
<td>1.1129</td>
<td>0</td>
<td>42.635</td>
</tr>
<tr>
<td>1</td>
<td>0.5</td>
<td>14.984</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>34.823</td>
<td>38.212</td>
<td>1.1129</td>
<td>0</td>
<td>42.635</td>
</tr>
<tr>
<td>1</td>
<td>0.75</td>
<td>14.16</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>34.823</td>
<td>38.212</td>
<td>1.1129</td>
<td>0</td>
<td>42.635</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>19.316</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>34.823</td>
<td>38.212</td>
<td>1.1129</td>
<td>0</td>
<td>42.635</td>
</tr>
<tr>
<td>1</td>
<td>1.5</td>
<td>13.146</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>34.823</td>
<td>38.212</td>
<td>1.1129</td>
<td>0</td>
<td>42.635</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>12.921</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>34.823</td>
<td>38.212</td>
<td>1.1129</td>
<td>0</td>
<td>42.635</td>
</tr>
<tr>
<td>1</td>
<td>2.5</td>
<td>8.485</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>34.823</td>
<td>38.212</td>
<td>1.1129</td>
<td>0</td>
<td>42.635</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>16.437</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>34.823</td>
<td>38.212</td>
<td>1.1129</td>
<td>0</td>
<td>42.635</td>
</tr>
<tr>
<td>1</td>
<td>4</td>
<td>10.724</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>34.823</td>
<td>38.212</td>
<td>1.1129</td>
<td>0</td>
<td>42.635</td>
</tr>
<tr>
<td>1</td>
<td>6</td>
<td>8.7352</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>34.823</td>
<td>38.212</td>
<td>1.1129</td>
<td>0</td>
<td>42.635</td>
</tr>
<tr>
<td>1</td>
<td>8</td>
<td>7.697</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>34.823</td>
<td>38.212</td>
<td>1.1129</td>
<td>0</td>
<td>42.635</td>
</tr>
<tr>
<td>1</td>
<td>12</td>
<td>4.479</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>34.823</td>
<td>38.212</td>
<td>1.1129</td>
<td>0</td>
<td>42.635</td>
</tr>
<tr>
<td>1</td>
<td>16</td>
<td>2.4183</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>34.823</td>
<td>38.212</td>
<td>1.1129</td>
<td>0</td>
<td>42.635</td>
</tr>
<tr>
<td>1</td>
<td>20</td>
<td>4.7586</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>34.823</td>
<td>38.212</td>
<td>1.1129</td>
<td>0</td>
<td>42.635</td>
</tr>
<tr>
<td>1</td>
<td>24</td>
<td>2.3655</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>34.823</td>
<td>38.212</td>
<td>1.1129</td>
<td>0</td>
<td>42.635</td>
</tr>
</tbody>
</table>

Table 1.4 Sample data for subject 1 from data set 2.

1.3 Pharmacokinetic models

The application of PK modeling to drug development has grown exponentially (Sheiner and Steimer, 2000; Csajka and Verotta, 2006). Currently, the PK model is an important tool in drug development. In pre-clinical studies, it helps interpret data from animals to prepare for testing in humans; in clinical testing, it efficiently quantifies the inter-variability and intra-variability even with very few measurements per subject (Prez-Urizar et al., 2000).

PK models were first introduced by Teorell (1937), in order to explain drug absorption, distribution, and elimination. He proposed compartments to separate organs and tissues, linked by first order kinetic rate. Since then, the PK model has been accepted as the main way to analyze dose-response relationship. There are several kinds of models: empirical, physiologically based and compartmentally based models (Shargel et al., 2004). This thesis focuses on compartmentally based model, which considers human body as a collection of compartments.

Let’s look at a simple one-compartment model with bolus intravenous injection (Figure 1.1) described in Welling (1997),
Figure 1.1  One compartment model with bolus intravenous injection. $A$: the amount of drug in the body. $A = C \times V$, where $C$ is the concentration of drug in body fluids and $V$ is the drug distribution volume; $k_{el}$: the elimination rate constant, i.e., the first order rate constant for drug elimination from the body.

Utilizing a differential equation, we can get,

$$\frac{dA}{dt} = -k_{el}A \quad (1.1)$$

where $t$ is time. Integrating, we obtain,

$$\ln A - \ln A_0 = -k_{el}t \quad (1.2)$$

and then,

$$e^{\ln(A/A_0)} = e^{-k_{el}t} \quad (1.3)$$

$$A = A_0 e^{-k_{el}t} \quad (1.4)$$

Finally, we can translate the amount of drug in the body ($A$) to the concentration of drug ($C$) by dividing by the volume of distribution ($V$),
Another example I would like to explain here is the two-compartment open model with rapid intravenous injection (Figure 1.2) discussed in Welling (1997) and Bourne (2010).

It is described by the following equations,

\[
\frac{dA_1}{dt} = k_{21} A_2 - k_{12} A_1 - k_{el} A_1
\]  

(1.6)

\[
\frac{dA_2}{dt} = k_{12} A_1 - k_{21} A_2
\]

(1.7)
After solving these equations as described in Bourne (2010), we get the final equation for the central compartment as follows,

\[ C_1 = Ae^{-\alpha t} + Be^{-\beta t} \] (1.8)

where,

\[ A = \frac{D(k_{21} - \alpha)}{V_1(\alpha - \beta)} \] (1.9)

\[ B = \frac{D(k_{21} - \beta)}{V_1(\alpha - \beta)} \] (1.10)

\[ \alpha = \frac{1}{2}[(k_{12} + k_{21} + k_{el}) + \sqrt{(k_{12} + k_{21} + k_{el})^2 - 4k_{21}k_{el}}] \] (1.11)

\[ \beta = \frac{1}{2}[(k_{12} + k_{21} + k_{el}) - \sqrt{(k_{12} + k_{21} + k_{el})^2 - 4k_{21}k_{el}}] \] (1.12)

### 1.4 Population pharmacokinetic (PopPK) models

The PopPK model (Sheiner et al., 1972, 1977) extends the PK model to incorporate individual variability. The PopPK model can quantify the dose-response relationship with only a few measurements per subjects, and incorporate related clinical data from other resources, such as age and weight of the subject. It is superior to the PK model, which can only adequately describe a homogeneous population with many observations. Developing the PopPK model has become the main research endeavor in pharmacokinetics in this decade (Samara and Granneman, 1997; Minto and Schnider, 1998; Sun et al., 1999; Ette et al., 2004). Many new algorithms have been developed: two-stage, nonlinear mixed effect models, and Bayesian hierarchical model (Bauer et al., 2007). In regards to the significance of PopPK modeling in drug development, the Federal Drug Administration (FDA) issued a guideline that analysts use this model in order to standardize the drug development process (Food and Drug Administration, 1999).
The PopPK model is developed based on nonlinear mixed effects model, and the main goals of this model are twofold: 1) model the relationship between concentration and dose dependent on the covariates; 2) estimate mechanic parameters and related variability. The random effects component extends the model to describe not only intervariability (between subjects), but also intravariability (within subjects).

1.4.1 General formulation for PopPK models

Let’s define the general formulation for PopPK models (Pinheiro and Bates, 2009):

Intravariability level,

\[ y_{ij} = f(x_{ij}; \psi_i) + \epsilon_{ij} \quad i = 1, 2, ..., M; j = 1, 2, ..., N_i; \]  

where \( i \) is the subject ID; \( j \) is the number of samples; \( x_{ij} \) is a vector of known quantities, such as dose, time; \( \psi_i \) is a vector of parameters, such as clearance, volume of distribution; \( \epsilon_{ij} \) is the measurement error; \( y_{ij} \) is the observed response.

Intervariability level,

\[ \psi_i = g(\theta, z_i) + \eta_i \quad \eta_i \sim N(0, \Psi) \]  

where \( \psi_i \) is a vector of parameters for \( i \)th subject, such as clearance, volume of distribution; \( \theta \) is the population parameter; \( z_i \) is the vector of individual covariates, such as weight, age, gender; \( \eta_i \) is the vector of random effects for \( i \)th subject; \( \Psi \) is the variance-covariate matrix.

Covariates can be divided into two categories: 1) intrinsic factors: age, weight, height and race; These factors do not change during study. 2) extrinsic factors: dose, smoking status, etc. These factors may change during study.

Now let’s consider a real example and write these models in the context of pharmacokinetics. We use two parameters (CL and V), one covariate (WT) and proportional residual error model.

Intravariability level,

\[ y_{ij} = f(dose; t_{ij}; \beta_n)[1 + \epsilon] \]  

(1.15)
where dose is the initial dose; $t$ is time; $\beta_n$ is a vector of $n$ mechanic parameters, such as clearance and volume of distribution.

Intervariability level,

$$V_i = V(WT/70)e^{n_i}$$  \hspace{1cm} (1.16)

$$CL_i = CL(AGE)e^{n_i}$$  \hspace{1cm} (1.17)

### 1.4.2 PopPK model building

Ette and Ludden (1995) explained this process as five steps: 1) build structural PK model; 2) diagnose the distribution of random effects; 3) screen covariates; 4) build covariate model; 5) evaluate parameter estimates.

Of these five steps, determining the covariates is a challenge. The general approach is to regress covariates against the estimated parameters to detect any patterns between them. Mandema et al. (1992) applied the generalized additive model to covariate model selection based on first extracting empirical Bayes estimates. Semmar et al. (2005) applied hierarchical cluster analysis to combine single covariate. By building a multivariate categorical covariate, they achieved a three-cluster model with better performance than basic model.

### 1.5 Model diagnostics

Model diagnostics is also known as “model evaluation” (Kutner et al., 2004), “model assessment” (Cook, 1998) and “model appropriateness” (Ette et al., 2003; Brendel et al., 2007). Cook and Weisberg (1994) claimed that model diagnostics help to disprove model with the information obtained from data. In this thesis, I focus on two diagnostic approaches: checking PopPK model assumptions and goodness of fit, and examining resampling statistics.
1.5.1 Checking assumptions and goodness of fit for population pharmacokinetic models

In this section, we would like to cover goodness of fit plots, evaluation of parameter uncertainty, and diagnostics for multiple submodels (structural model, residual error model, and covariate model).

Goodness of fit (GOF) is one important statistical tool in assessing models. Especially because some numerical metrics may be misleading (Cook and Weisberg, 1994), the graphic characteristics of GOF discover the true pattern and evaluate models efficiently. GOF plots dwell on scatter plot of predictions, observations and other related variables to check model fitness, numerical model assumptions, and qualification of variability. We explore GOF plots according to Karlsson et al. (1998), Wade et al. (2005) and Brendel et al. (2006), including population predictions (PRED) versus observations or time, weighted residuals (WRES) versus PRED or time, individual predictions (IPRED) versus observations or time.

Uncertainty of parameter is one of the main model assessments. The reliability of parameter not only affects final prediction, but also helps to determine model robustness. Generally, the distribution of parameters is investigated with regards to standard error (SE) and confidence interval (CI). In addition, the linear relationship among parameters is also explored to detect associations.

PopPK models utilize many submodels. These submodels contribute to model complexity and it is essential to diagnose these submodels respectively to detect any hidden patterns. Karlsson et al. (1998) explained how to check assumptions for covariate and statistical submodels. Later, Wilkins extended to a software called Census (Wilkins, 2005a), including the following diagnostics for submodels:

- Structural model diagnostics: PRED vs DV | IDV, IPRED vs DV | IDV, WRES vs IDV, WRES vs PRED, PRED vs DV | Covariates
- Residual model diagnostics: Distribution of WRES, individual distribution of WRES, absolute weighted residuals (|WRES|) vs PRED, |WRES| vs PRED | Covariates, |WRES|
vs IPRED | Covariates, and autocorrelation of WRES

- Covariate model diagnostics: scatterplot matrix of covariates, parameters vs covariates, ETAs vs covariates, WRES vs covariates.

In this thesis, we will follow the same guideline for submodel diagnostics.

1.5.2 Diagnosing resampling statistics for population pharmacokinetic modeling

Resampling techniques have been widely applied to PopPK modeling to assess the uncertainty of parameters, detect influential observations and test hypothesis (Bruno et al., 1996; Ette, 1997; Ishibashi et al., 2003; Rigby-Jones et al., 2005; Takama et al., 2006; Fasanmade et al., 2009; Marier et al., 2010). Generally, hundreds or thousands of resampling are generated for specific research goals. Users will only investigate the numeric summary related to the resampling methods rather than extract detailed information from those resampling data sets. However, the PopPK model employs complex statistical models, and it will be informative to explore these resampling data sets and gain deep understanding of the data and model. At this time, there is no research done in this field.

In this thesis, we demonstrate how to visualize multidimensional resampling data based on the framework of interactive graphics. Diverse visualization techniques, including histogram, parallel coordinate plots, and multidimensional scaling are implemented to explore the data structure. By linking these visualization techniques through interactive graphics, we can explore these multidimensional data from an integrated and systematical perspective. To our knowledge, this is the first attempt in pharmacokinetic field to incorporate interactive graphics for data analysis.

1.6 Software

PK data focus on dose-response relationship, and because of time constraints, ethnic issues and budget, these kinds of data have the following unique features: unbalanced design, sparse data, and non-optimal design (Dartois et al., 2007). Nonlinear mixed effects models have been
employed to analyze these PK data for a few decades. Currently, there are many software available for PopPK modeling. In this section, I will focus on four main software tools (NONMEM, Monolix, R nlme package, and SAS NLMIX procedure) in this field. The first two are specifically designed for PopPK models, while the other two aims to analyze nonlinear mixed effects models and provide broader perspective not limited to PopPK models. In the following paragraph, I will review these software and provide some background with emphasize on the first two tools.

### 1.6.1 NONMEM

NONMEM was the first program for PopPK model, which was developed by Beal and Sheiner (1980) in University of California, San Francisco. The original goal was to handle sparse data from clinical trials. It was recognized as the most widely used software in PK model and has been evolving until now (current version is NONMEM 7.0).

NONMEM estimation method is based on least squares type criterion, consisting ordinary least square criterion and weighted least squares criterion. There are two estimation methods explored in NONMEM, first-order approximation (FO) and first-order conditional estimation (FOCE). FO method was the first approximation approach taken by NONMEM (NONMEM user guide), and still available in these days. However, because of estimation bias, the first order condition estimation (FOCE) was developed to address this question. FOCE implemented an alternative method to calculate individual random effects based on condition, the approach that incorporates interaction between random effects and statistical errors. In addition, NONMEM provides additional options for estimation, including centering and hybrid to deal with various data sets and diversified models.

NONMEM is developed in Fortran, and as a result, all model diagnostics are explored in a comparatively simple way with regards to the lack of graphical ability. NONMEM diagnostic tools consist of DV versus PRED plot, residual plots, index plots of residuals, plot of WRES versus independent variable, and goodness of fit plots.

Though NONMEM is leading software in the PopPK field, the accuracy of NONMEM
requires further improvement, especially for parameter estimation and variance approximation. This inaccuracy comes inherently from the three approximation approaches implemented in NONMEM: FO, FOCE and Laplacian.

### 1.6.2 Monolix

In 2003, Monolix is developed as a Matlab program (The Monolix group, 2010). It employed an alternative approach to calculate maximum likelihood estimators based on SAEM algorithms. This algorithm applies stochastic approximation to standard EM algorithms and includes three steps: simulation step, stochastic approximation and maximization step (Monolix user guide). During each iteration, the random parameters are simulated from conditional distribution in E step; the parameters are updated until SAME converges to the local or global maximum likelihood. To perform simulation step directly, MCMC is combined with SAME algorithm to take advantage of Hastings-Metropolis algorithm. In addition, since global maximum is a big challenge to reach, Monolix implements simulated annealing to address this question. By incorporating simulated annealing, SAME algorithm does not solely depend on initial guess of parameters, and thus convergence to global maximum is improved.

Based on Matlab, Monolix provides user-friendly graphical interface, powerful and convenient PK/PD model library, goodness of fit plots, and a standard-alone non-matlab program. In addition, C++ is utilized as a basic layer for complex ODE models to speed up calculation.

### 1.6.3 NLMIXED(SAS) and S/R nlme package

SAS NLMIXED procedure is available in SAS version 8.0 and higher, targeting nonlinear mixed effects model. This procedure implements two main approximation methods: first-order Taylor series and adaptive Gaussian quadrature for parameter estimation. Compared with NONMEM, SAS NLMIXED procedure takes advantage of Monte Carlo simulation for integral approximation to maximize likelihood, and thus achieves more accurate results. However, this procedure aims at nonlinear mixed effects model, not particularly for PK models, and it runs more slowly.
S/R nlme package was originally developed by Jose C. Pinheiro and Douglas M. Bates (Pinheiro et al., 2009). It provides FOCE and Laplacian approximation methods. However, like NLMIXED(SAS), it aims at nonlinear mixed effects model, not particularly for PK models.

1.6.4 Other software

NONMEM and Monolix only provide limited functions for model validation. In recent years, more software packages appeared to enrich NONMEM with different features. For example, PsN, a perl module, tries to grant NONMEM more external environment (Lindbom et al., 2004). It not only offers convenient tools for management, but also implements case deletion diagnostics, bootstrap, and stochastic simulation for model validation. It can be connected with NONMEM to extract parameter estimation from output files, subset data sets, manage NONMEM runs, and perform additional model building procedure.

