UNIVERSIDADE DE LISBOA

FACULDADE DE FARMÁCIA

Mechanism-Based Approach to the Economic Evaluation of Pharmaceuticals

Nazira Suleman

Dissertation supervised by Professor Nuno Elvas Silva and co-supervised by Professor Dragana Lakić

Masters of Biopharmaceutical Sciences

2018

Resumo

A farmacoeconomia é uma disciplina que avalia o uso de medicamentos em termos de recursos na maximização da saúde da população. Dado que os recursos para os cuidados de saúde são finitos, a avaliação económica envolve a estimativa do custo de oportunidade, i.e., os benefícios marginais perdidos como resultado do deslocamento de tratamentos ou serviços existentes para financiar novos medicamentos.

A farmacocinética é a ciência que visa o estudo do movimento de fármacos no organismo, o que inclui a absorção, distribuição, metabolismo e eliminação destes e seus metabolitos. Com o advento da química analítica e métodos de quantificação sofisticados, bem como de um aumento do poder de computação, a farmacocinética como ciência tem tido um desenvolvimento exponencial. Uma das áreas da farmacocinética que se tem desenvolvido mais é a farmacocinética populacional: apesar da farmacocinética de um fármaco poder ser estudada individualmente em cada indivíduo, a abordagem populacional é benéfica para o estudo de grupos de pacientes que são difíceis de investigar, como a população de bebés prematuros, pacientes com insuficiência hepática ou renal.

Na farmacocinética populacional, cada indivíduo é avaliado simultaneamente com o modelo de efeitos mistos não-lineares (parametrização). Não linear significa que a variável dependente dessa concentração está relacionada não linearmente à associação de variáveis independentes e parâmetros do modelo. Efeitos fixos refere-se aos parâmetros que não se alteram em indivíduos, enquanto o efeito aleatório se refere àqueles parâmetros que se alteram através dos indivíduos.

O principal objetivo das estimativas de modelação farmacocinética populacional é o de procurar os parâmetros de farmacocinética populacional e fonte de variabilidade. Os objetivos restantes consistem em concentrações observadas da dose administrada pela deteção das covariáveis preditivas na população avaliada. Em farmacocinética populacional, os indivíduos poderão apenas fornecer dados de concentração plasmática escassos.

As cinco principais partes fundamentais para a construção de um modelo farmacocinético populacional incluem: dados, modelo estrutural, modelo estatístico, modelo de covariáveis e software de modelação. Os modelos estruturais definem o perfil de concentração plasmática ao longo do tempo nos indivíduos. Os modelos estatísticos descrevem a variabilidade aleatória na população que não é explicável (como a variabilidade entre as ocasiões), entre a variabilidade do indivíduo ou a variabilidade residual. Os modelos de covariável demonstram a variabilidade estimada pelas características da população, como covariáveis. O software de modelação, como o software de modelação de efeitos mistos não linear, permite a combinação de dados e modelos e aplica o método de estimativa para avaliar parâmetros para os modelos estatísticos, estruturais e de covariáveis que definem os dados.

Na modelação farmacocinética populacional, o software possui um algoritmo de minimização do valor da função objetivo, praticando a estimativa de máxima verossimilhança. No momento da adaptação dos dados populacionais, a concentração estimada para cada indivíduo é influenciada pela variância dos parâmetros populacionais e de cada parâmetro individual, e pela variação em cada valor das concentrações previstas e observadas. A avaliação da probabilidade marginal depende dos parâmetros de efeito aleatório (η) e efeito fixo da população. Não há existência de solução analítica para verossimilhança marginal. Enquanto buscava a máxima verossimilhança, inúmeras abordagens foram aplicadas para a aproximação da verossimilhança marginal. O FOCE e o LAPLACE são as abordagens mais antigas que estimam a verdadeira verossimilhança com uma função adicional simplificada.

O trabalho de dissertação no âmbito do Mestrado em Ciências Biofarmacêuticas teve por objetivo o estabelecimento de ferramentas baseadas em simulação de dados com base em modelos farmacocinéticos populacionais para uma posterior análise farmacoeconómica. Neste trabalho utilizou-se a informação disponível para a combinação fixa de Glecaprevir e Pibrentasvir (Mavyret®), medicamento usado no tratamento do vírus da hepatite C crónica. As simulações foram realizadas utilizando o software R e seu pacote Shiny. O R é uma linguagem para análise de dados de computação estatística e gráfica.

A população simulada no modelo foi agrupada de acordo com as covariáveis similares, sendo simulados 1000 indivíduos por cenário. O relatório de submissão da FDA do Mavyret® foi usado como referência na modelação farmacocinética populacional. Neste relatório encontra-se descrito o modelo farmacocinético populacional desenvolvido, com base nos estudos clínicos realizados para o medicamento. No modelo descrito, foram identificadas diferentes covariáveis. O modelo descrito foi então implementado no software R e o impacto das covariáveis foi estudado com a aplicação Shiny. A população observada foi categorizada em diferentes grupos, tais como doentes tratados com Glecaprevir / Pibrentasvir com compromisso renal e doentes com compromisso renal e cirrose. Foram criados modelos individuais para cada um dos grupos e a comparação entre cada grupo e seus perfis de concentração-tempo foi realizada pelo uso do navegador R e Shiny, onde a atualização nos resultados pode ser vista automaticamente com a alteração em qualquer da covariável ou da variável.

Para os diferentes modelos finais incorporados no software e para a população simulada, foram calculados os parâmetros farmacocinéticos AUC e Cmax para posterior análise estatística descritiva.

Apesar da implementação dos modelos farmacocinéticos populacionais ter sido realizada em R e Shiny, e os dados terem sido simulados para os diferentes cenários populacionais, a aplicação de metodologias farmacoeconómicas não foram realizadas.

Palavras-chave

Farmacoeconomia, Farmacocinética, Farmacocinética Populacional, R Modelação, Shiny aplicação, Estatística Farmacocinética.

Abstract

Pharmacoeconomics is the discipline concerned with optimal allocation of resources to maximize population health from the use of medicines. Given that resources for health care are finite, economic evaluation involves estimation of the opportunity cost, that is, the marginal benefits forgone as a result of displacing existing treatments or services to fund new medicines.

The purpose of this study is to use tools in pharmacoeconomic analysis for the examination of the positive and adverse impact of the fixed dose combination of Glecaprevir and Pibrentasvir (Mavyret$^{®}$), used to treat chronic hepatitis C virus. In order to examine the effects in pharmacoeconomics analysis, a population pharmacokinetic model was developed using R software and its package Shiny, where R is a language for data analysis of statistical computing and graphics.

The population simulated in the model was grouped according to the similar covariates with the number (n) of 1000. FDA submission report for Mavyret$^{®}$ was used as reference regarding population pharmacokinetics modelling, developed based on the clinical studies performed for the drug product. In the described model, different covariates were identified. The described model was implemented in the R software and the impact of covariates wwas studied with Shiny application. The population observed was categorized in different groups such as patients treated with Glecaprevir/Pibrentasvir having renal impairment and patients with renal impairment and Cirrhosis. Individual models were created for each of the groups and the comparison between each group and their concentration-time profiles was observed that was made easier by the use of R and Shiny web browser where the update in results can be seen spontaneously with the change in any of the covariate or the variable.

Different final models were produced and for the simulated population, the pharmacokinetic parameters AUC and Cmax were calculated for descriptive statistical analysis.

Despite the implementation of population pharmacokinetics models has been accomplished in R and Shiny, and data has been simulated for different population scenarios, pharmacoeconomic modelling and application of pharmacoeconomic methodologies was not practised.

Keywords

Pharmacoeconomics, Pharmacokinetics, Population Pharmacokinetics, R Modelling, Shiny Application, Pharmacokinetic Statistics.

Acknowledgement

It is my foremost duty to express my sincere thanks and deep regards to my supervisor **Professor Nuno Elvas Silva** for his able guidance and support in finalising my thesis and I pay my gratitude to him for cooperating at every step of progression. Unfortunately, I was not able to work with my co-supervisor **Professor Dragana Lakić**, with whom I once had a chance to meet, due to the shortage of time and lengthy learning or the R modelling.

I would also like to extend my gratitude and convey my heartfelt thanks to **Helder Duarte** who assisted me with an appropriate guidance and collaboration and provided me all the supervision as a co-supervisor, which was required throughout my project.

Without their help and proper supervision, my thesis might not have completed and I could not be able to undertake the whole project of learning.

In addition, I would convey special thanks to **Professor Cecilia Rodrigues** for the encouragement and support during the learning of my whole master programme.

Contents

I. Introduction

The macroeconomic factors are progressively affecting the budgets of healthcare. The limitations of these budgets can help in the measures of cost suppression in health care area. In health economics and outcome research, pharmacotherapy is also extensively involved. The relationship of pharmacoeconomic and pharmacokinetic has a significant role in the efficiency of pharmaceutical use. An appropriate pharmacokinetics is the basis of producing the cost-effective drugs aimed by the research and development investments. An appropriate monitoring of drug can aid in the adequate use of drugs with the cost-effective results. Thus, biopharmaceutics and pharmacokinetics assist in providing prospects for the proper use of novel or existing drugs and for balancing of adequate market share.

Hence, pharmacokinetics can provide substantial economic benefits that are the normal outcome of their design as constrained drug input and prevention of high plasma concentrations resulting in toxic effect can be controlled by the diagnosis and treatment of adverse effects. The rational monitoring and improvements can eradicate the requirement of expensive re-examination of the drugs.

1. Population Pharmacokinetics

Pharmacokinetics is the science related to drug movement in the body that includes the absorption, distribution, metabolism and elimination of drugs and its metabolites [1]. It has been profited enormously from the developed analytical chemistry and computer science. Despite the pharmacokinetics of a drug can be studied individually in each subject, a population approach is beneficial for studying patient groups that are challenging to investigate, like premature infants, hepatic or renal impairment patients, etc. [2].

Population pharmacokinetics, also referred as population PK or popPK, is the study that is defined when standard dosage is administered in patient population and the sources of variability in plasma drug concentrations is monitored between them. Measuring the variability between their characteristics such as age, weight, sex, race, renal function and drug interactions can support to modify pharmacotherapy [3]. Observing the population allows the exploration of the variability in pharmacokinetics that exists between the patients, for instance a patient taking a drug with renal impairment shows variations in drug concentration that is excreted in the urine [2], [4].