In 1999, Xpose was implemented to improve graphic abilities of NONMEM and auxiliary analyses (Jonsson and Karlsson, 1999). It includes the following features: simplify document production, create data set checkout plot and goodness of fit plot, and compare different model graphically. In addition, generalized additive modeling (GAM), bootstrap and tree-based modeling are incorporated into package to assist covariate model building. Census was later developed to function as the graphical user interface for Xpose as a management and graphical platform.

PsN, Xpose and Census were developed with specific functions and separate components, and these three software “form a tightly-knit web of open-source, freely-available applications for NONMEM modelling and simulation” (http://xpose.sourceforge.net/links.php).

In addition, there are a lot of other tools designed for PopPK analysis. For example, NONMEMMory, a management tool, aims to create user-friendly platform to control summary, comparison and modeling (Wilkins, 2005b). Also, with the exponential usage of R platform, many R packages have been written for NONMEM. PKfit, an R package, handles PK/pharmacodynamic models including following features: noncompartmental analysis, compartmental analysis, nonlinear kinetic process, drug absorption, pharmacodynamic data
modeling, simultaneous fitting and user-defined library (Farenc et al., 2000). Metrum Institute Open-Source tools are developed by Metrum Institute, consisting of NUMQual, MIfun, bugsParallel, bugsPKPDmodelLibrary and SASxport (Metrum Institute, 2010). PKbugs, a program based on BUGs, is available for Bayesian data analysis (Lun, 2010).

1.7 Research problem and scope

Biological problems often involve fitting nonlinear models to data. In pharmacokinetics, analysts study a subject’s response to drug doses, which will typically follow a quick increase in concentration as the drug circulates through the body, and a gradual nonlinear decrease as it is processed and eliminated. These models are diagnosed with the help of experimental data.

Some special modeling software exists for most of these problems, for example NONMEM, Monolix. General modeling software can also be used, such as PROC NLMIX in SAS and the package nlme in S/R. A common problem is that these tools do not provide ways to adequately diagnose the model fit. The FDA is encouraging new approaches to model diagnosis.

This thesis addresses this gap, with the following contributions: 1) developing methods and tools to check the model assumptions and goodness of fit for PopPK models; 2) visualizing resampling statistics for PopPK models; 3) implementing interactive graphics for PopPK model diagnostics; 4) exploring the effects of covariate correlation on PopPK covariate modeling.
CHAPTER 2. CASE STUDIES ILLUSTRATING THE USE OF GRAPHICS IN POPULATION PHARMACOKINETIC MODEL BUILDING

To be submitted to a Pharmacokinetic journal

Xiaoyong Sun\textsuperscript{1,2}, Dianne Cook\textsuperscript{1,2}

\textsuperscript{1}Binformatics and Computational Biology Program, \textsuperscript{2}Department of Statistics

Iowa State University, Ames, Iowa 50011, USA

Abstract

The more complex a statistical model is the more difficulty it can be to diagnose the fit. Population pharmacokinetic (PopPK) models are complex mixed effects models, fitting a non-linear trend and individual characteristics. Graphics plays an important role in PopPK model building by exploring for hidden structure among data before modeling, detecting anomalies during modeling, and validating results after modeling. In this paper two case studies are used to demonstrate these graphics for PopPK model building, focusing on these aspects, exploratory data analysis, goodness of fit, parameter and random effects evaluation, structural model diagnostics, residual model diagnostics and covariate model diagnostics. Interactive graphics are also illustrated. The approaches described in this paper enrich PopPK model building and provide systematic and comprehensive view of pharmacokinetic data.

2.1 Introduction

The application of population pharmacokinetic (PopPK) modeling in the drug development has grown in this decade because it can handle unbalanced design and sparse data (Samara
and Granneman, 1997; Sun et al., 1999; Ette et al., 2004). However, this advantage does not come without a cost. The complexity of model brings challenges in diagnosing the model fit. Graphics play an important role here, helping to explore for hidden structure among data before modeling, detecting extremity during modeling, and validating results after modeling (Cleveland, 1993; Karlsson et al., 1995; Ette and Ludden, 1995; Karlsson et al., 1998; Ette, 1998; Ette et al., 2001b; Petricoul et al., 2001; Brendel et al., 2006; Karlsson and Savic, 2007; Ene I. Ette, 2007).

From the statistical perspective, Ette (1998) gives a comprehensive tutorial for the application of graphics in PopPK modeling. He recommends making distribution plots, scatter plots, residual plots, partial residual plots, pairs plots, conditional plot, contour plots and star plots. From a model perspective Karlsson et al. (1998) proposes graphics to investigate assumptions required for the PopPK model. In this paper, the authors described 22 assumptions for various situations during the model development. By going through each stage of model building process with graphics, Bonate (2005) demonstrates how to facilitate modeling building with graphics, especially with the real PopPK examples.

This paper uses these approaches in the case studies and extends the methods to include interactive graphics. Interactive graphics has a long history in statistics and has proved to be a powerful tool for exploring the data structure and relationship, and diagnosing models (McDonald, 1982; Cook and Weisberg, 1994; Jog and Shneiderman, 1995; Swayne and Buja, 1998; Unwin et al., 2006; Cook et al., 2007). Jonsson et al. (2007) claimed that it would be a potential tool for pharmacokinetic data analysis. In this research, we cover general framework of pharmacokinetic diagnostics, including exploring data and checking model assumptions by analyzing two case studies. Interactive graphics is also explored in one case study.

2.2 Methods

Many authors (Ette, 1998; Karlsson et al., 1998; Bonate, 2005) had done extensively research in model assumption testing, and we followed these guidelines to automatically perform the following assumption testing: 1) exploratory data analysis; 2) goodness of fit plots; 3)
parameter and random effects evaluation; 4) structural model diagnostics; 5) residual model diagnostics; 6) covariate model diagnostics. We used the R packages *lattice* (Sarkar, 2008), *ggplot2* (Wickham, 2009) for static graphics, and *rggobi* (Temple Lang et al., 2009) for interactive graphics.

### 2.2.1 Exploratory data analysis

Dose history, covariate information, and diverse clinical trials for same purpose should be checked for correctness and accuracy before analyzing models, and data structure should be investigated to screen hidden patterns, outliers and extreme observations linked to individuals for further analysis. Currently, histogram and scatter plot combined with conditional plot were implemented to help achieve these goals. Karlsson emphasized the plots for each patient ID versus each variable in the data file (Karlsson et al., 1998), and Ette pointed out exploratory examination of concentration, distribution and correlations between covariates (Ette, 1998). All of these guidelines were implemented in the case studies.

In addition, we examined a unique technology called interactive graphics. This technique is a powerful tool for data visualization and has very unique feature to link diverse datasets. Cook et al. (2007) showed interesting examples for data mining. In this work, we used R package: *rggobi* to explore the data pattern and structures.

### 2.2.2 Goodness of fit plots

Goodness of fit plots play a key role in checking model fitting. These kinds of plots give an overall perspective of model performance, including scatter plot for concentration versus PRED, concentration versus IPRED, PRED versus time and IPRED versus time (European Medicines Agency, 2007). Most reports submitted to FDA are required to explain response from each patient, and individual plots for concentration/PRED/IPRED versus time are of great importance.
2.2.3 Evaluate parameters and random effects

Generally, there are assumptions for distribution of parameters during modeling process. The histogram was utilized to check this distribution. In addition, the correlation of parameters has significant effect on modeling performance, and it was checked by scatter plots or a scatterplot matrix. The assumptions for random effects were also tested for distribution and correlation by histogram, scatter plots or a scatterplot matrix.

2.2.4 Diagnose structural models

Structural model describes the model without the covariates. In practice, there are three popular structural models for use, including 1-, 2-, and 3-compartment models with different absorption models. After determining structural models, we can further build covariate models by incorporating right covariates. Structural model was diagnosed by PRED versus concentration conditioned on time, IPRED versus concentration conditioned on time, WRES versus time, WRES versus PRED, PRED versus concentration conditioned on covariates, IPRED versus concentration conditioned on covariates.

2.2.5 Diagnose residual error models

Residual model deals with random and unexplained variability ($\epsilon$ in the following function) due to model misspecification, assay errors, dosing history errors, etc.

$$y_{ij} = f(dose; t_{ij}; \beta_t) + \epsilon \quad (2.1)$$

Generally, PopPK model consists of the following common residual models:

- additive error

  $$y_{ij} = f(dose; t_{ij}; \beta_t) + \epsilon \quad (2.2)$$

- proportional error

  $$y_{ij} = f(dose; t_{ij}; \beta_t)(1 + \epsilon) \quad (2.3)$$
• exponential error

\[ y_{ij} = f(dose; t_{ij}; \beta_t)e^\epsilon \]  

(2.4)

• combined additive and proportional error

\[ y_{ij} = f(dose; t_{ij}; \beta_t)(1 + \epsilon_1) + \epsilon_2 \]  

(2.5)

Two assumptions (Karlsson et al., 1998) are related to this submodel: 1) homoscedastic variability; 2) symmetrically distributed residuals. To test these assumptions, we applied the following techniques: 1) histogram for distributions of WRES; 2) histogram for individual distribution of WRES; 3) scatterplot of |WRES| versus PRED to check the shape of residual; 4) scatterplot of |WRES| versus PRED conditioned on covariates to screen the covariate effects; 5) autocorrelation of WRES.

2.2.6 Diagnose covariate models

In general, covariate models study how to incorporate covariates into the model. By linking subject-specific characteristics with model parameters, we can identify right covariates for model. Parameters, ETA and WRES are of great use to help screen proper covariates. We utilized the following methods to check covariate models: 1) scatter plot for parameters versus covariates, ETAs versus covariates, WRES versus covariates; 2) scatterplot matrix of covariates.

2.3 Case study 1

2.3.1 Introduction

The data is from a study of the pharmacokinetics of drug theophylline, which was used to treat asthma. It is data set 1 described in Chapter 1.

The primary question is: “How does theophylline concentration in plasma change with time?” Second or additional questions: “How does covariate, dose, affect drug concentration?”

Since there are only 12 subjects in this research, we will assume normality for within-group errors.
2.3.2 Analysis

2.3.2.1 Data restructuring:

Dose, and weight can be considered as subject-specific variables since it is same for each subject, and concentration is response variable, which changes with measured time. As a result, we split these four variables into two separate groups: subject-specific variable and time-dependent variable.

2.3.2.2 Summary statistics:

To summary subject-specific variables such as dose, and weight, histogram or dot plot is proper to view counts for each subject. Scatter plot facilitates users to identify relationship or pattern between variables. Line plot or time plot can easily show how the concentration changes with time, including minimum, maximum, and trend.

Subject-specific variables: Dose and Weight  Dose for all subjects is between 3 and 6 (Table 2.1 and Figure 2.1) and weight is between 50 and 90 (Table 2.2 and Figure 2.2). Most subjects has dose amount between 4.5 and 5 and weight between 60 and 80. From scatter plot (Figure 2.3), there is a negative linear relationship between dose and weight.

<table>
<thead>
<tr>
<th>Subject</th>
<th>Dose</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4.02</td>
</tr>
<tr>
<td>2</td>
<td>4.40</td>
</tr>
<tr>
<td>3</td>
<td>4.53</td>
</tr>
<tr>
<td>4</td>
<td>4.40</td>
</tr>
<tr>
<td>5</td>
<td>5.86</td>
</tr>
<tr>
<td>6</td>
<td>4.00</td>
</tr>
<tr>
<td>7</td>
<td>4.95</td>
</tr>
<tr>
<td>8</td>
<td>4.53</td>
</tr>
<tr>
<td>9</td>
<td>3.10</td>
</tr>
<tr>
<td>10</td>
<td>5.50</td>
</tr>
<tr>
<td>11</td>
<td>4.92</td>
</tr>
<tr>
<td>12</td>
<td>5.3</td>
</tr>
</tbody>
</table>

Table 2.1  Dose amount for each subject. The minimum amount is 3.10 for subject 9, and the maximum amount is 5.86 for subject 5.
Figure 2.1 Histogram of dose. Doses range between 3 and 6 mg/kg.

**Time-dependent variable: concentration**  Concentration was measured between 0 to 24 hours with most measurements early and one measurements later near 24 hours (Table 2.3, Table 2.4 and Figure 2.4). Figure 2.5 shows two peaks for the concentration: one around 0 and the other between 4 and 8. Figure 2.6 demonstrates a negative linear association between concentration and weight, which is surprising. Figure 2.7 shows the time series of concentration for all subjects.

**2.3.3 Model fitting**

**2.3.3.1 Observed concentration versus time**

Figure 2.7 shows concentration by time separately for each subject, spiking quickly and declining slowly. Figure 2.8 is ordered by maximum concentration, while Figure 2.9 is ordered by time reaching peak, with decreasing order respectively. Subject 6 has lowest peak value, and subject 5 has highest peak value.
2.3.3.2 First-order Two Compartment Model

For this data set, first-order two compartment model was utilized to analyze the data set as most literature used this model for pharmacokinetics. The equation is as follows:

\[ \text{conc} = Dose \cdot e^{\varphi_{1i}} \cdot e^{\varphi_{2i}} \cdot e^{-e^{\varphi_{1i}} \cdot \text{Time}} - e^{-e^{\varphi_{2i}} \cdot \text{Time}} + \epsilon_{ij} \quad i = 1, ..., M, j = 1, ..., n_i \]

\[ \varphi_{li} = A_l \beta + B_l b_i \quad b_i \sim N(0, \Psi) \quad l = 1, 2, 3 \]

In this model, there are three nonlinear parameters, \( \varphi_{1i}(\text{lke}) \), \( \varphi_{2i}(\text{lka}) \), and \( \varphi_{3i}(\text{ICl}) \), ranging from 0 to positive infinity:

- \( \varphi_{1i}(\text{lke}) \): a numeric parameter representing the natural logarithm of the elimination rate constant.
- \( \varphi_{2i}(\text{lka}) \): a numeric parameter representing the natural logarithm of the absorption rate constant.
- \( \varphi_{3i}(\text{ICl}) \): a numeric parameter representing the natural logarithm of the clearance.
- \( i \): patient ID.
<table>
<thead>
<tr>
<th>Subject</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>79.6</td>
</tr>
<tr>
<td>2</td>
<td>72.4</td>
</tr>
<tr>
<td>3</td>
<td>70.5</td>
</tr>
<tr>
<td>4</td>
<td>72.7</td>
</tr>
<tr>
<td>5</td>
<td>54.6</td>
</tr>
<tr>
<td>6</td>
<td>80.0</td>
</tr>
<tr>
<td>7</td>
<td>64.6</td>
</tr>
<tr>
<td>8</td>
<td>70.5</td>
</tr>
<tr>
<td>9</td>
<td>86.4</td>
</tr>
<tr>
<td>10</td>
<td>58.2</td>
</tr>
<tr>
<td>11</td>
<td>65.0</td>
</tr>
<tr>
<td>12</td>
<td>60.5</td>
</tr>
</tbody>
</table>

Table 2.2  Weight for each subject. The minimum weight is 54.6, and the maximum weight is 84.6.

- $j$: number of sampling.
- $\beta$: a vector of fixed effects.
- $b_i$: a vector of random effects for ith group.
- $A_i, B_i$: are of appropriate dimensions and depend on the group and possibly on the values of some covariates at the jth observation.
- $\Psi$: variance-covariance matrix.

Figure 2.10 - Figure 2.11 shows several curves generated by this function, with four parameters, Dose (initial dose), lke, lka, lCl, three fixed and the fourth changing, respectively.

2.3.3.3  Fit the model

R package: *nlme* was used to fit the first-order two compartment model. The summary about fitting is shown in Table 2.5 and Table 2.6.

The equation with fitted parameter values (lke: -2.45; lka: 0.47; lCl: -3.23) is as follows,

$$conc = Dose \cdot e^{-2.45 + 0.47 + 3.23} \cdot \frac{e^{-e^{-2.45 \cdot Time}} - e^{-e^{0.47 \cdot Time}}}{e^{0.47} - e^{-2.45}}$$
Why does fixed effects part of the model fit poorly for some subjects?

1. The dose is universally related to weight (Figure 2.3): heavier subjects get lower dose (interesting!). Subjects 1, 9 are two of the heavier patients, and they get lower dose.

2. Weight may also have effect on the fitting. In Figure 2.3, it shows that there is a negative relationship between dose and weight.

3. For subject 1, 7 and 9, it is difficult to explain with variables available why the model fits poorly (Figure 2.15). It may suggest that we should include more covariates into research, such as age, sex, etc.

Difference between fixed effects (Figure 2.15) and mixed effects (Figure 2.16)

1. Fixed effects for subject 6, 7, 8, 11 overestimated the observed value, while those for subject 1, 5, 9 underestimated the observed value.

2. Mixed effects for subject 5 and 9 underestimated the observed values. All mixed effects (fitted values) are less than the observed values.
Figure 2.4  Time histogram. Almost all concentration data were measured between 0 and 15 hours, except that one was measured around 24 hours after single dose.

Figure 2.5  Concentration histogram. There are two peaks, one around 0, and the other between 4 and 7.
Table 2.3  Time summary for each subject.

3. Subject 1 and 9 have biggest residuals for fixed effects (Figure 2.13). For subject 5, it is interesting that the model for subject 5 fits better with fixed effect than with mixed effect.

4. Deviance residual has decreased from 285.34 (fixed effect) to 55.15 (mixed effect). Figure 2.16 claims similar results: fitting with mixed effect is much better than fitting with fixed effect.

2.3.3.4 Residual versus Time

The residual is distributed randomly in Figure 2.14. Some data may be outliers, but since there are only 11 measured data for each subject, it should be proper for this analysis.

2.3.3.5 Fixed effects versus concentration

For subject 6, 7, 8, and 11, the values of fixed effects are more than observed value; for subject 9, and 1, the values of fixed effects are less than observed value (Figure 2.15). Fixed effects for subject 6 and 11 overestimates lag of curve, while those for subject 10 and 1 underestimates tail.
Table 2.4 Concentration summary for each subject.

<table>
<thead>
<tr>
<th>Subject</th>
<th>mean</th>
<th>max</th>
<th>sd</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>6.43</td>
<td>10.5</td>
<td>3.03</td>
</tr>
<tr>
<td>2</td>
<td>4.82</td>
<td>8.33</td>
<td>3.03</td>
</tr>
<tr>
<td>3</td>
<td>5.08</td>
<td>8.2</td>
<td>2.68</td>
</tr>
<tr>
<td>4</td>
<td>4.94</td>
<td>8.6</td>
<td>2.92</td>
</tr>
<tr>
<td>5</td>
<td>5.78</td>
<td>11.4</td>
<td>3.54</td>
</tr>
<tr>
<td>6</td>
<td>3.52</td>
<td>6.44</td>
<td>2.18</td>
</tr>
<tr>
<td>7</td>
<td>3.91</td>
<td>7.09</td>
<td>2.49</td>
</tr>
<tr>
<td>8</td>
<td>4.27</td>
<td>7.56</td>
<td>2.46</td>
</tr>
<tr>
<td>9</td>
<td>4.89</td>
<td>9.03</td>
<td>2.72</td>
</tr>
<tr>
<td>10</td>
<td>5.93</td>
<td>10.21</td>
<td>3.05</td>
</tr>
<tr>
<td>11</td>
<td>4.51</td>
<td>8</td>
<td>2.55</td>
</tr>
<tr>
<td>12</td>
<td>5.41</td>
<td>9.75</td>
<td>3.50</td>
</tr>
</tbody>
</table>

Table 2.5 Summary for nonlinear mixed-effect model fit by maximum likelihood.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Value</th>
<th>Std.Error</th>
<th>DF</th>
<th>t-value</th>
<th>p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>lke</td>
<td>-2.46</td>
<td>0.05</td>
<td>118</td>
<td>-46.77</td>
<td>0.00</td>
</tr>
<tr>
<td>lka</td>
<td>0.47</td>
<td>0.20</td>
<td>118</td>
<td>2.34</td>
<td>0.02</td>
</tr>
<tr>
<td>lCl</td>
<td>-3.23</td>
<td>0.06</td>
<td>118</td>
<td>-53.78</td>
<td>0.00</td>
</tr>
</tbody>
</table>

2.3.4 Conclusion

Here is a summary of the main findings:

1. For all subjects, the concentration peaks quickly, around 1-2 hours, and then decrease slowly.

2. Each subject responds differently: the peak concentration varies by subject, and the time to peak differs.

3. The mixed effects model fits better than the fixed effects model, reducing the deviance more than 5-fold (285.3 to 55.2). This suggests subject to subject variation is important and not well-modeled by the covariate weight. The mixed effects model inadequately fits all the peaks - it’s always too low.
Figure 2.6 Scatterplot for maximum concentration versus weight. There is a negative linear relationship between weight and maximum concentration for each subject.

<table>
<thead>
<tr>
<th>AIC</th>
<th>BIC</th>
<th>logLik</th>
</tr>
</thead>
<tbody>
<tr>
<td>368.05</td>
<td>388.23</td>
<td>-177.02</td>
</tr>
</tbody>
</table>

Table 2.6 AIC, BIC and logLik values.

4. Some surprising results are: Subjects 1 and 9 are seriously underfit by the fixed effects model. This is not explained by the weight or dose, suggesting missing covariates, for example, gender or age, might be needed. This is somewhat improved in the mixed effects model, perfectly for subject 1, but less so for subject 9. Subject 5 is also badly underfit by the mixed effects model.

2.4 Case study 2

2.4.1 General nonlinear mixed effects model formulation

Model formulation:

\[ y_{ij} = f((\varphi_{li}, \theta_{ij}) + \epsilon_{ij} \quad i = 1, ..., M, j = 1, ..., n_i \]
Figure 2.7 Time plot of concentration versus time for each subject. The highest concentration peak is 11.4 for subject 5.
Figure 2.8  Observed values versus time for each subject, ordered by maximum concentration.

\[ \varphi_{li} = A_i \beta + B_i b_l \sim N(0, \Psi) \quad l = 1, ..., Q \]

Parameters in the model:

- \( i \): patient ID.  \( j \): number of sampling.
- \( \varphi_{li} \): a group-specific parameter.
- \( \theta_{ij} \): a covariate vector.
- \( \epsilon_{ij} \): normally distributed measurement error, a within-group error term.
- \( \beta \): a vector of fixed effects.
- \( b_i \): a vector of random effects for \( i \)th group.
- \( A_i, B_i \): are of appropriate dimensions and depend on the group and possibly on the values of some covariates at the \( j \)th observation.
- \( \Psi \): variance-covariance matrix.
Figure 2.9  Observed values versus time for each subject, ordered by time reaching maximum concentration.
2.4.2 One compartment model with zero order absorption and first order elimination

This is an example used to illustrate PK/PD modeling with NONMEM. There are total 100 patients, and in this pilot study, only first ten patients were sampled for demonstration. We are interested in the concentration of the drug in the organism in response to the dose, over time. Here’s the formulation for a one compartment model:

\[ y_{ij} = Dose \cdot \frac{1}{\varphi'_{1i}} e^{-\frac{\varphi'_{2i}}{\varphi'_{1i}} t} + \epsilon_{ij} \quad i = 1, ..., M, j = 1, ..., n_i \]

Parameters in the model:
Figure 2.12  Mixed effects model fitted values (blue line) and observed concentration (black points) versus Time. Subjects 5 and 9 are substantially underfit.

Figure 2.13  Dot plot for maximum fixed effect by each subject.
Figure 2.14 Time versus Residual.

Figure 2.15 Fixed effects (red line) and observed concentration (black points) versus Time.
Figure 2.16  Mixed effects (blue line), fixed effects (red line), and observed values (black points) versus Time.
• $\varphi_{1j}$: V, volume of distribution

• $\varphi_{2j}$: CL, clearance of drug

• KE: CL/V, elimination rate.

Assumption for model fitting:

• Random effects are normally distributed with mean 0, and homogeneous variance $\eta$.

• Measurement error is normally distributed with mean 0, and homogeneous variance $\sigma$.

2.4.3 Data description

The data is described as data set 2 in Chapter 1. Figure 2.18 - Figure 2.19 are plots of the data. The time plot for 10 patients is shown in Figure 2.17.

![Figure 2.17](image)

Figure 2.17 Time plots for concentration versus time, for patient 1 to patient 10.

2.4.4 Data analysis strategy

In this analysis, we follow these steps,
Figure 2.18  Left figure: A few more male(1) patients than female(0) ones; Right figure: Two different dose amounts, 100mg and 250mg have been given to the equal number of patients.

Figure 2.19  Left figure: Most patients weigh between 20 to 80kg with several very heavy patients around 100kg; Right figure: Age ranges from 10 to 80.
• Investigate the effects of the parameter in the model.

• Fit the data with one compartment model using NONMEM (time is the explanatory variable).

• Check diagnostic plots for models, and along with the covariates (weight, age, etc.).

• Incorporate additional covariates into the model.

• Evaluate the model with interactive graphics.

2.4.5 Parameters

In pharmacokinetics, clearance (CL) determines the area under the curve (AUC), V determines the peak concentration (Welling, 1997). Figure 2.20 - Figure 2.21 show the curves resulting from different values of CL, V, and dose. Increasing the dose increases the initial concentration, effectively shifting the curve up. Increasing KE changes the shape of the curve reducing the time for the drug to be removed, and increasing CL has a similar effect, although it is more moderate. Changing V has little effect.

![Changing_dose](image1)

![Changing_KE](image2)

Figure 2.20  Left figure: Time plot for concentration with different dose, same parameter values, CL, V; Right figure: Time plot for concentration with different KE (elimination rate), with same dose.
2.4.6 Model fitting, building and selection

2.4.6.1 Error model

Additive error model was used for this case study.

2.4.6.2 Parameters

After fitted with NONMEM, we got CL and V as 0.356 and 8.59, respectively.

\[
y_{ij} = Dose \times \frac{1}{8.59} e^{-0.356 \times \frac{0.59}{8.59} t} + \epsilon_{ij} \quad i = 1, \ldots, M, j = 1, \ldots, n_i
\]

2.4.7 Diagnostic plots for model: mixed effect and fixed effect

Plots for mixed effects (IPRED), and fixed effects (PRED) versus time on the data are utilized to verify the fitting (Figure 2.22 and Figure 2.23). In Figure 2.23, blue line represents mixed effects, red line represents fixed effects, and black points represent observed values. In addition, patients with ID 1-10 are selected to display the fitting status.

The three patients (ID: 8, 4, 9) also have a short reaction to the drug and not fitted well by the fixed effects model. The mixed effects model improves the fit for these patients.
Figure 2.22  Fixed effects versus time with observed value (black points) and fixed effect/population effect (red line).

Figure 2.23  Fixed effects (red) and mixed effects (blue) model versus concentration for each patient.
Figure 2.24  Fixed effect (blue), and mixed effect (black) versus concentration.

2.4.8 Diagnostic plots for model: residuals (RES) and weighted residuals (WRES)

Residuals are checked with the following plots,

- Concentration versus predicted concentration for fixed and mixed effects model (Figure 2.24).

- Residuals versus predicted values for fixed and mixed effects model (Figure 2.25).

For a good fit, the points in plot of CONC vs IPRED should be on the line of identity, and the points of RES vs TIME plot should be randomly distributed.

For this case study, fixed effect is far from that identity line, and highly biased (Figure 2.24). Residual (RES) is not randomly distributed (Figure 2.25) and weighted residual (WRES) does a better job.

2.4.9 Covariate relationship

For covariates, such as age, weight, and gender, etc., they partly enter the model by affecting the parameters. So to explore the relationship among these variables, we first check whether or not there is correlation.

In this model, clearance, CL, is the first parameter. We find that there is positive association between the random effect of CL and weight (Figure 2.26), but we don’t find any association
between CL and age or gender. As an example, we incorporate weight as a covariate in the
model,

\[ y_{ij} = \text{Dose} \frac{1}{\varphi_{1i}} e^{-\frac{\varphi_{2i}}{\varphi_{1i}} t} + \epsilon_{ij} \quad i = 1, ..., M, j = 1, ..., n_i \]

\[ \varphi_{1i} = \varphi_1 \frac{WT_i}{70} e^{n_1} \]

We rerun NONMEM and got CL and V as 0.905 and 9.76 respectively.

2.4.10 Basic diagnostics with interactive graphics

Figure 2.27 shows the data loaded into an interactive graphics style. The five plots are as
follows,

- Dose histogram (bottom left): high peaks (orange) are mostly from high dose.

- Weight histogram (bottom center): high peaks (orange) are mostly from light weight;
while low peaks (yellow) are mainly from heavy weight.

- Age histogram (Up center): there is no obvious difference in age for peaks.

- Time plot (Up right) for concentration versus time.
Figure 2.26  Left figure: Age versus random effect of clearance; Right figure: Weight versus random effect of clearance. There is some positive association between weight and random effect of clearance.

- Gender histogram (bottom right): some male patients have highest peak in time plot.

Patients with high peak have been highlighted with orange, while those with low peak have been marked with blue. From this figure, patients have high peak generally have low weight. Curves with 100mg dose cluster at bottom. Also, patients with heavy weight may have lower peak than patients with light weight.

Let’s look at model fitted by NONMEM. Most outliers in a scatter plot of mixed effect (IPRE) versus concentration(CONC) comes from patients with 250mg dose.

Using interactive graphics, I am going through each evaluating criteria one by one: 1) maximum time to peak: this value directly affects response of patients; 2) maximum concentration: high response may give side effects to patients; 3) AUC: the effective period for drug. We can see that all highest residual can contribute to patients with dose: 100mg, light weight, which means something.

2.4.11 Summary of model fit findings

Here is a summary of the main findings:
Figure 2.27  Diagnostics plot. Patients with the largest response have low weight.
1. For all subjects, the concentration peaks quickly, around 0-2 hours, and then decreases slowly.

2. Each subject responds differently: the peak concentration varies by subject, and the time to peak differs.

3. The mixed effects model fits better than the fixed effects model (Figure 2.23). This suggests subject to subject variation is important and not well-modeled by the covariate weight.

4. In this model, weight is found to have positive association with clearance (CL).

5. All of the models fail to capture the peak height for these 10 patients. Under estimating, the maximum concentration could be problematic in providing side effects from a drug. To improve model, the mathematical model will need adjusting.

2.5 Conclusions

Because of the complexity of PopPK models, it is essential step to check model assumptions and goodness of fit during model building. Two case studies were utilized to demonstrate these tests for exploration data analysis, goodness of fit, parameter and random effects evaluation, structural model diagnostics, residual model diagnostics and covariate model diagnostics.

In addition, we applied interactive graphics to PopPK model diagnostics. With brushing and linking, it helped to identify influential patients and evaluate model robustness. To our knowledge, this is the first attempt to implement interactive graphics in pharmacokinetic field, and this new approach will enrich the field of model diagnostics and provide systematic view of pharmacokinetic data.
CHAPTER 3. VISUALIZING RESAMPLING STATISTICS FOR POPULATION PHARMACOKINETIC MODELING

Modified from a paper to be submitted to Journal of Biomedical Informatics

Xiaoyong Sun\textsuperscript{1,2}, Dianne Cook\textsuperscript{1,2}

\textsuperscript{1}Binformatics and Computational Biology Program, \textsuperscript{2}Department of Statistics
Iowa State University, Ames, Iowa 50011, USA

Abstract

This work develops methods for diagnosing pharmacokinetic (PK) models by visualizing resampling statistics, such as case deletion and bootstrap. Population pharmacokinetic (PopPK) modeling targets the dose-concentration effect of drugs. Its application has grown in this decade because it can deal with a large number of patients, sparse sampling strategy, and an unbalanced design. Resampling statistics assist in model validation because they provide a way to measure the uncertainty of parameter estimates, and the influence of individuals. In this work, we adapt visual methods from multivariate analysis, parallel coordinate plots and multidimensional scaling, to examine resampling statistics. Multiple models are fit, parameter estimates and fit diagnostics are extracted and the results are visualized. With careful scaling the dependencies between different statistics can be examined and single patient fits better understood. This work is implemented in the R package, PKgraph.

3.1 Introduction

Graphics has been proven to be an important tool to detect patterns, screen outliers, and test hypotheses (Ette and Ludden, 1995; Ette, 1998; Ette et al., 2001a; Petricoul et al., 2001;
In the field of pharmacokinetics, many multidimensional data are generated to address various biological questions. Ette (1998) gives a comprehensive tutorial on applying statistical graphics to pharmacokinetics and pharmacodynamics. By explaining each plot type in detail, he systematically reviewed the application of graphics and pointed out that “the use of graphic techniques in data visualization aids understanding of the data structure that would lead to an informative data analysis”. At the same time, from a model perspective Karlsson et al. (1998) investigated assumption testing comprehensively for population pharmacokinetic model (PopPK) based on graphics. In that paper, the authors described 22 assumptions for various situations during the model development. Bonate (2005) gave a detailed demonstration on how to facilitate modeling building with graphics. With real PopPK examples, he went through each stage of model building process with graphics. Recently, some new graphical approaches have been developed and explored. Bhasi et al. (2006) developed a novel multidimensional visualization technique called VizStruct. Through a simulated data set, the author demonstrated a subtle difference between one and two compartment models.

Resampling techniques have been widely applied to PopPK modeling to assess the uncertainty of parameters, detect influential observations and test hypothesis (Bruno et al., 1996; Ette, 1997; Langdon et al., 2005; Lehr et al., 2010). Generally, hundreds or thousands of resampling statistics are generated for specific research goals and users tend to examine the numeric rather than graphical summaries. The PopPK model fitting employs complex algorithms, and visualizing the resampling statistics can assist in obtaining a deeper understanding of the data and model.

In this paper, we demonstrate how to visualize resampling statistics using multivariate data plots and interactive graphics. Visualization techniques include parallel coordinate plots, and low dimensional projections. Linking between plots using interactive graphics, helps to digest the findings and explore associations between statistics. We believe this is the first attempt in pharmacokinetic field to incorporate interactive graphics for resampling data analysis. This paper is organized as follows. Section 2 explains the statistical graphics and methods. Section
3 reports the results for this research. The discussion and conclusions are combined in Section 4.

3.2 Statistical graphics and methods

3.2.1 Overview of resampling statistics

Resampling statistics is a term used to describe statistical methods that take multiple samples from a data set, and calculate quantities on each. These methods include bootstrap which help measure uncertainty associated with parameter estimates (Shao and Tu, 1995; Efron and Tibshirani, 1994; Davison and Hinkley, 1997), and case deletion methods, which help to detect outliers that overly influence the model fitting.

In the field of pharmacokinetics, case deletion methods delete all of the observations for a subject. The model is fit for the reduced data set, and the subject is considered to be influential if the parameter changes substantially. The process is repeated for each subject. Some methods allow the deletion of groups of subjects, which would guard against the masking of influence among subjects. In some circumstances single observations may be deleted and the model fit to find single influential data points.