The population pharmacokinetics methodology assisted the achievement of better prescribing by the examination of drug concentration time data that is gained from scheduled therapeutic analysis. Pharmacokinetic parameters derived from population pharmacokinetics study such as clearance could assist in prescribing patients individually [2], [5].

Traditional pharmacokinetic is typically related to healthy volunteers where several samples are taken at specific times while population pharmacokinetic involve patients being treated with different doses and obtaining blood samples at different intervals. The muddled blood sampling and dosing regimens result in sparse data of 3-4 samples from each patient [2], [3].

2. Pharmacokinetic Modelling

Modelling and simulation are significant tools for incorporating data, information, and mechanisms to assist in reaching at sensible conclusions concerning drug development and its use. ***Figure 1*** demonstrates a summary during the drug development procedure where modelling and simulation are frequently engaged. Building proper models can aid to analyse the time duration of exposure and response for multiple dosing routines [6]. An extensive implementation of population modelling methods can offer an outline for quantitating and monitoring variability in drug exposure and response.

The term population pharmacokinetics denotes to mixed-effects modelling that is a combination of random effects (variance model) and fixed effects (structural model). Random effects parameters comprise of inter-subject variability and unexplained variability when the model is fitted to the data. On the other hand, fixed effects are parameters that include clearance and those factors that expressively effect clearance such as age and weight.

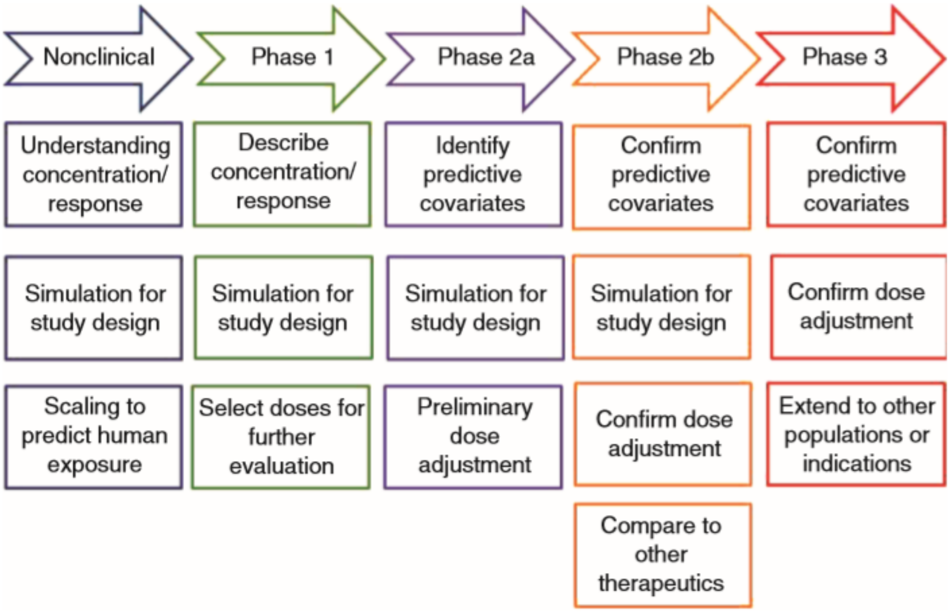

Nonclinical
Phase 1
Phase 2a
Phase 2b
Phase 3
Understanding concentration/ response
Describe concentration/ response
Identify predictive covariates
Confirm predictive covariates
Confirm predictive covariates
Simulation for study design
Simulation for study design
Simulation for study design
Simulation for study design
Confirm dose adjustment
Scaling to predict human exposure
Select doses for further evaluation
Preliminary dose adjustment
Confirm dose adjustment
Extend to other populations or indications
Compare to other therapeutics

Figure 1: Modelling and simulation during drug development [1]

Models are the basic tool for understanding and explaining the time duration of drug exposure and response when multiple formulations or doses of a drug are administered to the subjects. It is also a mean of assessment of linked parameters such as volume of distribution and clearance. Population models could be comprised of a small number of observations from every individual and can be compared to the subsequent parameter observation that helps to define the consistency between populations or observations. It can also give the comparison between other drugs that are in associated therapeutic category to develop the possibility of new therapeutic drug. Thus, it can be concluded that the main aim of assessing population modelling is to create a mathematical method which defines drug's pharmacologic time course in the variety of doses assessed in clinical trials.

Between-subject variability (BSV) in exposure and response is revealed in all drugs and the aim of their development is to identify and quantify this variability. For the improved safety and efficacy or appropriate controlling of variability in drug exposure it is important to understand the effect of factors that include weight, genotype, age, renal and hepatic function on exposure and response of drug [1].

Population modelling is a mean of identification and explanation of the association between the observed drug exposure and response and physiologic characteristics of individuals. Population pharmacokinetic modelling was first introduced in 1972 by Sheiner et al [5]. At the beginning, this method was introduced to work with the sparse pharmacokinetic data that was collected during the analysis of therapeutic drug but shortly it was widened to embrace models relating drug concentration to response such as pharmacodynamics [7], [8]. Subsequently, modelling is now significant measure in the development of drug.

Population parameters were initially predicted by two approaches that include naive pooled approach in which the data of all subjects is fitted collectively by ignoring their differences, and two stage approach in which the data of each subject is fitted independently, and parameter estimated of each individual is combined to calculate mean population parameters. Both of these approaches carry intrinsic difficulties that get worse when deficiencies and errors are present such as missing samples or dosing compliance which eventually cause biased parameter estimates [9]. The Sheiner et al. method solved the problems related to the previous approaches and allowed the combination of sparse data of numerous individuals to evaluate between subject variability (BSV), population mean parameters and the covariate influence that identify and quantify variability in drug response and exposure. This methodology also generated SE which permitted a degree of parameter accuracy.

The significance of each subject in population models is emphasized by estimation of variability, by recognizing the fluctuations in drug exposure with the variation of each covariate of the individual such as age or weight or consequent estimation of subject's characteristics. The practice of pharmacometrics can expand the observation of the linear and saturable metabolism mechanism, notify to test the primary variety of doses, improve the dosage selection for subpopulations of subjects, and assess the study design precision [3].

2.1. Kinds of pharmacokinetic models

Pharmacokinetic models define the concentration and time association. Compartment is the primary concept of all PK models that is defined by the body area in which the drug is kinetically homogenous and fully blends. Compartments are recognised as the universal and essential component of PK models but the models are described by the difference of how the compartments are linked. In different tissues, the equilibrium between the drug concentrations does not appear instantly. Therefore, the hypothesis of one-compartment model often becomes void. After the administration of few drugs, mammillary model is sometimes essential to define the plasma concentration data mathematically [10]. Mammillary models usually comprise of central compartment that demonstrate plasma with some peripheral compartments interrelated to the central compartment by constant rates such as K12 or K21 [11]. Often mammillary models have compartments that can be actual physiologic region e.g. extravascular fluid and blood but is not represented by any specific area of the body.

Physiological based models have one or more than one compartments that demonstrate a distinct organ in the body with those organs that are linked with the blood flow [12]. Physiological based models usually require tissue and plasma concentrations and the parameters should resemble the literature values. However, mammillary PK models can be represented by blood or plasma concentrations only. Consequently, the application of physiological based models to clinical data is complex but it gives the understanding of the disease and physiologic effects in drug nature. It can also provide an opportunity to render preclinical outcomes to clinical surroundings. The simple mammillary open model is a two-compartment model where the drug is introduced in both, central and peripheral compartments **Figure** 2.

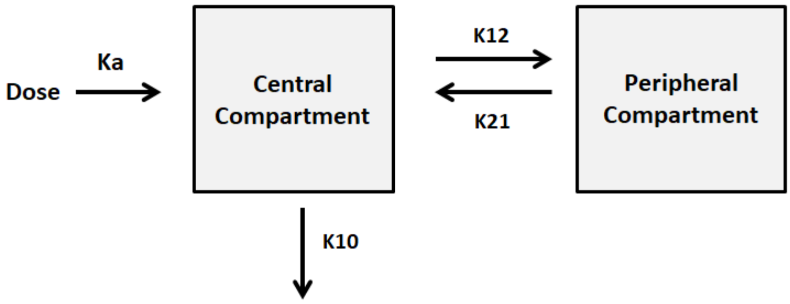


Figure 2: Two-compartment model. K12, K21 and K10 are first-order rate constants: K12 = rate of transfer from central to peripheral compartment; K21 = rate of transfer from peripheral to central compartment; ka = rate of absorption and k10 = rate of elimination from central compartment.

- **One-compartment model**

In one-compartment model, the central compartment (X1) consisting plasma or serum of blood that is used for sampling. The body represents kinetically homogenous division after the administration of the drug which means that the drug is distributed instantly all over the body and the drug rapidly equilibrates between tissues being highly perfused with blood such as heart, kidneys, lungs, brain and liver [13].

- **Two-compartment model**

In two-compartment model, the peripheral compartment (X2) consisting organs and tissues, the body is resolved into both central and peripheral compartments. It involves tissues that are not well perfused with blood such as fat, skin and muscle. After the administration of drug into central compartment, the drug is distributed in central and peripheral compartment but the distribution is not instantaneous due to less perfusion of tissues [13]. Inter-compartmental distribution follows a first order process.

- **Multi-compartment model**

In multi-compartment model, the distribution of drug is into more than one compartment [13].

2.2. Meta-models

Meta-analyses mean "the analysis of analyses" [14]. These analyses from numerous subject studies are potentially strategic analyses of collective results such as mean to incorporate outcomes and create summarised calculations. Meta models play vital role in the comparison of the efficacy and safety of novel therapeutics with those treatments that are missing individual data e.g. to compare the treatments with the products that are in competition [15]. They are also useful for the re-examination of the data from the studies that has mixed results [16]. Meta models can define the progression of disease or pharmacodynamics and are currently used commonly in the drug development to make the success or failure decisions [17]. However, for the meta-analysis few steps should be taken into consideration such as:

a) Before commencing the work, the aims and objectives of the studies must be outlined.
b) The data to be used should be comprehensive, unbiased and compatible. Only the successful trials data should not be included.
c) Between treatment arms variability and between subject's variability must be defined.
d) The aggregate data and individual data should merge sensibly like the method of combining the data should depend on the model [18].

The ambiguity of model is overlooked by the practice of choosing one model from a sequence of projected models and producing interpretations on the base of particular model. This could result in spoiled analytical presentation and ignore the better structures of other models.

2.3. Bayesian model

Bayesian model averaging is the practice of combining models and notifying the ambiguity of the model [19]. The Bayesian method is usually used in places where the drug has many models in the literature and the decision of choosing the appropriate model is not certain to evaluate novel study. The estimates of the accessible models can be definitely fitted and a particular model can be established which integrates many models. Consequently, the Bayesian model averaging approach permits the contribution of all the models for the simulation with the pre stated principles for the input to be decided according to the worth of the model or data and many other features [19].

2.4. Pharmacokinetic and pharmacodynamics models

Pharmacokinetic and pharmacodynamics models are significant for associating pharmacokinetic information to clinical settings and involve drug effect [20]. Continuous pharmacodynamics metrics in models usually appear as a continuous function with concentration effect bond. The concentration in pharmacodynamics model can be defined as direct drug concentration in central compartment or as indirect in which the response of pharmacodynamics lags after the drug concentration in plasma. The discrete effect of pharmacodynamics models uses logistic equations frequently to transform the influence to a probability in individuals that can be linked to pharmacokinetic model. These discrete pharmacodynamics effects include treatment, success or failure and the adverse effects. The class of pharmacokinetic and pharmacodynamics models are exposure response models in which instead of time, a metric that defines steady state drug exposure is an independent variable such as maximum plasma concentration (Cmax), area under the curve (AUC) and dose.

2.5. Population model gears

Population modelling demands precise information of covariates, dosing and measurements. These models include number of components such as stochastic models, covariate models and structural models. Stochastic models demonstrate the random effects or variability in the data evaluated [21]. Covariate models define the effect of factors like time course of response in a disease or demographics of individual while structural models are demonstrated as differential or algebraic equations and define the measured response time course.

Advantages	Disadvantages
Pharmacokinetic analysis generally include individuals taking drugs	Comparatively large number of patient is involved (more than 40)
Can deal flexible study designs that take place during treatment	Difficult pharmaco-statistical analyses
Few samples are required from each patient involved in the study	Compilation, collection and verification of large amount of data is required
Opportunistic sampling could be cost-effective	Building model could be tiresome, time consuming and labour intensive
Quantification and screening of covariates for determining variability is needed	The diagnostics of models can be complex and time consuming
Inter-individual and intra-individual variability can be differentiated	Problems with controlling missing data such as all covariates in every patient
Modelling software is easily accessible such as NONMEM and R	

Table 1: Advantages and Disadvantages of population pharmacokinetics modelling

3. Population methods

Population pharmacokinetic modelling approach is practiced in groups that are comprised of more than 40 individuals. In this study, instead of individuals, population is evaluated. Patients taking different doses on different timings are sampled. Population pharmacokinetics can predict oral bioavailability and the drug clearance. The most repeated value (mode) is usually used as a parameter which helps to achieve mean of population with the increase of patients. Therefore, in population observation, the information gained by each individual is used to evaluate their potential value of parameter. The accuracy of these parameters depends on extent of data estimated from each individual and on the difference between their predicted values and standard population value.

Population pharmacokinetic approach is not a substitute method for the existence of sparse data or model building with many covariates because there are limitations in dealing with observed sparse data in population method. For instance, more than one data point should be available from each patient else there will be mystified inter individual variability. It is claimed in clinical perspective that a covariate should be included in a model only if it sufficiently decreases the pharmacokinetic variability to alter the prescribing. For instance, in the modelling gentamicin pharmacokinetics, renal function must be involved. When there are more than two covariates present in a model such as age and sex, the problem of masking rise in defining the source of variability and ultimately these complex models increase the errors in prescribing and are difficult to practice clinically.

4. Application of population pharmacokinetic models

Population pharmacokinetic modelling is labour extensive, time consuming and a complex method [22]. Population pharmacokinetic model gives the appropriate prediction of unknown but accurate values of pharmacokinetic parameters like all mathematical models. Plasma concentrations estimated in the model are uncertain up to some extent due to the ambiguity involved in the true value of the evaluated parameter from a data in which model is fitted. According to a saying it can be said that ‘all models are wrong, but some are useful’.

Population evaluations have several beneficial clinical applications like in those subjects for whom traditional pharmacokinetic analyses is hard due to difficulty in recruiting, such as patients under intensive care or infants.

Population pharmacokinetics is enormously practiced in Australia and has the possibility of better-quality clinical results by prescribing individually [23]. For instance, population pharmacokinetics approach is used to create a dosage nomogram for caffeine treating infants affected by apnoea of prematurity [24].

Population pharmacokinetic approach is a developing and significant measure of drug development, clinical and pre-clinical studies, and for investigation of post marketing. The pharmaceutical industry reveals outstanding reviews [25] and regulatory perspectives [26], and web based guidelines generated by regulatory agencies [27], [28]. However, these studies are playing great role in clinical application and research in an extensive range of patients and situations such as clotting disorders [29], serious infections [30], diabetes [31], pregnancy [32], malignancy [33], organ transplantation, arthritis, self-poisoning [34] and apnoea of prematurity [24], [35].

Many aspects should be taken into consideration for the pharmacokinetic model evaluation. The parameter estimation usually differentiates models that are at initial stages of development and eradicates inadequate models. For further stages, simulation based approaches like visual predictive check (VPC) are beneficial when models with limited subjects are evaluated in final model [36]. For model diagnostics, Karlsson and Savic have given tremendous evaluation [37]. Model evaluations must be opted for the satisfaction and surety of the suitable model for proposed use.

5. Population pharmacokinetics modelling methodology

Population pharmacokinetics is the study of population where each individual is assessed simultaneously with nonlinear mixed effects model refers to the parameterization. Nonlinear mean that the variable which is dependant such a concentration, is related to the associated to independent variables and model parameters nonlinearly. Fixed effects refer to the parameters that do not change in individuals while random effect refers to those parameters that change through individuals.

The main aim of population pharmacokinetic modelling estimations is to look for the parameters of population pharmacokinetic and source of variability. The rest aims consist of observed concentrations of the dose administered by detecting the predictive covariates in evaluated population. Like single subject analysis, population pharmacokinetic approach does not demand scheduled time for sampling nor many observations from each individual. Therefore, few observations from each subject or sparse data and combination can be analysed.

The main five key parts for building a pharmacokinetic model include; data, structural model, statistical model, covariate model and modelling software. Structural models define the concentration time course in the subjects. Statistical models describe random variability in population that not explainable such as between occasion variability, between subject variability or residual variability. Covariate models demonstrate variability that is estimated by the characteristics of the population such as covariates. Modelling software such as nonlinear mixed effects modelling software combine data and models and apply the method of estimation to evaluate parameters for the statistical, structural and covariate models which define the data [38], [39].

5.1. DATABASES

Population analysis requires appropriate production of databases that is the most critical and time consuming part of the analysis [1]. To ensure the accuracy of the data, it should be well inspected. Before modelling, the graphical examination of data can detect possible errors and problems. Data records could reveal errors during the beginning of model evaluation or during data cleaning such as temporary or rapid fall of concentration which can be observed if they warrant an error that could harm the development of model.

Every evaluation consists of a lower concentration limit, in which the concentration could not be calculated appropriately if it is below that limit. On the calibration curve, the lower limit of quantification (LLOQ) is considered as the lowest standard which is 80-120% accurate and

20% precise [40]. The data below the limit of quantification (BLQ) is the data that is below the lower limit of quantification. If there are any samples in the data that are below the limit of quantification then the data detected close to the lower limit of quantification is normally censored. The effect of censoring can be observed by adding the line lower limit of quantification horizontally on plot of concentration vs. time. However, many studies show that the influence of censored data changes according to the circumstances when dealing with the below the limit of quantification data in population modelling [41]–[44]. Censoring could interpret variations in the outcomes when practised on the same data as population modelling approaches have more strong impact of censoring by lower limit or quantification than the methods of non-compartmental studies.

5.2. Structural model

Structural models have allegations for the selection of covariates [39]. Hence, evaluation of structural models should be cautiously done. The structural model is equivalent to an absorption model that defines the distribution of drug in blood for extravascular dosing and systemic model that defines kinetics after intravenous dosing. Mammillary compartment models take superior place in the literature, although pharmacokinetic models based on physiology play vital and developing role [12], [45].

Concentrations generally display one, two or three exponential phases when a particular part of the body gives data, which ultimately can be presented by systemic model with one, two, or three compartments respectively. By the plot of log concentration vs. time, the understanding of suitable compartment could be accomplished. When log concentrations decrease or increase with steady state in constant rate infusion, every distinct linear phase will require personal compartment.

Models with fewer compartments do not define the data accurately and ultimately illustrate bias in residuals vs. time plots while models with excess compartments display slight parameter estimation enhancement for increasing the number of compartments. Thus, the selection of number of compartments should be sensibly done. For extra peripheral compartment, parameters will meet the plasma concentration values that have slight influence such as low inter-compartment clearance CL and high volume or vice versa; or the parameters could be evaluated inappropriately. A significant attention should be paid to the accurate prediction of first order elimination. The rate of elimination in first order system is proportionate to concentration whereas clearance is constant.

The law of superposition illustrate the concept of increase in concentration with the increase of dose [6]. On the contrary, the rate of elimination is not dependant on concentration for zero order systems. Concentrations will rise by more than two folds when dose is doubled, as clearance in dependant on dose. With the rise of concentration, elimination progressively transfers to zero order state from first order state and saturate the elimination passages. To measure saturable elimination, pharmacokinetic data gathered from the population received single dose of drug is hardly enough, hence comparatively high range of doses is required. If steady state kinetics is not able to be estimated from single dose data then multi and single dose studies both show saturable elimination. Indication of nonlinearity could be revealed by non-compartmental or graphical studies like, dose-normalized AUC dependant on dose, dose-normalized concentrations that cannot be superimposed, multi-dose Css or AUCτ which is greater than estimated by single dose clearance and area under the curve.