Bootstrapping is sampling with replacement (Efron, 1979). By taking many bootstrap samples, and re-fitting the model, the distribution of the parameter estimates can be explored. This method is typically used to add error bands or confidence intervals around the parameter estimates.

When multiple simulated data sets are created and models fit the resampling statistics data, the final form is shown in Figure 3.1.

3.2.2 Statistical graphics

Starting simply, basic plots such as histograms and scatterplots, provide the fundamental way to examine resampling statistics. In addition, we also examine the multivariate matrix using specialized parallel coordinate plots and multidimensional scaling. We call these static graphics. Linking between these representations allows us to better detect influential subjects.
Figure 3.1 Multiple simulated data sets for resampling statistics. In the first simulation data set (sim1), when the subject is absent from the simulation no parameter is estimated, so a missing value (NA) is generated. One of these tables is generated for each parameter (CL, V and Ke). We can consider this to be a multivariate data set, which we will use to examine the influence of each subject and assess the variability in the parameter estimates.

The R packages lattice (Sarkar, 2008), and ggplot2 (Wickham, 2009) were used for producing the static graphics, and rggobi (Temple Lang et al., 2009) for the interactive graphics.

3.2.2.1 Parallel coordinate plot

Parallel coordinate plot was developed by Inselberg (1985) and Wegman (1986). As the name implies, a parallel coordinate system is used instead of an orthogonal axis system. This enables the high-dimensional space to be represented on a two dimensional page. The construction is as follows:
1. A set of parallel axes is created usually from the columns of the data matrix. For our usage each axes represents one simulation data set, matching each case deletion run in this research.

2. To scale the values for comparison, global minimum and maximum were added to each case deletion run. As a result, all values are scaled by difference between global minimum and maximum, which makes all simulation data sets comparable. For example, in Figure 3.5 all case deletion runs have same unit length: 1.0 (global maximum) - 0.0 (global minimum) = 1, which puts all case deletion runs in same scale. These values for each subject are marked in these axes.

3. The values for each subject are connected.

The final results for each parameter are a set of plots that show how the parameter estimate varies across simulations. If the estimate changes a lot we will see this, and it indicates that the subject(s) deleted in that run was influential on the model fitting.

3.2.3 Multidimensional scaling

Multidimensional scaling (MDS) transforms high dimensional data into low dimensional representation of data while preserving the interpoint distances. It operates by minimizing a loss function called “stress”, which measures the difference between the distances in the high dimensional space and those in the low dimensional space. The low dimensional representation is plotted. A good discussion of MDS can be found in Borg and Groenen (2009). There are many possible loss functions for MDS, including classical, metric and non-metric scaling.

Bonate (2005) pointed out that principal component analysis would be a valuable tool to identify influential cases. When the euclidean distance between points is used, which is classical MDS, MDS performs like principal component analysis. More generally the MDS framework allows a lot of flexibility in how the low dimensional representation is constructed. Using a different distance metric and loss function can allow nonlinear mappings from high to low dimensional space.
In this research we used classical MDS, to obtain a linear projection of the resampling statistics data. The simulations were used as the dimensions, to be reduced. Thus, MDS is summarizing the variability across the simulated data sets. Points correspond to the subjects.

### 3.2.4 Interactive graphics

Interactive or direct manipulation, graphics plots are active, so the user can make changes in plots using mouse action. It is most useful if plots are also linked, so that changing the elements in one plot propagates to all other visible plots. Some useful references include McDonald (1982); Jog and Shneiderman (1995); Swayne and Buja (1998); Temple Lang and Hornik (2001); Heer (2005); Unwin et al. (2006); Cook et al. (2007). In this paper, we linked a histogram, scatter plot, parallel coordinate plot and MDS plot through interactive graphics.

### 3.2.5 Data

We illustrate visualizing resampling statistics using a data set from https://www.accp1.org. The data contains 100 patients, and each individual was sampled with 0, 0.25, 0.5, 0.75, 1, 1.5, 2, 2.5, 3, 4, 6, 8, 12, 16, 20, 24 hours post-dose. Several covariates were measured for each subject, including weight (WT).

The resampling statistics were generated and fitted using the \textit{cdd} and \textit{bootstrap} functions in PsN (Lindbom et al., 2004), and the results were analyzed with R package, PKgraph. In each bootstrap 100 patients were resampled and the number of bootstrap replicates was 50.

### 3.2.6 Model

Drug concentration was fit by a one compartment model i.v. bolus model using NONMEM:

\[
C_{ij} = \frac{Dose}{V_i} e^{-\frac{CL_{i}}{V_i}t_{ij}}
\]  

(3.1)

All data were fitted with NONMEM using the first-order conditional method with interaction.
3.3 Results

3.3.1 Resampling design

It is a good idea to double-check the design of the resampling. In the case deletion methods each subject typically is sequentially deleted. For bootstrap we would expect the distribution of subjects picked for the sample to be fairly uniform.

Figure 3.2 shows the resampling design plots. The plot on the left is for case deletion diagnostics. In resampling data, each subject was not deleted sequentially for analysis. The process is described by the PsN documentation as first selecting subjects to form a “perturb” pool. Subjects are basically deleted sequentially, and periodically the pattern is broken by selecting one of the subjects from the “perturb” pool.

The plot on the right in Figure 3.2 is for bootstrap. A dot indicates that the subject was selected for the sample. We would expect these to be fairly uniformly distributed over the square, if the sampling is truly random. We have no reason to doubt that here because there are no obvious patterns.

3.3.2 Distribution of demographic covariates in resampling data

Ideally the resampling design also is independent of the covariates used in the model fitting. To check this we plot the distribution of the covariates for each simulation. Figure 3.3 shows density plots for the covariate, weight (WT), which is measured in kg. The density plot generally reveals that subjects in the study were typically around 50kg, with another small peak of overweight subjects around 95kg.

The left plot shows a density plot of weight (WT) for the case deletion statistics. Each color indicates a different simulation (sample with one subject removed). The distribution of weight is similar for all samples. The right plot shows distribution of weight for each bootstrap sample. We would expect to see more variation here than the case deletion statistics, and we do. Some run only included light weight patients, resulting in a skewed distribution of weight.
Figure 3.2  Plots to examine the resampling designs (left) for case deletion, (right) bootstrap. In each case the run id is plotted horizontally, and the vertical axis displays which cases were in or out of the sample. For the case deletion design we plot the id number of the subject deleted for the run. Lines connect sequential runs. If the deletion was done sequentially by PsN then we would see a straight line from (1,1) to (100, 100). This is not the case: in the first run, the first patient was deleted, in the second run, the 10th patient was deleted, and in the third run, the 100th patient was deleted.

3.3.3 Distribution of parameter in resampling data

Estimates of the model parameters, such as clearance (CL), and volume of distribution (V), determine the model fit. Examining the density of these estimates across samples can be useful also. Figure 3.4 shows the distribution of clearance for each sample. The basic shape is bimodal, which means that there are two concentrations of clearance around 0.275, and 0.5. The left plot shows the case deletion statistics. The distribution of clearance is fairly similar for most samples. There is one subset where the clearance is much higher in the second mode. The right plot shows the distribution of clearance in the bootstrap samples. There is a lot more variability than the case deletion samples, but no individual sample that has substantially different density than the others.
Figure 3.3  Distribution of weight for the samples: (left) case deletion statistics (right) bootstrap. Each sample is represented using a different color. The density is similar across the case deletion samples. There is a lot more variation in the distribution of weight in the bootstrap samples, as is to be expected.

3.3.4 Parallel coordinate plot and multidimensional scaling for case deletion diagnostics

Figure 3.5 shows the parallel coordinate plot for the scaled estimates for clearance for each model fit. Values for each simulation are connected, and different colors are used. Values for each subject are connected. There were two samples that caused the estimated clearance values to change substantially when they were deleted (marked with red arrows). These are subjects 52 and 20. Subject 52 has a very low clearance value generally, so it is one of the extremes on the data, and its effect on other subject’s clearance values is effectively to push them away from the mean, when it is included in the sample. This subject is an outlier, and influential. Subject 20 has medium value of clearance, so it is not an outlier. When it is deleted the estimated clearance for most other subjects drops, which means that the estimates are inflated when it is included in the sample.

Figure 3.6 shows the MDS plot. Patient 52, 20 are outliers in this plot suggesting that they are influential on the clearance estimates.
Figure 3.4 Distribution of clearance (CL) in (left) case deletion statistics, and (right) bootstrap samples. There is more variability in the bootstrap samples, as expected. In the case deletion samples, one run had a noticeably higher second peak than those in other samples.

3.3.5 Interactive graphics

The parallel coordinate plot and MDS plot can be linked to check if the two observations identified as influential are the same, and also linked to a plot of the subject concentration profiles. This is shown in Figure 3.7 in the ggobi software. The two outlying observations in the MDS plot (right) are brushed (red, blue). Linking is obtained using the ID of the sample (resampleID). The points corresponding to these two samples are colored simultaneously in the parallel coordinates plot (left). This was the approach that we used to learn that these are the same two subjects identified earlier as influential. These two subjects are also highlighted in the time series plots of the concentration for all subjects (middle).

After the two subjects: 52 and 20 were deleted, clearance (CL) changed clearly (middle) from Figure 3.7 (left), we cannot tell the difference of these two subjects from others. The linked plots allow us to compare the information learned separately from individual plots, and examine the other information about these subjects.

Figure 3.8 shows how interactive graphics can be useful in examining the bootstrap statis-
Figure 3.5 Parallel coordinate plot for diagnosing case deletion runs for the clearance estimates. Each line in the figure connects values for each subject, across runs. The clearance estimates change substantially for two runs, suggesting there are two patients which have undue influence on the model fit.

tics. At left is a scatter plot of the variance in the clearance estimates plotted against ordered ID. The subject having the largest variance in the clearance estimates is brushed (blue). This plot is linked to the time series plots of concentration by subject (right). This subject (13) had very low concentration.

3.4 Conclusion

This paper describes new visualization methods for resampling statistics including case deletion and bootstrap. Two new multivariate graphics were designed. How to use interactive graphics to explore for associations between findings and parameters was discussed. And simple plots to examine the resampling design and distributions of parameter estimates and covariates
Figure 3.6 MDS plot for diagnosing case deletion runs for the clearance estimates. The IDs in the figure match the case deletion run ID, and it means that the patient with this ID was deleted. The plot indicates that subjects 52 and 20 were influential on clearance estimates.

for the samples were also described. These methods are implemented and available in the R package PKgraph. This research presents an important contribution to the field of PopPK modeling, and enhances the existing methods for diagnosing these models.
Figure 3.7 Interactive graphics for case deletion diagnostics: (left) parallel coordinate plot, (middle) time series plot for all patients, and (right) MDS plot. Two outliers with case deletion ID: 52 and 20 are brushed in the MDS plot, and the corresponding elements of the other two plots are colored accordingly. The two outliers in the MDS plot correspond to the runs in the parallel coordinates plot indicating the same two subjects as being influential. The time series plots would not have suggested that these two subjects are influential.
Figure 3.8  Interactive graphics for bootstrap statistics. In the left side plot variance of clearance estimates is plotted against subject ID. There is one subject with very large variance, which is brushed (blue). The plot is linked to the time series plot for all subjects (right). This patient has very low concentration.
CHAPTER 4. PKGRAPH: AN R PACKAGE FOR GRAPHICALLY DIAGNOSING POPULATION PHARMACOKINETIC MODELS

Modified from a paper to be submitted to Computer Methods and Programs in Biomedicine

Xiaoyong Sun\textsuperscript{1,2}, Kai Wu\textsuperscript{3}, Dianne Cook\textsuperscript{1,2}

\textsuperscript{1}Binformatics and Computational Biology Program, \textsuperscript{2}Department of Statistics, Iowa State University, Ames, Iowa 50011, USA, \textsuperscript{3}Novartis Institutes for BioMedical Research, Inc., 220 Mass. Ave., Cambridge, MA 02139, USA

Abstract

Population pharmacokinetic (PopPK) modeling has become increasingly important in drug development because it handles unbalanced design, sparse data and the study of individual variation. However, the increased complexity of the model makes it more of a challenge to diagnose the fit. Graphics can play an important and unique role in PopPK model diagnostics. The software described in this paper, PKgraph, provides a graphical user interface for PopPK model diagnosis. It also provides an integrated and comprehensive platform for the analysis of pharmacokinetic data including exploratory data analysis, goodness of model fit, model validation and model comparison. Results from a variety of modeling fitting software, including NONMEM, Monolix, SAS and R, can be used. PKgraph is programmed in R, and uses the R packages lattice, ggplot2 for static graphics, and rggobi for interactive graphics.

4.1 Introduction

Population pharmacokinetic (PopPK) models in drug development study the absorption and distribution of a chemical substance in a living organism. Compartmental analysis uses
kinetic models to describe and predict the concentration-time curve, as the drug moves through various compartments in the organism. It is generally preferable to noncompartmental analysis in which only the total drug exposure is estimated. The population pharmacokinetic (PopPK) model was developed (Sheiner et al., 1972, 1977) to adjust for individual characteristics in the drug response. In the field of Statistics the PopPK model is called a mixed effects model, using fixed effects to estimate demographics such as gender, and random effects to quantify the individual variation. It would also be considered a nonlinear model in that the absorption and elimination of the drug is considered to follow specific kinetics. The PopPK model has become the standard method used in drug development in the past decade, and has been further developed by new algorithms and fitting procedures (Samara and Granneman, 1997; Sun et al., 1999; Ette et al., 2004).

NONMEM (Beal and Sheiner, 1980) was a software developed specifically for PopPK modeling. General purpose software such as SAS NLMIXED, R nlme, WinBUGS and Monolix can also be used to fit the models. Although popular, NONMEM has limited functionality for diagnosing the model fit, providing simply the parameter estimates and some basic scatter plots. In response, several software packages have been developed to enrich NONMEM, Xpose (Jonsson and Karlsson, 1999), Census (Wilkins, 2005a), and PsN (Lindbom et al., 2004), etc. Xpose is an R package that provides additional functionality for NONMEM, such as graphics for exploring the data, covariate selection, model comparison and diagnosis. Census is a Windows front-end that helps manage NONMEM and facilitate the use of Xpose. PsN includes several computer intensive model validation techniques, using case-delete-one, bootstrap, and jackknife subsampling.

The PKgraph software (Sun, 2010) reported in this paper provides a supplement to these existing packages. It provides some additional advantages: (1) interactive graphics to link plots of multiple features for data exploration and model diagnosis, (2) an integrated and comprehensive platform for model diagnostics, and (3) access to model-fitting a wider variety of modeling software, Monolix, SAS NLMIX, R nlme in addition to NONMEM. PKgraph is programmed in R, and uses the R packages lattice, ggplot2 for static graphics, and rggobi for
interactive graphics.

The paper is organized as follows. Section 2 gives an overview of the software and explains the specific functionality. Section 3 demonstrates how to use this software using an example NONMEM data set. Future work is discussed in Section 4.

4.2 PKgraph infrastructure

PKgraph is a platform designed for PopPK model diagnostics. The main functions consist of exploratory data analysis, PK model diagnostics, model validation (case deletion diagnostics and bootstrap), and model comparison. The software incorporates a key concept: interactive graphics to link various datasets and diagnostics plots. The framework is programmed using RGtk2 (Lawrence and Temple Lang, 2009) and consists of two types interfaces (Figure 4.1),

(1) main, containing links to all parts of the software, and handles the basic data management, and links to diagnostic modules, and (2) graph, which provides tools specifically for each diagnostic module.

4.3 Graphical user interface

4.3.1 Main interface

The main interface (Figure 4.2) of PKgraph provides the links to all components of the software. There are four areas: (1) tool area (tool bar and menu bar, top), (2) directory area (middle-left), (3) data area (middle-right) and (4) status bar (bottom). The tool area (1) has menu items linking to the basic management modules (project, configuration, data management) and the diagnostic modules (exploratory data analysis, PK models, model validation, model comparison and interactive diagnostics). These are menu items containing numerous functions associated with each of the different types of diagnostics. The directory area (2) shows current directory and all of its files. These files might be data files, or code, depending on the modeling software used. Clicking on any of the data files, will open them and display them in the data area (3). Choosing the file also brings up a panel allowing for different formats to be read, thus handling all possible modeling software formats. The data files might contain
Figure 4.1 Software architecture of PKgraph. Blue indicates the management interface and basic modules, and green the graphical interface accessing the diagnostic modules. Data can be model fit results from NONMEM, Monolix, SAS or R, and users can manage the data through graphical user interface to perform model diagnostics. Figures can be generated and save based on two R graphic packages: lattice and ggplot2.
raw data, and model diagnostics such as parameter estimates, fitted values and residuals and these are displayed in the table view of the data area. The stats bar displays the progress of the different functions, for example here it says “Data is loaded successfully” to indicate that there were no problems with opening the data file.

Figure 4.2 Main interface of PKgraph. Area 1 has menu items linking to the basic management modules, and the diagnostic modules. Area 2 shows current files. Area 3 is the data area, showing model fit from NONMEM. Area 4 displays the progress of loading data.

4.3.2 Graph interface

The graph interface (Figure 4.3) provides specific functionality for diagnostic modules. Selecting an item from one of the menus on the main interface brings up a graph interface.
The style of the interface is the same for all of the diagnostic modules. It contains three areas: 1) parameter setup area, 2) tool bar, 3) plot area.

Figure 3 shows the interface that appears when a user selects the univariate plots item from the exploratory data analysis menu. The parameter area setup (1) allows choice of variable, plot labels, layout for trellis or faceted plots. A choice of lattice or ggplot2 graphics is provided. The tool bar (2) allows the plots to be saved, opening the plot in ggobi for interaction on the plot, synchronize subset selection from ggobi to the data in R, and close ggobi. The plot area (3) displays the figure, and multiple figures if more than one are created.

4.4 Basic module

This functional unit is the main management platform of PKgraph. It consists of the following operations: process input data, configure diagnostics, manage data, and output either data or graphics for further usage.

4.4.1 Data Input/Output

PKgraph takes model fit results from NONMEM, Monolix, SAS and R as input data. The input file types should be a text file, either as txt or csv format. For example, in NONMEM, using $TABLE the user can specify the interested variables and output file name to produce a file for PKgraph.

Generally, PKgraph requires the following variables for post-processing: unique ID for each patient, time, concentration (CONC), dose, model parameters such as clearance (CL) and volume of distribution (V), model diagnostics such as residuals (RES), or weighted residuals (WRES), and covariates such as weight, height and gender. By matching these variables to a default metric system, the software can recognize the value of these variables and perform routine operations, such as automatically generating plots for exploratory data analysis and PopPK model diagnostics. For model validation such as bootstrap and case deletion diagnostics, users can specify the directory and file name to process these hundreds or even thousands of runs, and then visualize these results using PKgraph.
Figure 4.3 Graph interface of PKgraph. It shows the histogram for concentration generated from lattice graphic package. Area 1 is parameter setup area for figures. Area 2 shows tool bar (four buttons) for the following four functions: saving, opening the plot in ggobi for interaction on the plot, synchronizing subset selection from ggobi to the data in R, and closing ggobi. Area 3 is the data area, showing model fit from NONMEM. Area 4 displays the figure, and multiple figures if more than one are created.
4.4.2 Configuration

The purpose of configuration module is threefold: 1) set up the working directory for current projects; 2) set up a format for saving figures; 3) set up additional annotations for figures. Currently PKgraph supports bmp, jpeg, png, tiff, pdf and win.metafile formats for saving figures. Multiple choices will generate multiple figures with all selected formats. Colors can be set, and a choice of line types, such as a loess fit, can be added to any figure. Users need to set up basic choices in the configuration before using the functional modules, and the configuration is saved for further analysis.

4.4.3 Data management

This module currently has two functions: subset and factor. The subset function enables the user to select subgroups for further analysis. The factor function is utilized to setup categorical variables if needed. Often variables have numerical values, such as 0=male and 1=female, which may be interpreted as numerical data when it should be considered to be categorical. Correct interpretation of variables as factors is important for much of the functionality in PKgraph.

4.5 Diagnostic modules

When the data is imported into PKgraph, and the configuration is complete, users can execute the following functions for model diagnostics: exploratory data analysis, goodness of fit, model validation and model comparison.

4.5.1 Exploratory data analysis (EDA)

EDA helps to find patterns in the data, and explore for unexpected features, such as, subjects with a drug concentration pattern that is different from other subjects, individual concentrations that are unusually extreme, whether males and females follow similar elimination trends, or if the patient’s weight changes the elimination pattern. Generally, histograms and scatterplots are used for univariate and bivariate data. PKgraph provides univariate and
bivariate data analysis panels for EDA. To explore multivariate data, PKgraph also offers scatterplot matrices and parallel coordinates plots (Inselberg, 1985).

### 4.5.2 PK models

The PK models module is designed to enable the user to assess the model fit. There are a lot of tools in this module, which include:

- **Configure model results**: PKgraph expects to see this set of variables, ID, TIME, CONC, CL, V, PRED, IPRE, RES, and WRES, matching the naming system of NONMEM. If the data has different names for these variables, the configure panel allows the user to match them to these quantities. Setting up these variables is important for efficient model fit assessment.

- **Individual plots**: Make plots of each patient. Generally users make a scatterplot of concentration versus time for each subject, while this module also provides functionality for any scatterplots of any pair or variables for each subject.

- **Check goodness of fit**
  
  - **Basic goodness of fit**: The selection of plots in this panel helps to assess the model performance. They include scatterplots for concentration versus predicted values, concentration versus predicted values for random effects, predicted values for fixed effects versus random effects, and these values versus time.

  - **Parameters**: Generally, the model fitting procedure requires certain assumptions about the distribution of parameters. These assumptions are checked using histograms. In addition, the correlation between parameters can affect model performance, and this can be checked by scatterplots or a scatterplot matrix.

  - **Random effects**: As we did for parameters, we also need to check distribution and correlation for random effects using histogram, scatter plot or a scatterplot matrix.

- **Check model assumptions**
Structural model: In practice, there are three popular structural models, 1-, 2-, 3-compartment models with different absorption models. The structural model can be diagnosed by plotting predicted values versus concentration conditioned on time, predicted values for random effects versus concentration conditioned on time, weighted residuals versus time, weighted residuals versus predicted value, predicted value versus concentration conditioned on covariates, predicted value for random effects versus concentration conditioned on covariates.

Residual error model: Residual plots are a vital component of model assessment. This panel allows the user to plot histograms and normal probability plots of the residuals, and residuals versus predicted values, or residuals versus covariates. Two assumptions (Karlsson et al., 1998) are related to this submodel: 1) homoscedastic variability; 2) symmetrically distributed residuals. The plots above help to check these two assumptions.

Covariate model: Covariate models study how to incorporate covariates into the full model. By linking subject-specific characteristics with model parameters, we can identify the best covariates for model. The parameters estimates, random effects and weighted residuals are useful for screening covariates. We can utilize the following methods to check covariate models: scatter plots for parameters versus covariates, random effects versus covariates, weighted residuals versus covariates, a scatterplot matrix of covariates.

4.5.3 Model validation

Model validation makes use of several types of resampling methods. Bootstrap, where the data is sampled with replacement, provides confidence intervals for parameter estimates. Case deletion methods leave a single value, or a single subject out, and re-fit the model. These are used to identify influential cases and subjects.

PKgraph offers two approaches utilizing techniques. The first is to examine the results produced by PsN, which provides extensive resampling techniques for PK data. Alternatively
users can do multiple model fits, and compare these results in PKgraph. Users need to specify a directory to read the PsN.

To visualize the results from resampling methods visual methods from multivariate analysis, parallel coordinate plots and multidimensional scaling (Figure 4.4), are used. Generally, parallel coordinate plot was built as follows: 1) a set of parallel axes was created and these axes matched exactly the data variables. Each axes represented one resampling, matching each case deletion run in this research; 2) The values were marked in these axes scaled by respective variables, representing the parameter values for all subjects in each case deletion run. To scale the variables for comparison, global minimum and maximum were added to each case deletion run; 3) The values from each patient across multiple case deletion runs were connected. In this research, the parameter values (CL) for each subject were connected across axes. Multidimensional scaling (MDS) can transform high dimensional data to low dimensional data by loss function called “stress” and visualize the proximities (Borg and Groenen, 2009). There are four kinds of MDS differing in loss function (distance scaling versus classical scaling, metric scaling versus nonmetric scaling. In this function, we used distance scaling to show difference of multiple case deletion runs based on parameter values (CL).

4.5.4 Model comparison

This module allows users to compare two models. Users need to first match the parameter estimates for the two models, which is basically matching the column names or variable names from two sets of results. There are then three choices for comparison: “histogram comparison” (distribution comparison), “scatter plot comparison” and “transform comparison”.

The “histogram comparison” plots a histogram of each model’s results in order for the distributions to be compared (Figure 4.13). The “scatter plot comparison” plots the values from one model against those of the other model. The “transform comparison” transforms data by ratio or log ratio in order to visualize the difference between variables from two models. These model summary data sets can be passed into ggobi for interactive exploration.
Figure 4.4  Graph interface for case deletion diagnostics. The left area is the parameter setup area. The right area is the figure area. In this figure, a multidimensional scaling plot is generated for visualizing case deletion runs. The IDs in the figure match the case deletion run ID, and it means that the patient with this ID was deleted, and model was re-fit with NONMEM.
Figure 4.5  Model comparison. Top figure: “histogram comparison” for comparing distributions of “CL” from two models. The left area is parameter setup area. In the setup, clearance (CL.x, CL.y) from two models are compared. The figure shows that there is not much difference between two models in the distribution of CL. Bottom figure: “scatter plot comparison” for comparing values of “CL” from two models. The figure shows that two models are quite similar for small “CL” values, while there is some difference for high “CL” values.
4.5.5 Interactive graphics

This functional module incorporates a unique feature: interactive graphics into every step of model diagnostics. It includes three steps: select datasets; configure mapping; and diagnostics, and the resulting data set is passed into ggobi. By linking diverse data sets with a key variable, users can seek patterns by brushing, linking and diagnosing patterns conveniently.

4.6 Example

A dataset from NONMEM is used to demonstrate the PKgraph software. It has 100 patients with the covariates, gender (ISM), age (AGE), and weight (WT). A one compartment model with zero order absorption and first order elimination is fitted using NONMEM. The text file containing the fitted model results is imported into PKgraph. In the “open” dialog, the file format for reading uses default parameters. If the data loads successfully it shows up in the data panel at right, and a message, “Data is loaded successfully” appears in the status bar at bottom of the GUI.

To explore data, we choose “Bivariates” from “Exploratory Data Analysis” located at menu bar to check the scatter plots of interested variables. Figure 4.6 shows the concentration vs time plots, where individual profiles are shown. We can see that concentration peaks at 0 consistent with the zero absorption model) and drops off rapidly for most patients. Studying this relationship conditional on a covariate can be achieved by setting the variable in the “cond” text box. Figure 4.7 shows the individual profiles separately for females (0) and males (1). We can see that the concentration peaks are lower for females. The lattice package was used to make these plots but users can choose to have them produced using the ggplot2 package.

Interactive graphics can be used to drill down into the data. Click the second image button on the right panel. This will start ggobi and load the data. Two windows will be visible console window and plot window (Figure 4.8). To open mode plots use the Display menu, and select the type of plot, then select the variables to include. Here we have a profile plot of all of the individuals, and a scatterplot of weight versus age. In order to link the plots appropriately users need to choose the “Brush” option on the ‘Interaction” menu, and select
Figure 4.6 Exploratory data analysis. The left figure is the parameter setup area. x, y are set as TIME, CONC; time variable is set as TIME; ID variable is set as ID. The right figure is figure area, showing concentration versus time (hr).
Figure 4.7 Exploratory data analysis. The left figure is the parameter setup area. $x$, $y$ are set as TIME, CONC; time variable is set as TIME; ID variable is set as ID. The right figure is figure area, showing concentration versus time (hr) conditioned on ISM (0: female, 1: male).
ID as the variable to connect points in one plot with points in the other plot. In the profile plot subjects with the highest peak concentration are highlighted (blue) and the corresponding covariate values for these subjects are highlighted in the other plot. These two subjects tend to be slightly older and are light in weight.

Figure 4.8 The model results are loaded into ggobi for interactive exploration. The plot at left shows the concentration versus time for all subjects, and the plot at right shows two of the covariates, weight versus age. The two plots are linked by subject ID. Points near the peak concentration are brushed (blue) and from the covariate plot we can see that these two subjects are both light and relatively old.

The detailed information for these two patients can be examined by clicking the third button in the toolbar of the PKgraph graph interface (Figure 4.9).

The next step is to check the model assumptions and diagnose the fitted model, using the
Figure 4.9 Exploratory data analysis. After clicking the third button in toolbar (above figure area), the detailed information for two patients (ID: 55 and 58) is selected for investigation.
“PK model” option. Figure 4.10 shows a plot used to check the structural model, weighted residuals versus predicted values. The residuals are relatively evenly spread between -3 and 3, which says that the structural model looks ok.

![Figure 4.10 Structural model diagnostics. The left area is parameter setup area. It has default parameter setup, and users can change them to their specific interests. The right area is figure area. In this figure, it shows weighted residual (WRES) versus predicted values (PRED). The residuals are randomly distributed.](image)

To dig deeper into the diagnostics, we can examine the resampling statistics generated by PsN, using the “model validation” option. In this process, we have 100 NONMEM leave-one-out runs saved in a directory generated by using PsN function, cdd. For these runs we examine the clearance values (CL). The parallel coordinates plot is shown in Figure 4.11. Lines connect the clearance values for each subject from the 100 models fit with one subject excluded. We
can see some patients have more influence on clearance than others, indicated by a big change in the clearance for all subjects when this one is excluded.

Let’s identify these influential cases with interactive graphics. The data is loaded into ggobi (Figure 4.12) and the parallel coordinates plot is linked to an MDS view of the multiple runs. Brushing an outlier in the MDS plot shows that this subject (52) is one of the influential cases in clearance values. We would also find that subject 20 is the other influential case.

To compare model we use the “model comparison” option. Figure 4.13 shows the comparison of an additive error model (2_CS1_IV1ESTFPDF.fit) with proportional error model (3_CS1_IV1ESTFPDF.fit2). The distribution clearance in two models, separately by gender, is examined using a density plot. There is not a lot of difference, but the first model has slightly higher density values in the second clearance mode for both males and females.

### 4.7 Conclusion and future work

PKgraph is an R package for diagnosing population pharmacokinetic model fitting. In the framework supported by R graphics packages, lattice, ggplot2, and rggobi this program can generate high-quality figures for FDA submission, and provide rapid diagnosis of models using interactive graphics. This R package provides a user-friendly graphical user interface so that the learning curve for users is reasonably short. The PKgraph software serves as a supplement to the existing packages: NONMEM, Xpose and PsN for diagnosing models. Currently PKgraph is based on rggobi for interactive graphics, and rggobi is not so flexible to be integrated with the whole system. The next step would be to incorporate a new R package: qtpaint (qtinterfaces project group, 2009) to replace rggobi and provide a more flexible and faster working environment.

### 4.8 System requirements, availability and installation

PKgraph is an R packaged built on the following R packages: RGtk2, gWidgets, gWidgetsRGtk2, lattice, and ggplot2. It requires R (>2.0) (R Development Core Team, 2008) and GTK+, and runs under Windows, Linux and Mac. The detailed installation guide is available
Figure 4.11 Influence analysis for case deletion diagnostics. The left area is parameter setup area. The right area is figure area. In this figure, it is parallel coordinate plot for case deletion diagnostics. The y axis is case deletion run. Since there are 100 patients, there are 100 case deletion runs. Each run means one patient ID is deleted, model is re-fit, and parameter (CL here) for this patient is calculated. X axis is the parameter (CL) value. Each line in the figure matches parameters (CL) for one patient across multiple case deletion runs. When two patients were deleted in two case deletion runs (marked with red arrows), the values of parameter CL of all other patients changed clearly compared with those in other runs. Parameter (CL) value for some patients increases, while other decreases.
Figure 4.12 Influence analysis for case deletion diagnostics: linking results from multidimensional scaling and parallel coordinate plots. The plot at left shows the parallel coordinate plot (x=simulation ID, y=clearance), and the plot at right shows multidimensional scaling plot. The two plots are linked by case deletion ID. One outlier point (case deletion ID: 52) is brushed (blue). When this subject is left out, the estimated clearance values for other subjects are lower.
Figure 4.13  Density plot comparison for comparing distributions of clearance (CL) from two models (blue, pink), separately by gender (ISM: 0=female, 1=male). The distribution of CL for two models is slightly different, for both genders: model 2 has slightly higher density of values at high clearance, with a higher peak in the second mode.
in the vignette with the package, and users are free to download PKgraph under the terms of GNU license at http://cran.r-project.org/web/packages/PKgraph/index.html.
CHAPTER 5. PKREPORT: REPORT GENERATION FOR CHECKING POPULATION PHARMACOKINETIC MODEL ASSUMPTIONS

To be submitted to Computer Methods and Programs in Biomedicine

Xiaoyong Sun\textsuperscript{1,2}, Dianne Cook\textsuperscript{1,2}

\textsuperscript{1}Bioinformatics and Computational Biology Program, \textsuperscript{2}Department of Statistics
Iowa State University, Ames, Iowa 50011, USA

Abstract

Graphics play an important and unique role in population pharmacokinetic (PopPK) model building by exploring hidden structure among data before modeling, detecting extremity during modeling, and validating results after modeling. We have seen that some practicing analysts have their own code to generate a collection of their favorite diagnostic plots and statistics. The work described in this paper is about a new R package called PKreport, which will generate a collection of plots and statistics for testing model assumptions, visualizing data and diagnosing models. PKreport provides 1) automate plots for users to visualize data and models, 2) automatically generated R scripts that are used to create the plots; 3) implement an archive-oriented management tool for users to store, retrieve and modify figures, 4) generate high-quality graphs based on the R packages, lattice and ggplot2. The general architecture, running environment and statistical methods can be readily extended.

5.1 Introduction

The application of population pharmacokinetic (PopPK) modeling in the drug development has grown in this decade because it can handle unbalanced design and sparse data (Samara and
Granneman, 1997; Sun et al., 1999; Ette et al., 2004). However, this advantage does not come without a cost. The complexity of statistical model brings diverse variability to the results, and as a result, bias is a big challenge in some situation. Graphics play an important and unique role in PopPK model building by exploring hidden structure among data before modeling, detecting extremity during modeling, and validating results after modeling (Cleveland, 1993; Karlsson et al., 1995; Ette and Ludden, 1995; Karlsson et al., 1998; Ette, 1998; Ette et al., 2001b; Petricoul et al., 2001; Brendel et al., 2006; Karlsson and Savic, 2007; Ene I. Ette, 2007).