For one compartment model and described rate of dose, saturable elimination is typically depicted by Michaelis–Menten equation [6].

$$C = \frac{A}{V}$$

$$\frac{\mathrm{dA}}{\mathrm{dt}} = \text{dose rate} - \left(\frac{\mathrm{Vmax} * \mathrm{C}}{\mathrm{Km} + \mathrm{C}}\right)$$

where dA/dt show rate of change in the amount of drug, Vmax is represented as the maximum rate of elimination and Km depicts the concentration related to semi Vmax. When C is less than Km, the rate converts to Vmax/Km*C in which Vmax/Km is inferred as the apparent first order clearance but when C in greater than Km, the rate comes to be Vmax in which it is interpreted as apparent zero order clearance. The extensive interrelation of Vmax and Km can make the estimation challenging for both as random effects parameters such as the segment of between subject variability. Generally Vmax is assumed as a function of accessible amount of elimination enzymes or transporters while Km is assumed as a function of the structure of the eliminating enzyme or transporters and drug.

The saturable elimination involvement into plasma concentrations must be analysed prudently in the areas of drug elimination, framework of drug and in the route of administration. As example, saturation in active tubular reabsorption drugs increases renal clearance and lower concentrations under the anticipated values from superposition whereas saturation in active renal tubular secretion lowers renal clearance but rise concentrations above the predicted value from superposition.

The bioavailability (F) is defined by the fraction of the dose that is administered by extravascular routes and goes into blood stream. The drug that is not absorbed by the body does not influence blood concentrations and consequently the resultant concentration appears lower due to the absorption of fraction of actual dose (F). The amount of drug absorption is dependent on the route of administration. However, it also can be influenced by the absence of physical entrance of drug in body for instance the residual of per os dose in gastrointestinal tract, during absorption transformation to a metabolite like drug cleared hepatically, accumulation or precipitation at injection area or a slow absorption component that is identified during study plan such as subcutaneous administration to lymphatic uptake of compounds.

Absolute bioavailability is referred as complete availability of dose such as from intravenous administration where F will be 100%. It can be predicted only with the simultaneous existence of intravenous and extravascular data.

5.3. Statistical model

The statistical model defines variability in the structural model. In pharmacokinetic model, the basic sources of variability are between-subject variability (BSV) and residual variability. The BSV shows variation of parameter in the subjects while the residual variability (RUV) is the variability that is not described when other sources of variability are monitored. Between-occasion variability is also expected by some studies in which the administration of drug in each individual takes place on more than two occasions that could be divided by adequate interval for the variation of fundamental kinetics in between the occasions. It is significant to build a proper statistical model for covariate estimations, simulations, appropriate use of models and to demonstrate the extent of residual variability in the data [6].

Residual variability results from numerous sources such as the model misspecification, assay variability, and miscalculations of sample time collection. As between-subject variability, residual variability model is selected on the basis of the nature of data to be estimated.

5.4. Covariate model

In pharmacokinetics calculations, it is important to identify the covariates that can predict the variability of pharmacokinetic. The potential covariates are generally selected by the class of drug, physiology or the identified properties of drug. For instance, drugs that are extremely metabolized contain the covariates commonly like genotype, weight or liver enzyme. The covariates should also go through the preliminary evaluation as the extensive run time could create a problem. Thus, the number of covariates in the model must be limited. Covariate screening can decrease the amount of evaluations using comprehensive additive models, techniques dependant on regression, or by correlation analysis that estimates the significance of covariates selected. Covariates are distinctly verified without covariate screening and all covariates are involved which are according to the required measures. The covariates identified in screening are individually evaluated with screening and the related covariates are all incorporated. The selection of covariates for nested models depends on the parameter estimation and likelihood ratio test (LRT). Hence, the specified levels in advance such as $P < 0.01$ or greater are set before the model based evaluations and the statistical significance can be caused by covariate effects. Then covariates are deleted backwards and fluctuations are analysed by LRT at tough criteria of parameter estimation. This method ends after the testation of all covariates and additional simplification of final model.

The inclusion of just statistically significant models in the model can create selection bias by practising stepwise method in model building. These models can result in exaggeration of significance of selected covariates. Multiple covariates evaluation with the extreme or moderate correlation such as weight and creatinine clearance can also cause selection bias that ultimately halts the true covariates discovery.

If values are continuous in sequence, extent and substance, covariates will also be continuous. On the other hand, if values are not connected and distinct or establish different classes, covariates will also appear distinctly which should be dealt differently. Both data should guarantee the physiological results by the parameterization of covariates.

5.5. Modelling software

There are number of available population modelling software. The selection of the appropriate package should be taken into attention considering the support for package, awareness of users with the package and the extent of package reputation with the regulatory reviewers. Many pharmacometricians are experienced in few packages (just one or two). The idea of parameter estimation is implied by most packages in order to reduce an objective function value (OFV) by practicing maximum likelihood estimation [6]. The calculation of the likelihood is much complex in population modelling than only fixed effect models [6]. At the time of population data fitting, the concentration estimated for each individual is influenced by the variance in population parameters and each individual parameters, and the variance in each values of predicted and observed concentrations. The evaluation of marginal likelihood depends on the random effect (η) and fixed effect population parameters. There is no existence of analytical solution for marginal likelihood. While looking for maximum likelihood, numerous approaches were applied for the approximation of marginal likelihood. FOCE and LAPLACE are the older approaches that estimate the true likelihood with additional simplified function [46].

Recent approaches such as SAEM contain stochastic elimination and filtering approximations partly by iteration of trial and error. Every approach of estimation comprise of pros and cons such as stability in over parameterized models and accuracy of parameters and complexity of primary parameter predictions [47], [48]. In nonlinear mixed effect model, the estimation method of original first order is of concern that results with biased estimations of random effects. The estimation methods and the difference in their approaches are often considerable. However, it is sensible to apply two or more methods in the early phases of model building such as, by estimating goodness of fit with stimulated or predicted data.

- **Modelling with R**

R is open source software environment and data analysis language for statistical computing and graphics. It can be run on diversity of Windows, MacOS and UNIX platforms. It can be downloaded from http://www.r-project.org. Multiple online learning sources of R are available. R software provides the combined collection of facilities to calculate, manipulate and display data graphically. It also offers the facility of:

- Data storage and handling it effectively
- Numerous operatives to evaluate groups, particularly matrices
- An integrated and rational suite of tools that aid in effective data analysis
- Service of graphical data analysis that can be displayed on the computer directly or can be provided as a hardcopy.
- Offers a programming language (“S”) which is an effective and well-built to deal loops, user defined recursive roles, facilities of input and output and conditionals.

R is a mean of novel emerging approaches of interactive data analysis and its rapid development has been expanded by a huge list of packages. Nevertheless, programs in R are temporarily written that are for only one study of data analysis.

Population models play an essential part in the regulation, development and appropriate use of pharmaceuticals [6], [49]. Nonetheless, the methods are really time-consuming to make predictions from population models and left the enthusiastic pharmacometricians with the use of special software that concise it's broader implication [50]. The flexible and sophisticated model output and data plotting are conceivable by the latest developments like ggplot2 package [51], [52] for the statistical language [53] and R data analysis. The models are required the process of re simulation and manual update to inspect different values for model parameters.

Advances in R and its packages specifically Shiny package have given an opportunity to R operators to display the output to web browsers for R [54]. Shiny, established by Rstudio is a package for R that can be installed in R or Rstudio. The installation of packages in R has numerous ways and the installation depends on the R interface and user's operating system. RStudio that is an integrated development atmosphere for R can be installed from http://www.rstudio.com/. To install packages, Tools and install package can be used. Further dependencies of the package will be installed automatically by RStudio.

The broad spectrum nature of R language has allowed the programming of interactive pharmaco-metric models with the package of Shiny that ultimately creates a web- browser interface which is accessible by internet access on the any computer. Some tools developed by Shiny package and R comprising R code can be seen without the installation of R software. These include such applications that are meant to educate students at high school or a tool related to the population model simulation along with simulated variability. To operate R and Shiny package, just prior knowledge of R language is needed which is more complicated in other methods of web page designing. Berkeley Madonna's software gives an access to substitute method that enables the models to specify as differential equations and the by the usage of sliders and radio buttons, simulated results for different parameters are presented [55]. This main objective of Berkeley Madonna is continued by R and Shiny that gives reactive update of output with the change in input by the help of widgets. Due to the blend of extended packages and flexibility in R language, the pharmacometricians are able to regulate the coding every component of a population model, attained output and the look of the user interface for the application. However, learning R and Shiny simultaneously is not suggested.

- **Shiny Application**

Building of Shiny applications require two R scripts that have an interaction in between them;

- A server script that is named as server.R (can be renamed as required) that integrates commands for the data processing, user input and output with the means of R language and from installed packages functions

- A user-interface script is named as ui.R that regulates layout and appearance of the application.

For the learning of Shiny applications, RStudio has introduced tutorials and exercises on the website of Shiny [56]. These tutorials are supported by the articles defining Shiny skills, the pages of references for Shiny functions and a list of examples containing code. RStudio has also referred eleven built in examples in Shiny package. It is compulsory to install Shiny package and its dependencies to run the Shiny applications in RStudio or in R and the required R scripts (ui.R and server.R) must be present in the same directory. To present the application from RStudio, ui.R and server.R scripts are needed to open on RStudio and the function of ‘RunApp’ present in the top right corner should be clicked. To launch the applications from R, working directory is required to set at the place of application folder and at the end RunApp option is required to use. Ultimately, a Web browser window will be open by Shiny where everything will be displayed.