From the statistical perspective, Ette (1998) gave a comprehensive tutorial for graphics application in PopPK modeling. By exploring distribution plots, scatter plots, residual plots, partial residual plots, pairs plots, conditional plot, contour plots and start plots, he extensively demonstrated the graphic ability in the field of PopPK. At the same time, from a model perspective Karlsson et al. (1998) investigated assumption testing comprehensively for PopPK model based on graphics. In that paper, the authors described 22 assumptions for various situations during the model development. By going through each stage of model building process with graphics, Bonate (2005) gave a detailed demonstration on how to facilitate modeling building with graphics, especially with the real PopPK examples.

In 1999, as a continuation of the work in 1998, Jonsson and Karlsson (1999) developed a software tool: Xpose to help model building with graphics. Equipped with data set checkout plots, goodness of fit plots and tools for covariate model selection, this software has gained great popularity. Later, Wilkins (2005a) further created a graphical user interface and management tool: Census, to help Xpose diagnose models. In 2003, Monolix was developed based on SAEM algorithms (The Monolix group, 2010). Monolix provides user-friendly graphical interface, powerful and convenient PK/PD model library, goodness of fit plots, and a standard-alone non-matlab program.

However, these tools do not provide automatically generated routine graphics. For example, the Federal Drug Administration (FDA) requires detailed graphical explanation, and every submission requires tedious work to generate a large quantity of graphs. Census and Monolix can only be utilized to diagnose models and generate figures for specific goals and
there is no way to reduce time and cost for this routine work, especially that many data share
similar diagnostics. Secondly, it requires users to have expertise in statistical knowledge and
computational skills to generate all related graphs. This may provide obstacles to users who
do not have graphical expertise.

In this work, I developed an automatic report generator, PKreport, for testing model
assumptions, visualizing data and diagnosing models. PKreport provides 1) automate plots
for users to visualize data and models, 2) automatically generated R scripts that are used to
create the plots, 3) an archive-oriented management tool for users to store, retrieve and modify
figures, 4) high-quality graphs based on the R packages, lattice (Sarkar, 2008) and ggplot2
(Wickham, 2009). The general architecture, running environment and statistical methods can
be readily extended.

The paper is organized as follows. Section 2 explains the methods implemented in the
report. Section 3 focuses on the software implementation. Section 4 demonstrates how to use
this package. The conclusions and future work are discussed in Section 5.

5.2 Methods

Many authors (Ette, 1998; Karlsson et al., 1998) had done extensively research in model
assumption testing, and we followed these guidelines to automatically perform the following
assumption testing: 1) exploratory data analysis; 2) goodness of fit plots; 3) parameter and
random effects evaluation; 4) structural model diagnostics; 5) residual model diagnostics; 6)
covariate model diagnostics.

5.2.1 Exploratory data analysis

Dose history, covariate information, and diverse clinical trials for same purpose should
be checked for correctness and accuracy before analyzing models. Data structure should be
investigated to screen hidden patterns, outliers and extreme observations linked to individuals
for further analysis. Currently, histogram and scatter plot combined with conditional plot
were implemented to help achieve these goals. Karlsson et al. (1998) emphasized the plots for
each patient ID versus each variable in the data file, and Ette (1998) described exploratory examination of concentration, distribution and correlations between covariates. All of these guidelines had been implemented in the PKreport package.

5.2.2 Goodness of fit plot

Goodness of fit plot plays a key role in checking model fitting. These kinds of plots give an overall perspective of model performance, including scatter plots for concentration versus PRED, concentration versus IPRED, PRED versus time and IPRED versus time (European Medicines Agency, 2007). Most reports submitted to FDA are required to explain response from each patient. Individual plots for concentration/PRED/ IPRED versus time can be explored for this purpose.

5.2.3 Evaluate parameters and random effects

Generally, there are assumptions for distribution of parameters during modeling process. The histogram was utilized to check this distribution. In addition, the correlation of parameters has significant effect on modeling performance, and it was checked by scatter plots or a scatterplot matrix. The assumptions for random effects were also tested for distribution and correlation by histogram, scatter plots or a scatterplot matrix.

5.2.4 Diagnose structural models

Structural model describes the model without the covariates. In practice, there are three popular structural models for use, including 1-, 2-, and 3-compartment models with different absorption models. After determining structural models, we can further build covariate models by incorporating right covariates. Structural model was diagnosed by PRED versus concentration conditioned on time, IPRED versus concentration conditioned on time, WRES versus time, WRES versus PRED, PRED versus concentration conditioned on covariates, IPRED versus concentration conditioned on covariates.
5.2.5 Diagnose residual error models

Residual model deals with random and unexplained variability ($\epsilon$ in the following function) due to model misspecification, assay errors, dosing history errors, etc.

$$y_{ij} = f(dose; t_{ij}; \beta_t) + \epsilon$$  \hspace{1cm} (5.1)

Generally, PopPK model consists of the following common residual models (Karlsson et al., 1995; Bonate, 2005):

- additive error model
  $$y_{ij} = f(dose; t_{ij}; \beta_t) + \epsilon$$  \hspace{1cm} (5.2)

- proportional error model
  $$y_{ij} = f(dose; t_{ij}; \beta_t)(1 + \epsilon)$$  \hspace{1cm} (5.3)

- exponential error model
  $$y_{ij} = f(dose; t_{ij}; \beta_t)e^{\epsilon}$$  \hspace{1cm} (5.4)

- combined additive and proportional error model
  $$y_{ij} = f(dose; t_{ij}; \beta_t)(1 + \epsilon_1) + \epsilon_2$$  \hspace{1cm} (5.5)

Two assumptions are related to this submodel: 1) homoscedastic variability; 2) symmetrically distributed residuals. To test these assumptions, we applied the following techniques: 1) histogram for distributions of WRES; 2) histogram for individual distribution of WRES; 3) scatterplot of $|WRES|$ versus PRED to check the shape of residual; 4) scatterplot of $|WRES|$ versus PRED conditioned on covariates to screen the covariate effects; 5) autocorrelation of WRES.

5.2.6 Diagnose covariate models

In general, covariate models study how to incorporate covariates into the model. By linking subject-specific characteristics with model parameters, we can identify right covariates for
model. Parameters, ETA and WRES are of great use to help screen proper covariates. We utilized the following methods to check covariate models: 1) scatter plot for parameters versus covariates, ETAs versus covariates, WRES versus covariates; 2) scatterplot matrix of covariates.

5.3 Software implementation

PKreport is an R package aiming to create an automatic pipeline for model assumption testing. Based on a hidden metric system matching default variables to data variables, this package turns the assumption testing to a fast, convenient and comprehensive routine. With the support of two powerful R graphical packages (lattice and ggplot2), this software can generate high-quality figures for diagnosis, archive all figures with specific folders for report and review, and utilize web browsers as the interface for viewing, archiving and analyzing.

5.3.1 Metric system

The default variables function as the currency for communicating between data sets and the package to generate special-purpose plots (Table 5.1). For example, PRED represents prediction calculated from nonlinear mixed effects model fitting, and RES is equal to the difference between observations and predictions (Boeckmann et al., 1994; Brendel et al., 2006). Users may use preferred software to calculate these related variables. As a result, each data set and fitting results have totally different variable names for further analysis. To facilitate model diagnostics, users need to match the package metric system with the variables from data sets. After matching, the package can process data, configure functions, and generate related diagnostic plots. This system provides ways to function for diverse software such as NONMEM, Monolix, R nlme package, and SAS NLMIX procedure.

5.3.2 Configuration

The whole system is configured by three lists: 1) graph list. This list helps user to choose proper figure format (jpg, pdf, png, etc.) as well as the graphical packages. Currently there are two graphical packages implemented for high-quality figures (lattice and ggplot2). 2) histogram
<table>
<thead>
<tr>
<th>Package variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>Patient ID</td>
</tr>
<tr>
<td>TIME</td>
<td>Time after dose</td>
</tr>
<tr>
<td>CONC</td>
<td>The concentration of drug in the body</td>
</tr>
<tr>
<td>PRED</td>
<td>Prediction generated from model fitting</td>
</tr>
<tr>
<td>RES</td>
<td>Residual</td>
</tr>
<tr>
<td>WRES</td>
<td>Weighted residual</td>
</tr>
<tr>
<td>IPRED</td>
<td>Individual prediction</td>
</tr>
<tr>
<td>IWRES</td>
<td>Individual weighted residual</td>
</tr>
<tr>
<td>COV</td>
<td>Covariates</td>
</tr>
</tbody>
</table>

Table 5.1 Package metric system. The first column (Package variable) contains default variable names. These default variable names match NONMEM naming system. The second column explains the details of these variables.

Currently PKreport offers console user interface to test model assumptions. It has the following functions: 1) Match metrics. By matching default package variables to data variables based on one-to one or one-to-many schema, this function sets up global variables for further analysis. 2). Configure figures. This module determines the figure format, figure size and other related properties of figures. 3). Generate figures. Depending on the research goals, users have access to 7 sub-functions for exploratory data analysis, overall goodness of fit plots, parameter diagnostics, random effect diagnostics, structural model diagnostics, residual model diagnostics, and covariate model diagnostics. Each sub-function will create a folder to store related figures as archives. 4). Display results. PKreport offers web browser as a management tool to explore the archives created in function 3 and R scripts in function 5. The main interface includes the names of file directories. 5). Generate R scripts. To improve efficiency and help users to generate high-quality figures, users have option to modify related R scripts to meet their specific requirements. All generated R scripts match the order of figures generated in
function 4. 6). Modify figures. Users can also update or modify certain figures with figure ID for particular purpose. 7). Clean archives. This module will delete all archives (file directories and figures) and clean the global variables in R environment. The general architecture is shown in Fig 5.1.

Figure 5.1  Software architecture of PKreport. It has seven main functions: 1) match metrics; 2) configure figures; 3) generate figures; 4) display results; 5) generate R scripts; 6) modify figure; 7) clean archives. The first two functions set up working environment, and the other functions help to generate related figures.

5.4 Demonstration

One data set from NONMEM was fitted with one-compartment model and utilized for demonstration of PKreport. The following R scripts enable users to input data, configure package, generate figure, display figure from archives, and finally clean up the archives.
Step 1: Read data and set up figure format.

```r
> library(PKreport)
> data(pdata)
# setup configuration
> general.list <- list(save.format="bmp", width = 480, height = 480, package=2)
> hist.list <- list(type=c("count"), layout=c(1,1), ind.layout=c(5,5))
> scatter.list <- list(span=0.25, type=c("p", "smooth"), layout=c(1,1),
ind.layout=c(5,5))
```

Step 2: Match metric system.

```r
> var.name <- list(ID="ID", DV="CONC", TIME="TIME", PRED="PRED", RES="RES",
WRES="WRES",IPRE="IPRE", IDV=c("CLCR", "WT"), COV=c("WT", "AGE"),
ETA=c("ETA1", "ETA2"), PARA=c("CL", "V"))
> PKdata(data=pdata, match.term=var.name)
> PKconfig(general.list, hist.list, scatter.list)
```

Step 3: Generate figures. Because of the page limit, I only demonstrate exploratory data analysis (PKreport.1), goodness of fit plots (PKreport.2) and covariate model diagnostics (PKreport.6).

```r
> PKreport.1(pdata)
> PKreport.2(pdata)
> PKreport.6(pdata)
```

Step 4: Display results in the web browser. Results are shown in Figure 5.2; Archives for figures shown in Figure 5.3; Figures for specific diagnostics shown in Figure 5.4 (left figure from R package: lattice, and right figure from R package: ggplot2); All related R scripts shown in Figure 5.5.

```r
> PKshow()
```

Step 5: Delete archives.

```r
> PKclean()
```
Figure 5.2 Diagnostic results shown in web browser. In this example, the categories in web browser match three diagnostic steps: exploratory data analysis (PKreport.1), goodness of fit plots (PKreport.2) and covariate model diagnostics (PKreport.6). The last one contains all related R scripts.

5.5 Future

The next step for this work is for the package to be released to CRAN and tested by PopPK modelers. We expect that several iterations of refinements to the software for it to become useful and commonly used by PopPK modelers.

5.6 Availability

PKreport is free to download at http://pkreport.sourceforge.net/.
Figure 5.3  Figure archives. In this example, *univar* and *bivar* folders match exploratory data analysis; *gof* folder matches goodness of fit plots; *cov* folder matches covariate model diagnostics. These folders contain all related diagnostic figures. *PKcode.txt* has all related R scripts for these figures.
Figure 5.4 Figures for covariate model diagnostics. Left figure: scatterplot of individual prediction (IPRE) versus concentration (CONC) generated with *lattice* package. Right figure: scatterplot of individual prediction (IPRE) versus concentration (CONC) generated with *ggplot2* package.
Figure 5.5  R scripts generated for goodness of fit plots. The figures generated with these R scripts are stored in `gof` folder. In this example, the R scripts are for two scatterplots (IPRE versus CONC and DOSE versus IPRE) and for each plot, the R scripts for both `lattice` and `ggplot2` packages are generated.
CHAPTER 6. THE EFFECTS OF COVARIATE CORRELATION ON POPULATION PHARMACOKINETIC COVARIATE MODELING

To be submitted to a Pharmacokinetic journal

Xiaoyong Sun\textsuperscript{1,2}, Kai Wu\textsuperscript{3}, Dianne Cook\textsuperscript{1,2}

\textsuperscript{1}Binformatics and Computational Biology Program, \textsuperscript{2}Department of Statistics, Iowa State University, Ames, Iowa 50011, USA, \textsuperscript{3}Novartis Institutes for BioMedical Research, Inc., 220 Mass. Ave., Cambridge, MA 02139, USA

Abstract

Covariate selection is one of most important components in population pharmacokinetic (PopPK) model building. This paper examines the effects of collinearity between covariates on covariate selection and compares the performance of two new algorithms, gradient boosting and random forests, with the existing generalized additive model (GAM) approach. We use empirical Bayes estimates of each subject’s pharmacokinetic parameters covariates based on the structural model to feed into the covariate selection procedures. Data was simulated using one and two significant covariates out of four, a single one compartment intravenous bolus model was fitted using NONMEM. We found that GAM is generally more sensitive to correlation structure than the other two algorithms in the covariate selection.

6.1 Introduction

In any kind of model building the objective is to obtain a parsimonious model, one with as few variables as possible that gives similar accuracy in prediction. Covariate selection is
a major endeavor in population pharmacokinetic (PopPK) model building (Xiao and Fiedler-Kelly, 2002; Miller, 1984; Maritre et al., 1991; Mandema et al., 1992; Jonsson and Karlsson, 1998; Wahlby et al., 2001; Wu and Wu, 2002; Bonate, 2005; Kowalski and Hutmacher, 2001; Wahlby et al., 2002; Bies et al., 2006; Ribbing et al., 2007). Wu and Wu (2002) established that an empirical bayes estimates (EBE) based method works the best. In this approach a model without any covariates is fit to the data, called the base or structural model. Each subject’s pharmacokinetic parameters are then determined using Bayesian estimation. From these estimates the covariates are screened using some regression-based approach. This is the approach used in NONMEM.

In 1992, Mandema et al. (1992) applied a generalized additive model (GAM) as the covariate selection method, using on empirical Bayes estimates for each subject. They showed that this method runs much faster than stepwise approach taken by NONMEM. Wahlby et al. (2002) investigated the selection bias related to stepwise selection procedures and found results favored the GAM approach too. Jonsson and Karlsson (1999) implemented this method in Xpose based on R package, GAM. Xpose has since become very popular.

Other methods examined recently include Wald’s approximation method (WAM) algorithm to approximate a likelihood ratio test (LRT) (Kowalski and Hutmacher, 2001), genetic algorithms (Bies et al., 2006) and lasso methods (Ribbing et al., 2007).

In this paper we report simulations used to address two questions:

1. How does the covariate correlation affect the EBE-based covariate model building?

2. Will two alternative methods, gradient boosting and random forests, perform better than GAM?

The paper is organized as follows. Section 2 explains the simulation setup and methods. Section 3 reports the results from GAM, gradient boosting and random forest. The conclusions and future work are discussed in Section 4.
6.2 Methods

6.2.1 Covariate data generation

The approach used by Mandema et al. (1992) is followed. Data was simulated using the R package *mvtnorm*. Four covariates were used: weight-WT (median=85kg, range = 51-137), height-HT (median=173 cm, range=140-188); age-AGE (median=56 years, range=25-69) and alkaline phosphatase - AP (median=324, 32-615).

6.2.2 Correlation structure

To investigate the effects of collinearity between covariates on the covariate selection, we simulated data using the following four correlation matrices (Table 6.1- 6.4),

<table>
<thead>
<tr>
<th></th>
<th>WT</th>
<th>HT</th>
<th>AGE</th>
<th>AP</th>
</tr>
</thead>
<tbody>
<tr>
<td>WT</td>
<td>1</td>
<td>A</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>HT</td>
<td>A</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>AGE</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>AP</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 6.1 Correlation structure 1: one true covariate (WT). A increased by 0.1 from 0 to 1, resulting in 10 values.