- **User-interface (ui.R)**

There is number of built-in widgets and modifiable layouts for applications in Shiny which enables the effortless and easier building of user-interface. Creators can choose any of the existing options of layouts that are adjustable to the sizes of different browsers of devices such as computer, phone and tablet. The tabs or sidebars can also be included that distinguish the input and output. Any alteration can also be made on the displayed layout after meeting particular conditions of input. The ui.R scripts include the code that instruct the layout of application, its appearance, widgets of input such as sliders, check boxes, buttons, selection boxes and so on, and the output. The basic components that define the user-interface of the application are;

```
fluidPage(fluidRow(

h2(“Heading”),

plotOutput(“plotCONC”),

sliderInput(“Title”, “Covariate:”, min = ‘value’, max = ‘value’, value = ‘value’, step = ‘value’),

align = “center”))
```

In the layout function, all code required for the user-interface contents should be in the brackets. The functions of layout (as stated above) like fluidPage is required to make a canvas for the interface and fluidRow is used to position the widgets of the user-input such as sliderInput is used to generate a slider and plotOutput function is needed to plot an object. Every function of layout possess its outline for placing elements while other functions such as fixedPage and navbarPage depending on their functions are capable of creating pages with different designs. Nevertheless, each function follows the similar classified structure in which the functions like sliderInput or widgets are placed in a layout function of fluidPage and in a positioning function of fluidRow. The same level functions are placed in sequence that are separated by “,” within their higher level function.

If there is any error from Shiny or R packages, the message of an error will prevail at the time of opening the application in the Web browser and after closing the application it will be seen in the R console. The evaluation of the appropriate opening and closing of the brackets prior to initiation of the application can aid to avoid error messages. After getting a detailed ui.R, functions written at the beginning in the script become more complicated to detect. For this problem free source code editor software or RStudio can help at the time of writing code as they highlight any error made such as unclosed brackets. The minor errors can be detected by these functions but if there is an existence of a major error with a non-functional application, a Web browser page will be displayed with the grey colour. Generally, other messages relating to error give a number of the line where the code is written or provide the name of function in question. While evaluating the pharmacometric model coding, it is sensible to write a generic R script to confirm its successful working before integrating the model in Shiny application.

The arrangement of server.R code has a critical impact on illustrating commands for the application whereas it enhances the speed or application and reduces unnecessary computation. The ShinyServer function needs input and output object from ui.R. Objects that are influenced by the input widgets present in ui.R e.g input$KA, are called “reactive”. Whenever there is change in input from a widget, reactive object also changes its value accordingly. To process and describe the reactive objects, the related expressions should be written in a render* function in order to get a reactive output to ui. The term (*) represents the description of the output object such as a text, plot or a table. By the render* function, a reactive expression used to deal with a list of reactive data frames which could be directed to user-interface.

The function of renderPlot that comprise of input objects such as (KA and V), calculation expressions for concentration and ggplot2 [52] to plot concentration vs time (plotobj), will update to reflect the change of every widget and the updated plot object will be saved for the output object as plotCONC. It is recommended to enclose code within ShinyServer and render* functions to avoid the sluggish speed of the application due to detailed code. At the start of the script, code is run just one time when the application is commenced, it does not require running every time with the change of widgets. Thus, it is considered an ideal area to load libraries, datasets, define constant expressions or source code. All of the functions and libraries could also be saved in another script that is named as ‘global.R’.

The code written in ui.R script is called in sequence by Shiny (from left to right and from top to bottom). The layout function arranges the elements in sequence and the user-interface show the elements accordingly.

fixedPage(fixedRow(

column(10, h2(“Title?”, align = “center”), offset = 1)), Above is the example of code that defines the layouts. A fixedPage layout aligns elements in a fixed width and in rows and columns such as widgets and output text or plot. FixedRow command show the elements in a same line and the column command allows the space horizontally and the elements are ordered in a wide grid of 12 units. However, fluidPage and fliudRow organize the page layout according to the browser dimensions in which the application in open. Each element is given a width and a column. If there are more elements to be added, they should be written under the heading with a new row by using fixedRow and column similarly as above while the dimensions can be defined according to the requirement of the content. The layout fixedPage is not restricted to fixedRow function, similarly fluidPage and fluidRow act in a same way. The utilization of sidebarLayout in the application forms a sidebar that appears as a bordered part in a user-interface along with a background. sidebarLayout can arrange the elements to the sidebar with the function of sidebarPanel or by mainPanel function to an unformatted area, instead to assigning elements into columns. There are other functions for layout such as tabsetPanel and navlistPanel which make the sections of the user-interface e.g. tabs on the navigation list divided for different tables and plot that can be mixed-up if not separated [56].

The widgets are elements that are interactive and give the users an opportunity of exploration of different categories or values of variables or parameters. The selected values are stored by the widgets and called by the server.R, render* functions or reactive function process them for output which ultimately are directed to the user-interface for the presentation. Hence, if a widget is changed, the value called by server.R will also change followed by the change in the output. There are plenty of prebuilt widgets combining R functions and an analytical thread of arguments in Shiny package.

The help can be acquired by writing in R the symbol "?" and then writing the name of input function of a widget (e.g. ?fixedPage). Each function of widget is named in order to be called by server.R from ui.R that is not visible to the users while a label argument that is also important to write, is visible to the user. To complete the function, other arguments required depend on the type of widgets such as selections for selection boxes and for sliders min, max, value and step values. Some widgets have advantage over others that they confirm users can select only possible values by limiting the biological possibilities or by restricting the code in server.R. The example of the widget code in ui.R for the selection box for dose frequency in sidebarPanel is presented below:

sidebarPanel(selectInput("FREQ", "Drug Frequency:",

choices = list("Once a day" = 1, "Twice a day" = 2, selected = 1)),

This code will create a selection box by the selectInput function. "FREQ" is the name of widget that will be called by the server.R as input$FREQ. the user-interface label here will be "Drug Frequency". An argument is also needed for choices such as a list of labels assigned to the numbers. It is modifiable and 'selected' can be included which is the allocated number to value from choices to be displayed on initiation of application. It can also allot the box width in pixels and give option for multiple choices. On the other hand, a slider widget can also provide these limitations in which the user can only slide the bar to the available values such as 1 = once a day, 2 = twice a day and so on. The example of code required for the slider is as follows:

sliderInput("FREQ", "Drug Frequency:",

min = 1, max = 2, value = 1, step = 1),

Rather than a slider, a selection box can give a precise classification by the help of text. Widget selection should be made on the basis of user audience concern, capability of communicating the aim and kind of variables to which it will attach such as categorical vs. continuous. The variety of widgets can mess the user-interface easily, thus it is recommended to limit the free availability of widgets by the user. For this purpose, an option provided by Shiny can be used that has an ability of concealing or displaying elements of widgets for particular situations with the usage of renderUI in server.R and conditionalPanel in ui.R. The example of conditionalPanel in checkboxes is as follows:

conditionalPanel(condition = "input.FREQ == 2", checkboxInput("Drug1", "Missed on 2nd Day:", value = FALSE), checkboxInput("Drug2", "Drug Doubled on 3rd Day:", value = FALSE)),

The conditionalPanel has a condition argument that is assessed frequently to decide the display of the following elements. However, these two conditionalPanel checkboxes will be displayed only when the selection box widget (Drug Frequency) will be selected as 2. The checkbox widget doesn't have numbers, thus the value argument relating to the first input is only dealt by TRUE and FALSE, if it is TRUE, the output will be affected while FALSE doesn't influence the output anyway.

The other types of widgets include radio buttons (radioButtons) that are most used in pharmacometrics, downloadbuttons (downloadButtons), and slider ranges in which two values can be selected on the ends of slider (sliderInput).

Heading (h) are the functions in Shiny used to define the heading and like widgets, they are coded in layouts in a fixedRow (positioning function). The heading can be made in variety of sizes and the code used for the size is h1, h6 and p. The first level of heading is largest (h1), the sixth level header is the smallest (h6) and text paragraph is generated by (p). The align argument is used to set the alignment of heading as illustrated below:

h2("Title", align = "center")

Breaks coded as br() and lines coded as hr() are used in the application to make a partition of the heading and the functional elements. They both can adopt the level of positioning functions (fixedRow) or similar to widgets.

*Output functions in Shiny call from server.R the objects that are reactive to the user interface. (*) in *Output describes an object i.e. table, text or a plot. In the user interface, they are constructed in an order similar to widgets by writing the function *Output in the user interface script in a positioning function. Following is a ui.R code where a reactive plot is included in the mainPanle by the use of plotOutput which accordingly updates with a widget change:

mainPanel(plotOutput("Output argument", height = 'value', width = 'value'))

The reactive object should be named to identify each *Output function when called by server.R. Labels cannot be assigned to *Output functions and output objects names are not visible to the users. Thus, in server.R, for the reactive object titles or headings are required in the expressions while in ui.R heading element is needed to build. Every *Output function can possess a particular argument due to its individuality such as plotOutput the arguments of height and width for specifying the plot dimensions. Other *Output functions own the names that define their goal such as imageOutput, tableOutput, uiOutput, htmlOutput and textOutput.

6. Pharmacoeconomic model

Pharmacoeconomics is related to the scientific authority where the value of drug therapy or pharmaceutical drugs is compared [57], [58]. The study of pharmacoeconimcs assesses the pharmaceutical products in terms of finance by its cost, effects, or efficacy. This study assists in leading scientifically towards the means of ideal allotment healthcare. Pharmacoeconomics emphasis on the pharmaceutical evaluation economically by practicing cost benefit analysis (CBA), cost utility analysis (CUA), cost minimization analysis (CMA) and cost effectiveness analysis (CEA) [59]. The quality adjusted life year (QALY) is a main health outcome of importance in pharmacoecomics evaluations that involves quantity and quality of life. Cost per QALY analysis is practiced by many studies. Economic evaluations of pharmaceuticals are progressively practiced and are executed in conjunction with the randomized controlled trials and decision analytic modelling approaches. The healthcare deciders focus and recognize the money value from healthcare interferences.

Pharmacoeconomics method plays a vital role in the economic assessment of several treatment decisions. The main complication associated to economics is scarcity that means it limits the choices for the allotment of healthcare funds. If the expenses in one region of healthcare are high, it will definitely affect the expenses to be made in other region with the limit amount. The economists use the method of prediction for the advantages by opportunity cost. Pharmacoeconomic evaluation method gives a prospect of defining the treatment options by estimating the amount of income to attain highest health benefit by money spent per unit, which can be gained by accessing opportunity cost of apportioning resources to a specific option of treatment.

With the development and licensing of costly pharmaceuticals, pharmacoeconomic evaluation prove to be extremely beneficial particularly for the developing countries where scarcity obstacles the resources for the implementing the ideologies of pharmacoeconomics for different treatment options and drugs. It is imperative in order to gain lowest cost with the maximum progression in the quality of healthcare and life [60].

6.1. Methods of pharmacoeconomics

Pharmacoeconomic evaluation methods are all similar in terms of evaluating input cost and the benefits achieved from the intervention of a drug [61]. Rather than just adding the cost of the drug, direct and indirect both costs are included in the price of drug therapy where direct costs are referred to the capital and staff cost whereas indirect costs may contain the losses related to the earnings and productivity or traveling costs to the hospital. A number of the costs are not easy to evaluate such as imperceptible costs for any discomfort or pain that is suffered by the patient. The variance among economic evaluations is stated in terms of extent of advantages as costs can only be defined in monetary form. The measurement of such advantages can be done in natural units like saved years of life by antiretroviral therapy or lipid lowering. The advantages can also be evaluated with regard to utility units like quality of life which include the physical activity evaluation such as psychosocial results like anxiety and mobility extent.

- **Cost benefit analysis (CBA)**

The cost benefit analysis measure the costs and benefits both in terms of monetary for the drug. This approach allows the evaluation of the expenses occurring in health area versus the expenses incurring in additional areas such as transportation and education. Due to an ethical opposition for giving importance to the monetary value instead of human health and life, some cost benefit educations are established. Nevertheless, monetary values are practiced for predicting the death or injury compensation in terms of health.

The perspective of interrelated decision makers is fundamentally required for the economic evaluations report. To meet the requirements of individual prescribers and governmental judges, the existing evaluation might be needed to analyse diversely. The social viewpoint may comprise of direct costs and indirect medical costs e.g. hospitalisation and drug prices (direct) and pain and productivity (indirect). Although, only direct costs are analysed by the viewpoint of main healthcare manager such as expenditures related to drug therapy, general practitioner consultation and laboratory observation. The social perspective of policy regulators is taken into attention as the economic analysis objective is to utilize the resources appropriately but the healthcare provider with a limited finance would ponder the additional costs of drug with more preference.

Multiple healthcare interventions are compared to gain the maximum benefits but the investment of healthcare resources could prevail with different schedule as compared to the benefits gained such as the comparison of precautionary therapies (statins with curatives) e.g. thrombolysis. Generally, benefits are aimed prior to any investment and to consider this optimistic time priority, upcoming consequences and costs are discounted in economic evaluation in order to show values by around 5% annual rate.

In economic evaluation, the adaptation of discount rate is only a suspicion where other uncertainties emerge with the deficiency of accuracy in costs and benefits analysis. The method of sensitivity analysis is practiced to handle these suspicions that also include modification of basic parameters and expectations to define their influence on economic evaluation. In an organized setting of clinical trial, the efficacy of an intervention might exaggerate the effectiveness in regular clinical practice. The cost effectiveness will be difficult to retain if the event rate is 15 % while the lipid lowering cost effectiveness over specific time duration has 25% decline in coronary event. Thus, in economic evaluation, sensitivity analysis is obligatory to analyse the influence of analytic assumptions [62].

- **Cost minimization analysis (CMA)**

Cost minimization analysis is very strict type of analysis that attentions completely on costs such as health services. Subsequently, this analysis is helpful in the similar health outcomes gained from two separate treatments which are required to be analysed individually. For instance, choosing the introduction of a generic drug instead of the branded that will give equal benefits with minimized costs. Generic prescribing is a great source of boosting cost effectiveness. Doctors gained an opportunity of easily understanding and implicating such kind of analysis extensively. Nevertheless, the therapies and programmes directing to altered outcomes cannot be evaluated by this form of evaluation [62], [63].

- **Cost effectiveness analysis (CEA)**

Cost effectiveness analysis is a term that is usually roughly used to denote every type of economic analysis. However, it appropriately discusses a specific type of evaluation where costs are stated in monetary terms and health benefits are estimated in terms of natural units such as saved years of life. Consequently, cost effectiveness analysis involves more than two therapies that have common objective of treatment but reveal different rates of efficacy. For example, if in severe reflux oesophagitis the aimed outcome is symptomatic aid, we can analyse the costs of each relieved patient with proton pump inhibitor comparing to the ones who use blockers of H2-receptors [62], [64]. Cost effectiveness analysis is widely practiced in economic evaluations still it is not ideal to apply this approach to compare completely different therapies having different benefits.

In the comparison of therapies the resource allocation evaluates the quantity of benefit obtained for the cost experienced. Therefore, the calculation of incremental cost effectiveness of each therapy is significantly required. Below is an equation that explains the incremental cost effectiveness ratio of one therapy (1) over another (2).

$$Incremental\ cost\ effectiveness\ ratio = \frac{cost\ (1) - cost\ (2)}{effect\ (1) - effect\ (2)}$$

$$ICER\ = \frac{Difference\ in\ costs}{Difference\ in\ effects}$$

The cost effectiveness (CE) plane is a significant measure used in cost effectiveness analysis and broadly applied in healthcare sector. The objective of cost effectiveness plane is to demonstrate the comparison in costs and effects between medical interventions, medical treatments or both combined. It aids in making sensible decisions by evaluating different strategies.

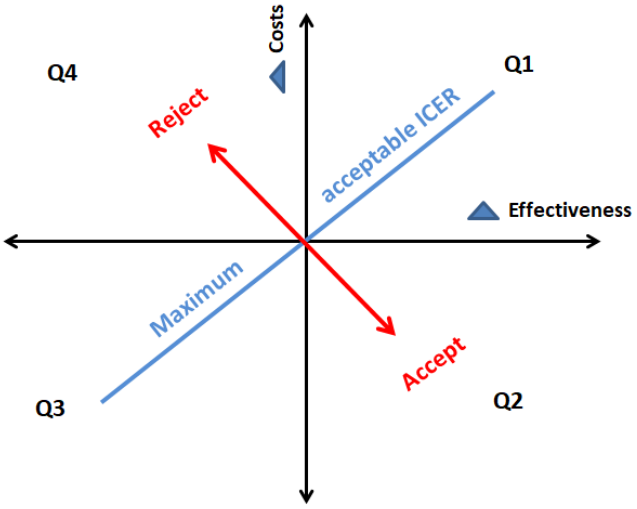


Figure 3 : Cost effectiveness plane

The cost effectiveness plane determines the results of incremental cost effectiveness ratios analysis ***Figure 3***. The second quadrant (Q2) demonstrate the high effectiveness and low cost of interventions which is dominant and results in the acceptance of the interventions while quadrant four (Q4) shows an area of rejection as it comprise of high cost and low effectiveness. Interventions in quadrant three (Q3) may be considered by developing countries as it has lower cost but also low effectiveness. On the other hand, developed countries normally consider quadrant one (Q1) in which new interventions result in high effectiveness and higher costs too.

- **Cost utility analysis (CUA)**

In cost utility analysis, the effects of therapy on quality of life and patient wellbeing are both measured in common unit. It is identical to cost effectiveness analysis where the outcome is predefined and the cost incurred is expressed in terms of monetary. Conversely, in cost utility analysis the measurement of outcome is not done on shared natural units. Health benefits in cost utility analysis are analysed on the basis of patient survival i.e. utility which is not comprised of a particular disease. Thus, cost utility analysis has an ability of comparing the worth of interventions in two or more medical areas. The utility measurement is a difficult task as the outcome measure such as QALY (quality adjusted life year) cannot be defined precisely that ultimately fails to transfer to other evaluation. Different viewpoints and priorities in different ailments can be gained by measuring quality of life. Specific cautions should be taken to empower league of QALY to provide evaluations of the value of money derived from the number of therapies [65]. These tables values cannot be compared as they are obtained from different approaches used at unscheduled time on variety of people [66].

II. Materials and Methods

7. Modelling in R and Shiny

Once R and RStudio were learnt, in order to practise the learnings and accomplish the aim of the study, modelling was commenced with simple plotting and calculations using some basic code of R. Later Shiny was explored and an application in Shiny was built by moving step by step in the creation of a model. Following are the steps that were used to enhance the learning in this study.

7.1. Basic code for plotting and PK parameters calculations

Based on data from Olmesartan concentration-time profiles obtained from two drug products in a bioequivalence study data (in house data), R code (See Attachment 1) was performed to plot dataas concentration vs time, log of concentration vs time and average concentrations vs time.. BLQ (Below Quantification Limit) was assumed as zero concentration value. Additionally, pharmacokinetic parameters Cmax, tmax and AUC0-t (by trapezoidal method) were calculated for all subjects and both formulations. Attachment 2 comprise of all the code used step by step for the successful plotting and calculations. Attachment 3 has the resulted plots of each formulation with the variations in the size and colour of the plots. The results for pharmacokinetic parameters required can be achieved from Attachment 4.

7.2. Model with Ordinary Differential Equation (ODE) in Shiny

After the successful attempt of dealing with the simple model in RStudio, Shiny was used to create simple models following by complexity in each new model built, in order to reach towards the targeted model.

At the beginning, model built created an application in the browser with the simple user interface that gave it an appearance (without any widget, slider, checkboxes, etc.). Later to increase the complexity, an ordinary differential equation (dA = -KE*A) was added in the model with fixed effects parameters (THETA) where the rate of elimination (KE) was estimated to be 0.09, and Volume of distribution equals to 25 (See Attachment 5). The R scripts used to create this model are available in Attachment 5.

7.3. Model with ETA in Shiny

Next, the model with the inclusion of ETA was built where ETA is represented by random effects as a quantity and demonstrates the difference in values of population and individual parameter. It is expected to be distributed normally or log-normally in evaluated population with zero mean and precise by its SD (Standard Deviation). In this model, number of subject (n) used was 10 and ETA for population volume (POPV) and population rate of elimination (POPKE) was used with the same ordinary differential equation (dA = -KE*A) (See Attachment 6).

7.4. Model with Confidence Interval (CI)

The previous model was then modified by the addition of confidence interval (CI) values that is used by statisticians to define the interval estimate in which true values of population parameters might exist. It appeared in a plot as a shaded region with the given colour (red), according to the defined CI limits. The percentiles used for the CI were 2.5 % and 97.5 % (Attachment 7). The concentration plotted in the model with CI was the median concentration of the population, where median concentration can be replaced by mean concentration that is shown in next model.