<table>
<thead>
<tr>
<th></th>
<th>WT</th>
<th>HT</th>
<th>AGE</th>
<th>AP</th>
</tr>
</thead>
<tbody>
<tr>
<td>WT</td>
<td>1</td>
<td>A</td>
<td>B</td>
<td>0</td>
</tr>
<tr>
<td>HT</td>
<td>A</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>AGE</td>
<td>B</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>AP</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 6.2 Correlation structure 2: one true covariate (WT). A increased by 0.1 from 0 to 1 and B decreased by 0.1 from 1 to 0 concurrently, resulting in 10 combinations.
Table 6.3 Correlation structure 3: two true covariates (WT and AGE). A increased by 0.1 from 0 to 1, resulting in 10 values.

<table>
<thead>
<tr>
<th></th>
<th>WT</th>
<th>HT</th>
<th>AGE</th>
<th>AP</th>
</tr>
</thead>
<tbody>
<tr>
<td>WT</td>
<td>1</td>
<td>A</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>HT</td>
<td>A</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>AGE</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>AP</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 6.4 Correlation structure 4: two true covariates (WT and AGE). A increased by 0.1 from 0 to 1 and B decreased by 0.1 from 1 to 0 concurrently, resulting in 10 combinations.

<table>
<thead>
<tr>
<th></th>
<th>WT</th>
<th>HT</th>
<th>AGE</th>
<th>AP</th>
</tr>
</thead>
<tbody>
<tr>
<td>WT</td>
<td>1</td>
<td>A</td>
<td>B</td>
<td>0</td>
</tr>
<tr>
<td>HT</td>
<td>A</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>AGE</td>
<td>B</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>AP</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

6.2.3 Pharmacokinetic data simulation

The simulation followed the procedure described by Jonsson and Karlsson (1998). Ten data sets were simulated with each data set containing 64 individuals. Half of the individuals were sampled with 0.5, 2, 4 and 6 hours after dose and the other half were sampled with 0.5, 2, 8, 12 hours post-dose.

Drug concentration was simulated based on one compartment model i.v. bolus model using NONMEM:

\[
C_{ij} = \frac{Dose}{V_i} e^{-\frac{CL_i}{V_i} t_{ij}}
\]  

(6.1)

For one true covariate (WT, sample mean: 85), we used the following model,

\[
CL = \theta_1 + \theta_2 \frac{WT}{85}
\]  

(6.2)

For two true covariates (WT and AGE, sample mean: 85 and 56 respectively), we used the
following model,

\[
CL = \theta_1 + \theta_2 \frac{WT}{85} + \theta_3 \frac{AGE}{56}
\]  

(6.3)

For each \( A \) value or combination of \( A \) and \( B \) value, we repeated simulation for 10 times. Thus, we got 100 data sets for each correlation structure, and 400 data sets for four correlation structures. Figure 6.1 shows time versus concentration for four data sets (for correlation structure 1 and 3, correlation between WT and HT is 0.1; for correlation structure 2 and 4, correlation between WT and HT is 0.1 and correlation between WT and AGE is 0.9). Four scatterplots for concentration versus weight/height/age/alkaline phosphatase are shown in Figure 6.2, Figure 6.3, Figure 6.4, and Figure 6.5 respectively. Two scatterplot matrices for four correlation structures are shown in Figure 6.6 and Figure 6.7 respectively.

All data were fitted with NONMEM using first-order conditional method with interaction.

6.2.4 Covariate selection method

In this research, for the generalized additive model, we utilized R package, \texttt{gam} (Hastie, 2009), and built a GAM model in a step-wise fashion. For gradient boosting, we used the R package, \texttt{gbm} (Ridgeway, 2007). For random forest, we used R package, \texttt{randomForest} (Liaw and Wiener, 2002), to fit the model. In all these methods, the parameter estimates (CL and V) from NONMEM were used as the response variables and covariates were considered as explanatory variables.

6.2.5 Performance measures

For GAM method, covariates selected by model were considered as true covariates. For gradient boosting and random forest, the relative importance of predictor variables was considered as the criterion for screening covariates. The variables with highest score and highest influence on response were considered as true covariates.

In 10 simulated data sets for each correlation structure, we performed covariate model selection with these algorithms for 10 times. We used the times to identify true covariates
Figure 6.1 Time versus concentration for four simulated data sets. For correlation structure 1 and 3, correlation between WT and HT is 0.1; for correlation structure 2 and 4, correlation between WT and HT is 0.1 and correlation between WT and Age is 0.9. There is no clear difference among these four correlation structures.

as the performance measures. For two covariates, if only one covariate was selected in some simulated data set, 0.5 was assigned as the time to identify true covariates.

6.3 Results

6.3.1 Correlation structure of covariates

6.3.1.1 One true covariate (WT)

There were two correlation structures for one true covariate: correlation structure 1 (Table 6.1) and correlation structure 2 (Table 6.2). The correlation structure 1 includes one
correlation between WT and HT. The times to identify the true covariate varied between 3 and 6 in Figure 6.8(left). When the correlation reached 0.9, GAM method only identified the true covariate for two times in 10 data sets. The correlation structure 2 includes two correlations, one between WT and HT, and the other between WT and AGE. For this correlation structure, the times to identify the true covariate jumped to 8, then dropped to 4 and stayed around 6 in Figure 6.8(right) when the correlation between WT and HT increased and the correlation between WT and AGE decreased. In these two correlation structures and only one true covariate, GAM method performed differently.
Figure 6.3  Height versus concentration for four simulated data sets. For correlation structure 1 and 3, correlation between WT and HT is 0.1; for correlation structure 2 and 4, correlation between WT and HT is 0.1 and correlation between WT and Age is 0.9.

6.3.1.2 Two true covariates (WT and AGE)

The accuracy of GAM method decreased when the correlation between WT and HT increased for both correlation structure 3 and 4. For simple correlation structure 3 (correlation between WT and AGE), the maximum times to identify correct covariates were 3.5 at correlation 0.1 and the minimum times were 2 at correlation 0.6 in Figure 6.9(left). For complex correlation structure 4 (correlations between WT and HT, WT and AGE), the maximum times and minimum times were 4.5 at correlation 0.2, and 1.5 at correlation 0.9, respectively in Figure 6.9(right).

GAM performed best in correlation structure 2 (Figure 6.10). In other three correlation structures (1, 3 and 4), GAM performed similarly. The results had highest variability in
corStructure 1

Figure 6.4  Age versus concentration for four simulated data sets. For correlation structure 1 and 3, correlation between WT and HT is 0.1; for correlation structure 2 and 4, correlation between WT and HT is 0.1 and correlation between WT and Age is 0.9.

correlation structure 1.

6.3.2 Comparison of covariate selection methods

Three algorithms performed differently for four correlation structures. For correlation structure 1, there was no obvious difference in three algorithms in Figure 6.11(left). However, gradient boosting generated smallest variability among three algorithms (Figure 6.13). For correlation structure 2, there was no difference in three algorithms in Figure 6.11(right), and GAM had smallest variability (Figure 6.13). For correlation structure 3, gradient boosting and random forest both performed better than GAM in Figure 6.12(left), and all three algorithms generated similar variability to identify true covariates (Figure 6.13). For correlation structure
Figure 6.5 Alkaline phosphatase versus concentration for four simulated data sets. For correlation structure 1 and 3, correlation between WT and HT is 0.1; for correlation structure 2 and 4, correlation between WT and HT is 0.1 and correlation between WT and Age is 0.9.

4, gradient boosting and random forest performed better than GAM in Figure 6.12(right). Gradient boosting had smallest variability (Figure 6.13).

By comparing the combination of correlation structures and methods, Figure 6.13 shows that for correlation structure 1 and 2, all three algorithms performed very similarly with some difference in variability; for correlation structure 3 and 4, both gradient boosting and random forest performed better than GAM.

6.4 Discussion

Contrary to our expectation, high correlation did not decrease accuracy of covariate selection for all three methods. Ribbing and Jonsson (Ribbing and Jonsson, 2004) discussed the
Figure 6.6 Scatterplot matrix for correlation structure 1 and 3. Correlation between WT and HT is 0.1. Conc: concentration. WT: weight. HT: height. AP: alkaline phosphatase.

simple collinearity of covariates with one true covariate in 2004. They found that high correlation hurts prediction power of true covariates. In that research, they only worked with one true covariate, and in the field of pharmacokinetic research, generally two to four covariates were included in the model building. In this paper, we investigated more complex correlation structures with one and two true covariates. Totally four covariates (age, weight, height and alkaline phosphatase) had been selected in the simulation. Our results did not find any relation between high correlation and low accuracy of covariate selection. This may result from small data sets we used in simulation.

Gradient boosting and random forest performs better than GAM when there are two true covariates. GAM method was proposed by Mandema et al. (1992), and implemented in Xpose (Jonsson and Karlsson, 1999). When the covariate correlation exists, GAM method has selec-
tion bias issue. Semmar et al. (2005) applied hierarchical cluster analysis to combine single covariate. By building a multivariate categorical covariate, they achieved a three-cluster model with better performance than basic model. Xpose also includes a decision tree approach to address this question. In this research, we took a new approach based on gradient boosting. This method emphasizes “the committee of weak classifiers” to improve performance of trees, and Hastie et al. (2009) announced this algorithm as the “best off-the-shelf classifier in the world”. The performance of random forest is quite similar to gradient boosting. As found in Figure 6.13, these two alternative algorithms: gradient boosting and random forest work better in covariate selection than GAM when there are two true covariates.

In this research, a few issues are not covered. First, we only used parameters as response
variables for covariate selection. Wahlby et al. (2002) found that random effect (ETA) is more sensitive for covariate selection. As a continuation of this work, it will be more informative to compare the difference of parameter, random effect, and weighted residual as responsible variables on covariate selection. In addition, we need to beware that the sample size is small in this research. We only simulated 10 data sets for each correlation structure and with large size of data, the selection methods may perform differently.

In a word, this paper focuses on the effects of covariate correlation on covariate model building. By investigating a series of correlation structures, we demonstrate that GAM is not so sensitive to covariate selection no matter what covariate correlation exists though its accuracy is not very high. In addition, we compare three related algorithms for covariate selection based on EBE: GAM, gradient boosting and random forest. Gradient boosting and random forest performs better than GAM when there are two true covariates. The comparison of these three algorithms provides further direction for incorporating these two algorithms to improve accuracy of this EBE-based method.

6.5 Conclusions

In this research, we focus on EBE-based method and explore the effects of covariate collinearity on covariate model building and compare performance of three algorithms, including generalized additive model (GAM), gradient boosting and random forest. We found that three algorithms perform similarly with one true covariate. When two true covariates were in the model, gradient boosting and random forest perform better than GAM. In regards of covariate correlation, GAM is more sensitive to correlation structure while the other two algorithms perform consistently. This research reveals the performance difference of three algorithms, and provides direction for improving accuracy of covariate selection.
Figure 6.8  Covariate selection for one true covariate and correlation structure 1 and 2. X is the correlation between weight (WT) and height (HT); Y is the times to identify the true covariate with GAM method in 10 data sets. Left figure: the times to identify the true covariate varied between 3 and 6. When the correlation reached 0.9, GAM method identified the true covariate for only two times in 10 data sets. Right figure: the times to identify the true covariate jumped to 8, then dropped to 4 and stayed around 6 when the correlation between WT and HT increased and the correlation between WT and AGE decreased.
Figure 6.9  Covariate selection for two true covariates and correlation structure 3 and 4. X is the correlation between weight (WT) and height (HT); Y is the times to identify the true covariate with GAM method in 10 data sets. Left figure: the maximum times to identify correct covariates were 3.5 at correlation 0.1 and the minimum times were 2 at correlation 0.6. Right figure: the maximum times and minimum times were 4.5 at correlation 0.2, and 1.5 at correlation 0.9, respectively.
Figure 6.10  Comparison of performance of GAM in four correlation structures. X is correlation structure; Y is the times to identify the true covariate with GAM method in 10 data sets. GAM performed best in correlation structure 2. Correlation structure 1 resulted in highest variability.
Figure 6.11  Performance of three algorithms for covariate selection (correlation structure 1 and 2). There was no obvious difference in three algorithms for both left figure and right figure. *gam*: generalized additive model; *gbm*: gradient boosting; *rf*: random forest.
Figure 6.12  Performance of three algorithms for covariate selection (correlation structure 3 and 4). Gradient boosting and random forest both performed better than GAM. *gam*: generalized additive model; *gbm*: gradient boosting; *rf*: random forest.
Figure 6.13 Performance of combination of correlation structures and methods. For correlation structure 1 and 2, all three algorithms performed very similarly with some difference in variability; for correlation structure 3 and 4, both gradient boosting and random forest performed better than GAM. \textit{gam}: generalized additive model; \textit{gbm}: gradient boosting; \textit{rf}: random forest.
CHAPTER 7. GENERAL CONCLUSION AND FUTURE WORK

7.1 Methodology

The main contributions of methodology in this thesis are the development of PopPK model diagnostics, described as follows:

• Interactive graphics is applied to model building, including the exploratory data analysis, goodness of fit, model validation and model comparison. This is a new addition to the practice of PopPK modeling. It provides a more systematic evaluation of these relatively complicated models.

• New visual methods have been developed to examine resampling statistics for PopPK modeling. Resampling statistics arise when multiple models are fit. The parameter estimates and fit diagnostics are extracted and results are visualized. This work expands the ability to diagnose PopPK models by visualizing resampling statistics. Visual methods from multivariate analysis, parallel coordinate plots and multidimensional scaling, are applied to create the new visualizations. The resampling statistics need to be carefully re-scaled. The visual methods help the analyst understand the effect of individual patients on the model, and dependencies between patients and parameter estimates.

• Preliminary work on exploring the effects of correlation structure on covariate selection in PopPK modeling has been performed. Three algorithms for identifying the best covariates are compared.
7.2 Software

To help users utilize the methods developed in this thesis for PopPK model diagnostics, I developed two R packages, PKgraph and PKreport. PKgraph source code is distributed through http://cran.r-project.org/web/packages/PKgraph/index.html. PKreport is available at http://pkreport.sourceforge.net/.

7.2.1 PKgraph

This package is a graphical user interface for diagnosing PopPK models in R. It reads in results from other specialist modeling software and provides the methods necessary to diagnose the models using the methods described in the previous section. PKgraph can generate high-quality figures for FDA submission, using the R graphics packages, lattice and ggplot2. Furthermore, plots for exploratory data analysis, goodness of fit, model validation and model comparison can be linked using interactive graphics, enabling deeper inspection of dependencies and correlations. The resampling statistics can be visualized. The PKgraph package serves as a supplement to the existing software, NONMEM, Monolix, SAS NLMIXED, R nlme, Xpose and PsN, used for fitting models and generating resampling statistics.

7.2.2 PKreport

This package generates automated diagnostics for model assumption testing. It is designed based on the current working procedures of PopPK analysts, who prefer to have a large collection of plots generated automatically with a model fit. It also uses the R graphics packages, lattice and ggplot2, and can generate high-quality figures for the FDA, archive all figures with specific folders for report and review, and utilize a web browser as the interface for viewing, archiving and analyzing.

7.3 Future work

The main contribution of this thesis has been to develop methods and software for diagnosing PopPK models. There are some obvious next steps for the research.
First, testing these methods and software on real data is vital. The software needs to be used in the pharmaceutical industry to assess its usefulness, and determine the direction of the development.

Second, PKgraph is based on rggobi for interactive graphics, and rggobi is not so flexible to be integrated with the whole system. The next step would be to incorporate a new R package: qtpaint (qtinterfaces project group, 2009) to replace rggobi and provide a more flexible and faster working environment. qtpaint is not in a state that can be used at this stage of the research, but it hopefully will be available in the coming year.

Third, the work on covariate selection with three algorithms, GAM, gradient boosting and random forest, is preliminary. The parameters were treated as response variables for covariate selection. Further work would compare the differences of the parameter, random effects, and weighted residual as response variables being considered in the covariate selection.
APPENDIX A. USER GUIDE FOR PKGRAPH PACKAGE

Introduction

Population pharmacokinetic (PopPK) modeling has become increasingly important in drug development because it allows unbalanced design, sparse data and the study of individual variation. However, this complexity of the model makes it a challenge to diagnose the fit. Graphics can play an important and unique role in PopPK model diagnostics. The software described in this paper, PKgraph, provides a graphical user interface for PopPK model diagnosis with interactive graphics. It also provides an integrated and comprehensive platform for analysis of pharmacokinetic data including exploratory data analysis, goodness of model fit, model validation and model comparison. It can be used with a variety of modeling fitting software, including NONMEM, Monolix, SAS and R. PKGraph is programmed in R, and uses the R packages lattice, ggplot2 for static graphics, and rggobi for interactive graphics. This R package is supported with a user-friendly graphical user interface so that users can easily control diagnosing with simple clicks. The PKGraph software serves as a supplement to the existing packages: NONMEM, Xpose and PsN for diagnosing models.

PKgraph is an R packaged built on the following R packages: RGtk2, gWidgets, gWidgetsRGtk2, lattice, and ggplot2. It requires R (> 2.0) and GTK+, and runs under Windows, Linux and Mac.