7.5. Model with Slide bars

In this model, slide bars were further included with an addition of code in both R scripts (ui.R and server.R). The slide bars added were for the dose administered, rate of elimination (KE), volume of distribution (V), standard deviation of V, and standard deviation of KE (Attachment 8). The aim of the added slide bars was to analyse the change in the plot with the change of slides when moved from one value to another. This reveals how these variables can affect the plotting curve or the concentration of drug. The concentration plotted in the model with CI was the mean concentration of the population.

7.6. Modelling with the two compartmental models

Further, the model was then updated to work with two compartmental models (Attachment 9), where concentrations from each compartment were plotted with varied colours. CONC1 was referred to central compartment while CONC2 was assumed for peripheral compartment. The volume of distribution in each compartment was defined by V1 and V2 in first and second compartment respectively. Dose of 300mg was administered here with a frequency of 1 that means one per day. This model was built without Shiny in order to confirm the appropriate working of the each code before integrating it into Shiny. Three ordinary differential equations were used (Attachment 9), which are as follows:

*dA[1] = -Ka*A[1]*
*dA[2] = Ka * A[1] - (K10 + K12) * A[2] + K21 * A[3]*
*dA[3] = K12 * A[2] - K21 * A[3]*

In these equations, A[1] predicts the amount of dose at the time of administration; A[2] demonstrates the amount of drug in central (first) compartment while A[3] depicts the amount of drug in peripheral (second) compartment. CONC 1 was calculated by dividing A[2] by V1, similarly CONC2 was calculated by dividing A[3] by V2. Consequently two data frames were created along with the calculations of their variables and concentrations that were later merged together with time to make concentrations vs time plot at the same chart for each concentration. Here the median of each concentration was used against time to make the plot. Moreover, the model was the improved by adding confidence interval for each of the concentrations in the plot (See Attachment 10).

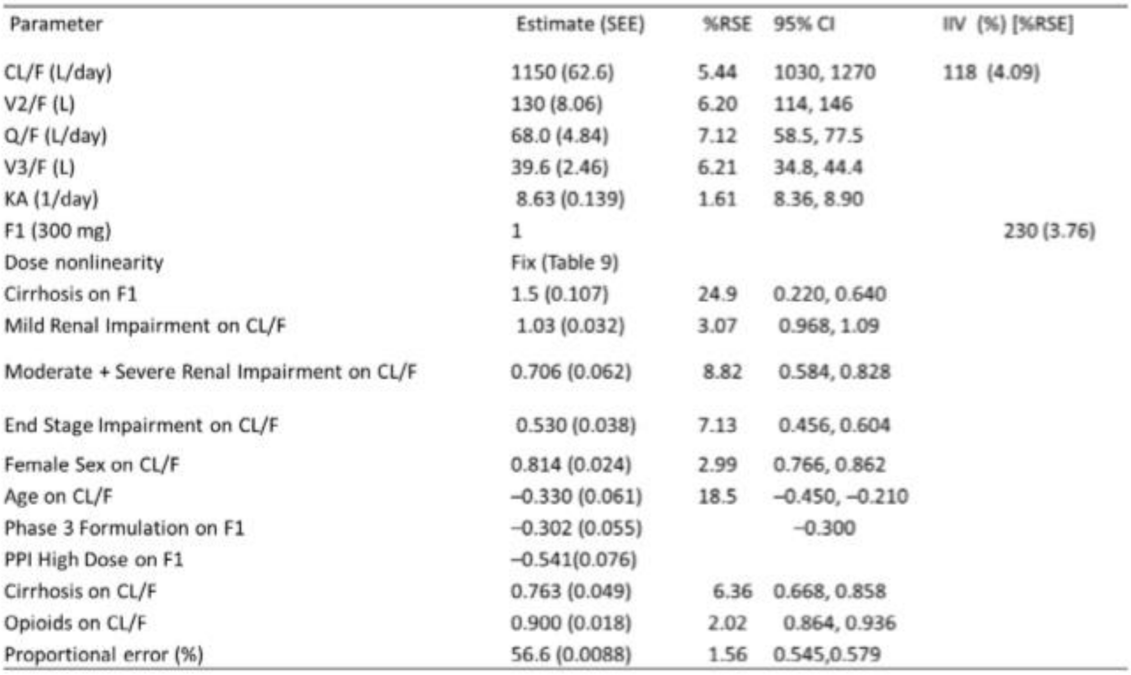
8. Final modelling

The aim of these entire model building was to get enough skilled for the building of the final model that is related to the population pharmacokinetic analysis of Glecaprevir and Pibrentasvir drugs. Mavyret® is the branded name of the fixed dose combination of these two drugs. Both Glecaprevir and Pibrentasvir are viral protein inhibitors: Glecaprevir (100mg) inhibits serine protease NS3/4A while Pibrentasvir (40mg) inhibits zinc binding hydrophilic phosphoprotein NS5A [67]. These proteins are significant for the viral RNA replication in hepatitis C, and by inhibiting these proteins this replication can be stopped [68].

In August 2017, Mavyret® was approved by the Food and Drug Administration (FDA) [69] for the treatment of hepatitis C virus (HCV) genotypes 1-6 with mild or no cirrhosis and with mild to severe kidney disease [70]. For the approval of Mavyret®, a population PK analysis was performed [71] or Glecaprevir and Pibrentasvir based on data from a population of patients with HCV (as monotherapy or combined therapy) , in order to classify the factors that can affect variability in the pharmacokinetics of Glecaprevir and Pibrentasvir..

8.1. Glecaprevir (GLE) modelling

A model was initially built only one drug (Glecaprevir), based on details described in the report from FDA for Mavyret®. Glecaprevir modelling in population pharmacokinetic analysis used data from subjects who received Glecaprevir with measurable concentrations. Quantity of dose administered in each subject was 300mg once a day. The popPK model was created with Shiny (Attachment 11) in order to observe the difference in the results with the changing of the covariates included as the selection boxes and checkboxes. This model had pretty much error and mistakes that needed to be altered to improve the appearance and working of the model such as the selection boxes were required to be replaced by the sliders or the check boxes for the ease in selecting the value. Later, with the modification of each model, many changing were made. The main purpose of this model was to deal with the values and equations required for the population pharmacokinetic analysis of Glecaprevir. As this model was not the final one, the number of subjects to simulate PK profiles was only 10. The values used for the analysis were taken from the table of parameter estimates (***Table 2***) [71]. The structural model was a two-compartment open model with a volume of distribution for the central compartment (V2/F) and for the peripheral compartment (V3/F), clearance (CL/F), rate of absorption (KA) and inter-compartmental clearance (Q/F). The two-compartmental pharmacokinetic model included also a first order absorption and elimination processes (***Table 2***) [71]. PopPK model showed a high inter individual variability (IIV) in bioavailability parameter (230%).



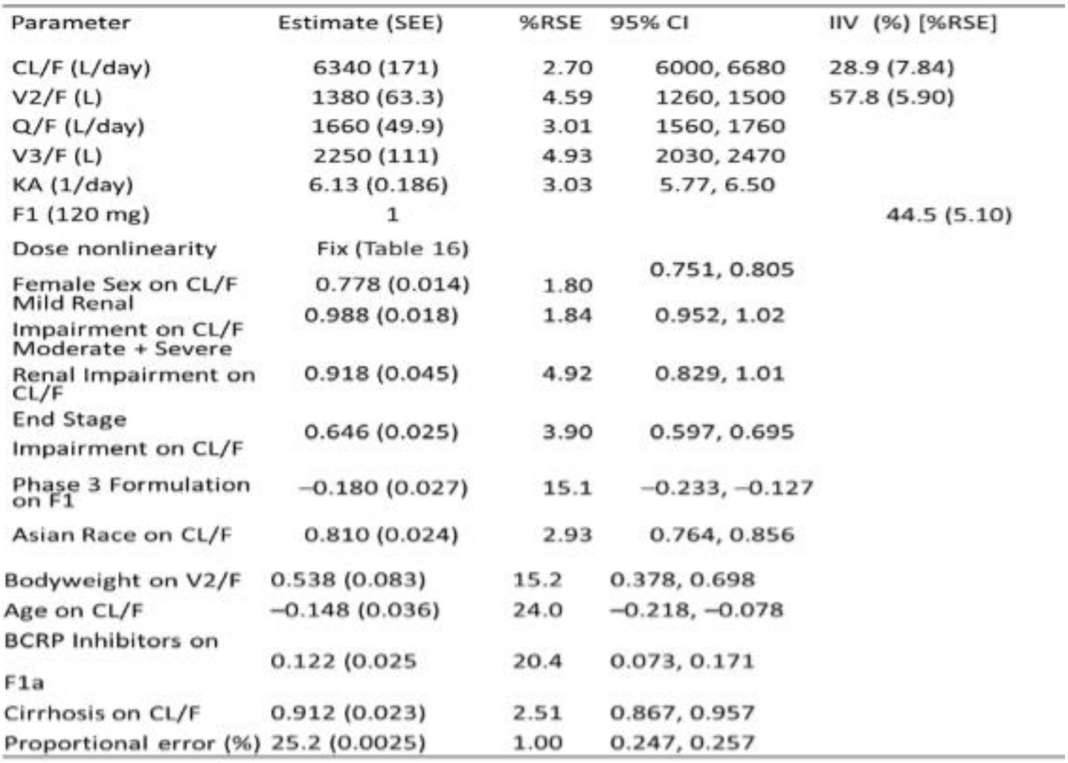
Parameter	Estimate (SEE)	%RSE	95% CI	IIV (%) [%RSE]
CL/F (L/day)	1150 (62.6)	5.44	1030, 1270	118 (4.09)
V2/F (L)	130 (8.06)	6.20	114, 146	
Q/F (L/day)	68.0 (4.84)	7.12	58.5, 77.5	
V3/F (L)	39.6 (2.46)	6.21	34.8, 44.4	
KA (1/day)	8.63 (0.139)	1.61	8.36, 8.90	
F1 (300 mg)	1			230 (3.76)
Dose nonlinearity	Fix (Table 9)			
Cirrhosis on F1	1.5 (0.107)	24.9	0.220, 0.640	
Mild Renal Impairment on CL/F	1.03 (0.032)	3.07	0.968, 1.09	
Moderate + Severe Renal Impairment on CL/F	0.706 (0.062)	8.82	0.584, 0.828	
End Stage Impairment on CL/F	0.530 (0.038)	7.13	0.456, 0.604	
Female Sex on CL/F	0.814 (0.024)	2.99	0.766, 0.862	
Age on CL/F	−0.330 (0.061)	18.5	−0.450, −0.210	
Phase 3 Formulation on F1	−0.302 (0.055)		−0.300	
PPI High Dose on F1	−0.541(0.076)			
Cirrhosis on CL/F	0.763 (0.049)	6.36	0.668, 0.858	
Opioids on CL/F	0.900 (0.018)	2.02	0.864, 0.936	
Proportional error (%)	56.6 (0.0088)	1.56	0.545,0.579	