Installation

PKgraph needs to install the following programs and R packages:

1. install GTK
For Windows, you can download the GTK Developer’s Pack from http://gladewin32.sourceforge.net/

For Unix, you can fetch the source files for the different libraries from ftp://ftp.gtk.org/pub/gtk/v2.8/

2. Install RGtk2 (Please see RGtk2 Installation notes if you have problems) (Lawrence and Temple Lang, 2009)
   a. Install R package, RGtk2: `install.packages("RGtk2")`

3. Install rggobi (Temple Lang et al., 2009)
   a. Download and install ggobi (www.ggobi.org)
   b. Install rggobi: `install.packages("rggobi")`

4. Install gWidgets (on the iwidgets code of Simon Urbanek et al., 2009)
   a. Install R package, gWidgets: `install.packages("gWidgets")`

5. Install gWidgetsRGtk2 (Lawrence and Verzani, 2010)
   a. Install R package, gWidgetsRGtk2: `install.packages("gWidgetsRGtk2")`

6. Install lattice (Sarkar, 2008)
   a. Install R package, lattice: `install.packages("lattice")`

7. Install ggplot2 (Wickham, 2009)
   a. Install R package, lattice: `install.packages("ggplot2")`
PKgraph infrastructure

The software incorporates a key concept: interactive graphics to link various datasets and diagnostics plots. The framework is programmed using RGtk2 and consists of main formats of interfaces, (1) main, containing links to all parts of the software, and handles the basic data management, and links to diagnostic modules, and (2) graph, which provides tools specifically for each diagnostic module. (2). Basic module: data input/output module, configuration module, and data management module.

Graphical user interfaces

Main interface

The main interface (Figure A.1) of PKgraph provide the links to all components of the software. There are four areas: (1) tool area (tool bar and menu bar, top), (2) directory area (middle-left), (3) data area (middle-right) and (4) status bar (bottom).

- The tool area has menu items linking to the basic management modules (project, configuration, data management) and the diagnostic modules (exploratory data analysis, PK models, model validation, model comparison and interactive diagnostics). These are menu items containing numerous functions associated with each of the different types of diagnostics.

- The directory area shows current directory and all of its files. These files might be data files, or code, depending on the modeling software used.

- Clicking on any of the data files, will open them and display them in the data area (3). Choosing the file also brings up a panel allowing for different formats to be read, thus handling all possible modeling software formats. The data files might contain raw data, and model diagnostics such as parameter estimates, fitted values and residuals and these are displayed in the table view of the data area.
The status bar displays the progress of the different functions, for example here it says “Data is loaded successfully” to indicate that there were no problems with opening the data file.

![Figure A.1 Main interface of PKgraph](image)

**Graph interface**

Selecting an item from a diagnostic module menu brings up a graph interface (Figure A.2). The style of the interface is the same for all diagnostic functionality. It contains three areas: 1) parameter setup area, 2) tool bar, 3) plot area.

- The parameter area setup allows choice of variable, plot labels, layout for trellis or facetted plots. A choice of lattice or ggplot2 graphics is provided.
• The tool bar allows the plots to be saved, opening the plot in ggobi for interaction on the plot, synchronize subset selection from ggobi to the data in R, and close ggobi.

• The plot area displays the figure, and multiple figures if more than one are created.

Figure A.2  Graph interface of PKgraph

Functional module

Functional module matches the menu items in PKgraph toolbar. It includes the following menu items:

• Project

• Configure
• Data management
• Exploratory data analysis
• PK models
• Model validation
• Model comparison
• Interactive graphics

In the next sections, I will go through each menu item in detail.

Functions

In this section, I will go through each function in the menu item of toolbar.

Project

This menu item is in charge of input, output and save data. It has the following functions (Figure A.3),

• Open data: open modeling fit result from NONMEM, Monolix, SAS, R or other software. It has options to setup the data format, start line and separation symbol.

• Save a file: save a file.

• Save a workspace: save a workspace for later usage. It generally saves a group of lists for configuration and related data.

• Restore old workspace: restore the workspace from the data and list you saved from previous step.

• Exit: exit from PKgraph.
Configure

This menu item is utilized to configure PKgraph. It has the following functions (Figure A.4),

- *Set working directory*: change current directory.
- *Set saving format*: set up saving format for figures, including pdf, jpg, tiff, png bmp, win.metafile, and figure width and height. If figure width and height is not setup, a default one will be used.
- *Save a workspace*: save a workspace for later usage. It generally saves a group of lists for configuration and related data.
• *Set figure configuration*: color and loess can be setup here for figures.

![Image of menu items in Configure]

**Figure A.4  Menu items in *Configure***

**Data management**

This menu item is utilized to manage data. It has the following functions (Figure A.5),

• *Subset*: subset current data.

• *Factor*: factor categorical variables. Graphical packages require variables to be factor type in order to display related symbol in figures.
Exploratory data analysis

This menu item is utilized to explore data and screen patterns. It has the following functions (Figure A.6),

- **Univariates**: plot univariate variables.
- **Bivariates**: plot bivariate variables.
- **Parallel coordinate plot**: Parallel coordinate plot for multivariate variables.
Figure A.6  Menu items in *Exploratory data analysis*

**Univariate**

When clicking this menu item, users will generate a “graph interface” (Figure A.2). In this interface, users can specify all parameters in the left area of window. In the right area of window, it has the following four buttons on the top:

- *save*: save figures.
- *ggobi*: open the plot in ggobi for interaction on the plot.
- *synchronize*: synchronize subset selection from ggobi to the data in R.
- *close*: close ggobi.
**Bivariate**

This menu item also generates a “graph interface”. It is similar to the Univariate interface, except that users will have two variables instead of one.

**Parallel coordinate plots**

This menu item provides access to lattice function: *parallel* function from lattice package.

**PK models**

This menu item is utilized to check model assumptions and goodness of fit. The guideline follows Census menu (http://census.sourceforge.net/). It has the following functions (Figure A.7),

- *Configure model result*: This is the key step to match data variables to default metric system. By this step, data from any platform (NONMEM, Monolix, SAS, R) can be interpreted graphically in figures.

- *Individual plots*: Bivariate plot for each individual.

- *Goodness of fit plots*: Goodness of fit plot is one of key tools to check model fitting. These kinds of plots will give an overall perspective of model performance, including scatter plot for concentration versus PRED, concentration versus IPRED, PRED versus time and IPRED versus time.

- *Parameters*: Generally, there are assumptions for distribution of parameters during modeling process. The histogram is utilized to check this distribution. In addition, the correlation of parameters has significant effect on modeling performance, and it can be checked by scatter plots or a scatterplot matrix.

- *Random effects*: The assumptions for random effects also need to be tested for distribution and correlation by histogram, scatter plots or a scatterplot matrix.
- **Structural model**: Structural model can be diagnosed by PRED versus concentration conditioned on time, IPRED versus concentration conditioned on time, WRES versus time, WRES versus PRED, PRED versus concentration conditioned on covariates, IPRED versus concentration conditioned on covariates.

- **Residual error model**: Two assumptions are related to this submodel: 1) homoscedastic variability; 2) symmetrically distributed residuals. To test these assumptions, we applied the following techniques: 1) histogram for distributions of WRES; 2) histogram for individual distribution of WRES; 3) scatterplot of |WRES| versus PRED to check the shape of residual; 4) scatterplot of |WRES| versus PRED conditioned on covariates to screen the covariate effects; 5) autocorrelation of WRES.

- **Covariate model**: Parameters, ETA and WRES are of great use to help screen proper covariates. We can utilize the following methods to check covariate models: 1) scatter plot for parameters versus covariates, ETAs versus covariates, WRES versus covariates; 2) scatterplot matrix of covariates.

Users have to configure data variable first before going to specific model diagnostics. To illustrate the usage of this menu item, I will use *Configure model result* and *Parameters* as examples.

**Configure model result**

The interface for this function is shown in Figure A.8. The fixed variables are from data, and the flexible variables are from default metric system (Table A.1).

**Parameters**

The interface for this function is shown in Figure A.9. After users choose proper figures in the left window, the system will produce all figures automatically. Users can pick specific figures for diagnosing with functions in the toolbar.
Model validation

Resampling methods have been extensively employed in the model validation. Currently, bootstrap targets for confidence interval, case deletion diagnostics identify influential cases, and stochastic simulation is utilized to compare models (PsN). PKgraph mainly focuses on case deletion diagnostics and bootstrap. It provides the following functions (Figure A.10),

- *Influence analysis summary (PsN)*: analyze PsN cdd results.
- *Visualization for influence analysis*: apply parallel coordinate plots and multidimensional scaling to visualize data from case deletion diagnostics (multiple NONMEM runs).
- *Bootstrap summary (PsN)*: analyze PsN boot results.
- *Visualization for bootstrap*: visualize data from bootstrap (multiple NONMEM runs).
Influence analysis summary (PsN)

This function is specifically for PsN \textit{cdd} results (Figure A.11). It takes two result files from PsN: \texttt{raw\_results1.csv} and \texttt{skipped\_individuals1.csv}, and generates a scatter plot for cov.ratio versus cov.score.

Visualization for influence analysis

This function is to visualize data from case deletion diagnostics (multiple NONMEM runs). Let’s use multiple NONMEM run form PsN (Figure A.12), and find file directory for these runs. Then we can select parameters as shown in Figure A.13. These parameters include:

- \textit{Target directory path}: the path for multiple NONMEM runs. It is a required parameter.
### Table A.1 Package metric system

<table>
<thead>
<tr>
<th>Package variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>Patient ID</td>
</tr>
<tr>
<td>TIME</td>
<td>Time after dose</td>
</tr>
<tr>
<td>CONC</td>
<td>The concentration of drug in the body</td>
</tr>
<tr>
<td>PRED</td>
<td>Prediction generated from model fitting</td>
</tr>
<tr>
<td>RES</td>
<td>Residual</td>
</tr>
<tr>
<td>WRES</td>
<td>Weighted residual</td>
</tr>
<tr>
<td>IPRED</td>
<td>Individual prediction</td>
</tr>
<tr>
<td>IWRES</td>
<td>Individual weighted residual</td>
</tr>
<tr>
<td>COV</td>
<td>Covariates</td>
</tr>
</tbody>
</table>

- Simulation folder pattern: the common name style for multiple NONMEM runs. For this example, it is `NM_run`. It is a required parameter.
- Patient ID: the ID for each subject. It is a required parameter.
- Plot variable: the variable you use to detect difference among patients. For this example, we choose `CL`. It is a required parameter.
- xlabel: the name label for each NONMEM run. It is optional.

**Bootstrap summary (PsN)**

This function is specifically for PsN boot results (Figure A.14). It takes two result files from PsN: `raw_results1.csv` and `included_individuals1.csv`, and generates related plots.

**Visualization for bootstrap**

This function is to visualize data from bootstrap (multiple NONMEM runs). Let’s use multiple NONMEM run form PsN (Figure A.15), and find file directory for these runs. Then we can select parameters as shown in Figure A.16. These parameters include:

- Target directory path: the path for multiple NONMEM runs. It is a required parameter.
- Bootstrap folder pattern: the common name style for multiple NONMEM runs. For this example, it is `NM_run`. It is a required parameter.
Figure A.9  Parameters in PK models

- **NONMEM result file name**: the fit result for each NONMEM run. In this example, it is `CS1_IV1ESTFPDF-1.fit`. It is a required parameter.

- **Bootstrap key table path**: the path for bootstrap key file, which is file describing the sampling schema for patient IDs. It is a required parameter.

- **Bootstrap key table name**: The file describes the sampling schema for patient IDs. In this example, it is `included_individuals1.csv`. It is a required parameter.

- **Patient ID**: the ID for each subject. It is a required parameter.

- **Plot variable**: the variable you use to detect difference among patients. For this example, we choose `CL`. It is a required parameter.

- **xlabel**: the name label for each NONMEM run. It is optional.
In this process, there are three main steps: 1) select datasets; 2) configure mapping; 3) comparison (Figure A.17). The first step is to select datasets for comparison. Currently the program only supports comparison of two models. Then users proceed to configure mapping by matching column names or variable names from two data sets. These matching variables are generally the variables from original data sets and they are not related to model fitting. When all parameters are set, the program offers three choices for comparison: “histogram comparison” (distribution comparison), “scatter plot comparison” and “transform comparison”.

Figure A.10  Menu items in Model validation

Model comparison

In this process, there are three main steps: 1) select datasets; 2) configure mapping; 3) comparison (Figure A.17). The first step is to select datasets for comparison. Currently the program only supports comparison of two models. Then users proceed to configure mapping by matching column names or variable names from two data sets. These matching variables are generally the variables from original data sets and they are not related to model fitting. When all parameters are set, the program offers three choices for comparison: “histogram comparison” (distribution comparison), “scatter plot comparison” and “transform comparison”.
Select datasets

This function is to select datasets available in the PKgraph data area. Figure A.18 shows there are three data sets available, including fit result 2: 2, CS1, IV1ESTFPDF.fit (fit with additive error model) and fit result 3: 3, CS1, IV1ESTFPDF.fit2 (proportional error model). In this example, we will compare these two models.

Configure mapping

This step will join two fit results. As a result, users have to match the original data variables between two fit results. For example (Figure A.19),

- **Matching variables**: ID, Time, Concentration, WT, AGE, etc must be matched in this step. These variables do not change with different models.
Figure A.12 Multiple NONMEM runs for case deletion diagnostics

- *Non-matching variables*: *RES*, *PRED*, *WRES*, etc are fit results, and should NOT be matched. These variables change with different models.

After mapping, a new dataset joining two fit results will show in data area of main interface.

**Comparison**

“histogram comparison” enables to compare distributions of matching parameters from two models. “scatter plot comparison” provides a environment to compare matching parameters by scatter plot. “transform comparison” transforms data by ratio or log ratio in order to visualize the difference between variables from two models. All these models can be linked directly to ggobi for interactive diagnostics by clicking second button in the tool bar area on the top right panel.
All variable names for model 1 will have additional “.x” label, and all variable names for model 2 will have additional “.y” label.

Let us look at “histogram comparison” as one example. First, we need to make sure that current data set is “4_ModelComparison” (Figure A.20); second, we click “histogram comparison”. To compare CL, we select ISM (gender) as the conditional variable, and the result is shown in (Figure A.21).

**Interactive graphics**

This functional module incorporates a unique feature: interactive graphics into every step of model diagnostics. It targets to link diverse data sets in one integrative platform. Users can have access to this feature through ggobi button in the graph interface. In addition, users
have flexibility to apply this feature to achieve their specific goals. In the toolbar, there is an option: \textit{interactive graphics}, designed for this purpose. It includes three steps: select datasets; configure mapping; and diagnostics. By linking diverse data sets with a key variable, users can seek patterns by brushing, linking and diagnosing patterns conveniently.

\textbf{Example}

One dataset from NONMEM is utilized to demonstrate PKgraph. This data set has 100 patients with covariates: ISM (gender), AGE, and WT. The data is fitted with one compartment model with zero order absorption and first order elimination.

As a text file, the fitting result from NONMEM is imported into PKgraph for further investigation and analysis. In the “open” dialog, we set up file format for reading with default
parameters, and as a result, the input data shows up on the right panel while a message, “Data is loaded successfully” appears in the status bar at bottom of panel.

To further explore data, first, we choose “Bivariates” from “Exploratory Data Analysis” located at menu bar to check the scatter plots of interested variables (Figure A.22, Figure A.23). The option “cond” from the functional model interface helps user to draw conditional plots to seek patterns for subgroups. Certainly, users can also select “ggplot2” graphic package with different taste of figure. Next, we can take advantage of interactive techniques to look at maximum concentration by clicking second image button on the right panel. This will start ggobi and load related data. GGobi includes two windows: console window and plot window. In order to link figures together, users need to open all interested figures by “Display” option
in the menu bar. The following figure clearly shows that maximum concentration comes from male patients (value: 1). To look at these data in detail, we go back to the figure graphical user interface and click third image button to check selected data set in ggobi. The selected data set pops up and links to patient with ID: 55. We repeat the same procedure for other variables to check patterns.

Next, we utilize “PK model” option to check model assumptions and diagnose model fitting. The program provides default names such as ID, TIME, COV, etc in order to automatically generate diagnosing results. After we match data variables to the default names, we can proceed to automatically generate routine goodness of fit plots for interested models. Figure A.24 is one of the results for structural model diagnostics.

To further look at the influential cases from same data set, we can link them together by
“model validation” option in menu bar. In this process, we have 100 NONMEM runs available at directory: C:\Projects\modelfit_dir1 using PsN function: cdd. Let’s input the path of these NONM runs, and select plot variable as “CL”. After clicking “OK”, we will have the parallel coordinates plot showing the CL variables for all NONMEM runs. From Figure A.25, we can see some patients have more influential effects on CL when records from these patients are deleted.

Let’s identify these influential cases with interactive graphics. Figure A.26 clearly demonstrates that these influential cases come from patient 52 and 20 based on multidimensional scaling and parallel coordinate plots.

In addition, we compare additive error model (2_CS1_IV1ESTFPDF.fit) with proportional error model (3_CS1_IV1ESTFPDF.fit2) by “model comparison” function in the menu bar.
By comparing the distribution of two models, Figure A.27 does not find significant difference between two models for CL. In addition, using gender as a conditional variable, we found first model always gave a higher peak value for both male and female.
Figure A.19 Configure mapping in Model comparison
Figure A.20  Current data set for *Model comparison*
Figure A.21  histogram comparison for Model comparison
Figure A.22  Exploratory data analysis. Peak is identified with brushing. This patient is from light weight and middle age group.
Figure A.23 Exploratory data analysis. The detailed information for this patient is selected for investigation.
Figure A.24  Structural model diagnostics.
Figure A.25 Influence analysis
Figure A.26  Influence analysis: linking results from multidimensional scaling and parallel coordinate plots.
Figure A.27  Histogram comparison for comparing distributions of CL from two models.
BIBLIOGRAPHY