Table 2 : Parameter estimates for the final Glecaprevir model [71]. **IIV = inter-individual variability, SEE = Standard Error of Estimate, % RSE = (Relative Standard Error),**

$$\%\mathbf{RSE} = \frac{\boldsymbol{SEE}}{\boldsymbol{Population\ estimate}} * \mathbf{100}$$

8.2. Pibrentasvir (PIB) modelling

Based on the popPK analysis described in the report form the FDA for Mavyret®, another drug model was built for Pibrentasvir with the intention of updating the successful modelling of each drug in a combined model. The dose administered in each subject was 120mg on the same regimen (once a day) as Glecaprevir. Like Glecaprevir model, the model for Pibrentasvir was also built in Shiny to observe the changings in results with the change of values of covariates (see Attachment 12) that are provided as a selection boxes for age, renal function and gender, while for Asian race and cirrhosis, checkboxes are available. Based on Shiny code, it is possible to observe any change in the plot by just changing the selected values. The number of subjects used to simulate PK profiles in this model was also 10, that was later increased to 1000 when a combined model was built. In ***Table 3*** [71] are presented the parameter estimates values used in the model,



Parameter	Estimate (SEE)	%RSE	95% CI	IIV (%) [%RSE]
CL/F (L/day)	6340 (171)	2.70	6000, 6680	28.9 (7.84)
V2/F (L)	1380 (63.3)	4.59	1260, 1500	57.8 (5.90)
Q/F (L/day)	1660 (49.9)	3.01	1560, 1760	
V3/F (L)	2250 (111)	4.93	2030, 2470	
KA (1/day)	6.13 (0.186)	3.03	5.77, 6.50	
F1 (120 mg)	1			44.5 (5.10)
Dose nonlinearity	Fix (Table 16)			
Female Sex on CL/F	0.778 (0.014)	1.80	0.751, 0.805	
Mild Renal Impairment on CL/F	0.988 (0.018)	1.84	0.952, 1.02	
Moderate + Severe Renal Impairment on CL/F	0.918 (0.045)	4.92	0.829, 1.01	
End Stage Impairment on CL/F	0.646 (0.025)	3.90	0.597, 0.695	
Phase 3 Formulation on F1	−0.180 (0.027)	15.1	−0.233, −0.127	
Asian Race on CL/F	0.810 (0.024)	2.93	0.764, 0.856	
Bodyweight on V2/F	0.538 (0.083)	15.2	0.378, 0.698	
Age on CL/F	−0.148 (0.036)	24.0	−0.218, −0.078	
BCRP Inhibitors on F1a	0.122 (0.025	20.4	0.073, 0.171	
Cirrhosis on CL/F	0.912 (0.023)	2.51	0.867, 0.957	
Proportional error (%)	25.2 (0.0025)	1.00	0.247, 0.257	

Table 3: Parameter estimates for the final Pibrentasvir model [71]. **IIV = inter-individual variability, SEE = Standard Error of Estimate, % RSE = (Relative Standard Error),**

$$\%\mathbf{RSE} = \frac{\boldsymbol{SEE}}{\boldsymbol{Population\ estimate}} * \mathbf{100}$$

8.3. GLE and PIB modelling (with Shiny)

Estimate values for model parameters and IIV presented in ***Table 2*** (Glecaprevir) and ***Table 3*** (Pibrentasvir) were used in an updated model where final PK profile simulations had to be done. The server script in Shiny controls the user input to show output in user interface. As shown in this model (Attachment 11 and 12), the server.R script consist of two parts where one part deals with the processing and calling reactive inputs and giving output inside the shiny server, whereas the other part include the non-reactive expressions and functions that are independent of widget inputs. The principle of dividing functions and expressions in both parts is to separate the functions or expressions that are required to evaluate every time when input changes with the ones that do not need re-evaluation again and again. Before introducing Shiny server, all the libraries of the required packages are loaded, including the time sequence, function for differential equation and ggplot2 themes. These functions are just executed when the application is started or it is re-opened in the Web browser. These commands do not react with the change of input; however they are placed in a nonreactive area i.e. outside of ShinyServer. The functions within the ShinyServer are re-executed every time with the widget change, thus reactive expressions and render* functions are placed inside the shinyServer as they depend on widget input. The reactive expressions used in this application are as follows:

```
shinyServer(function(input, output) {
sim.data <- reactive ({
   SEX <- input$SEX
   C <- input$C
   AR <- input$AR
   AGE <- input$AGE
   O <- input$O
   RI <- input$RI
```

In this code, SEX, C (cirrhosis), AR (Asian race), O (Opioids), Age and RI (Renal Impairment) are the series of input$X function that are used to make a data frame called sim.data.df (at the end) that includes the calculations of time and concentrations for further use in plots or for calculating the pharmacokinetic parameters. This data frame also changes according to the change in widget inputs. Changing the dosing regimen or covariate values allow the user to see rapid results simultaneously. The incorporation of variety of widgets offers to simulate different situations without changing the model code, R processing code for output or input dataset.

The differential equations are represented by the deSolve [72] package used in the server.R script (Attachment 13). The application use R language to simulate the population by sampling their parameters randomly so that each patient owns a parameter set. Here, the differential equation solver uses the input of differential equations and parameter sets to obtain the data of concentration-time for both drugs (Glecaprevir and Pibrentasvir) from 0 to 90 days with the difference of 0.02 days. In the subsequent data frame such as “sim.data.df” (Attachment 13), the mean, upper percentile and lower percentiles for Glecaprevir and Pibrentasvir concentrations are calculated. Consequently, the plots displayed in the user-interface show two solid lines, blue and red for Glecaprevir and Pibrentasvir mean concentrations respectively, whereas the shaded ribbon with the similar colours represent the upper and lower percentiles of each of their concentrations.

In Attachment 13, R generates random numbers that are simulated by number of random effect parameters (n) from normal distribution for every parameter, where mean is considered as zero and standard deviation is described according to the values placed in Table 2 and Table 3 for each drug. Each random effect parameter is corresponded with other parameter that is in the similar place of n-value log sequence, in which the values of population parameter are distributed log-normally and, for the corresponding parameter, the calculation uses the value of population and the value of each patient for the random effect. The population size here is dependent on the input n in this model that should be written in reactive expression in ShinyServer as the code to define parameter values is reactive. The number of population (n) was increased in this model to 1000 and the prediction of time-course was improved.

Several pharmacokinetic systems are very difficult to be represented as an analytical solution, thus we have used three as follows differential equations:

```
dA[1] = -Ka*A[1]
dA[2] = Ka * A[1] - (Cl/V1 + Q/V1) * A[2] + Q/V2 * A[3]
dA[3] =  Q/V1 * A[2] -  Q/V2 * A[3]
```

Where dA[1], dA[2] and dA[3] define the differential equations by demonstrating the rate of change in the amount of both drugs for the two compartments, where Cl/V1 is the rate of elimination (K10), Q/V1 represent the rate of transfer from central compartment to peripheral compartment (K12) and Q/V2 demonstrates the rate of transfer from peripheral compartment to central compartment i.e. K21.

The Isoda function (Attachment 13) is from deSolve [72] package that is used to analyse the amount of glecaprevir and pibrentasvir in each compartment with the time period of 0 to 90 days and interval of 0.02 days. Several arguments are taken by Isoda function.

```
sim.data.df.gle <- lsoda(A_0, TIME, DES, THETAlist,  events = list(data=DOSEdata))

sim.data.df.pib <- lsoda(A_0, TIME, DES, THETAlist,  events = list(data=DOSEdata))
```

In these code, A_0 describe the first values of the differential system, TIME defines the time to calculate the value of A in each compartment, DES function is a function of R that define the differential equations in the model, THETAlist describes the parameter values list that state the DES function and events = list(data=DOSEdata) define the data used for dosing regimen and the frequency of dose intake. Isoda can calculate the amount of drug at every defined time with the incorporation of differential equation system. Only one set of parameter values can be used by Isoda such as one value of every rate constant (Cl, V1, V2, Q and Ka) (Attachment 13). This model can evaluate the effects on population of different values of covariates and dosing regimens and can ultimately predict intervals and mean concentrations. In parameter data frame, parameter sets are arranged in one row to deliver an input to R function, such as “simulate.conc” function consisting of Isoda and input parameters and initial condition expressions. The “simulate.conc” function transfers through each row of parameter sets or data frame in order to calculate the amount of Glecaprevir and Pibrentasvir at the specified time. This process is done by using the “ddply” function from the “plyr” [73] package.

The automatic updating in the model of Shiny might require long time to update the plot upon the input change. Thus, the compiler [53] package in R (as shown in Attachment 13) can increase the speed of the process. Unlike R, compiled function save the code in an executable file of machine instructions while R save the code as text files that slow the speed at runtime. By applying a byte code compiler, benefits are provided by compiled code in R with the compiler package and “cmpfun” function (Attachment 13). This Attachment has the functions that define the differential equations (DES) in the model and simulate.conc is called to solve the system according to the number of individuals (n) defined with every change in widget as they are written for sim.data in the reactive expression. In the compiler package, the “cmpfun” function can be used to compile the functions as stated below (Attachment 13).

simulate.conc.cmpf <- cmpfun(simulate.conc)

As this model was built in shiny application, it can be seen in the user-interface or the application display that different input functions are implicated for different selections such as “numericInput” is used to select the age range of the individuals, for the renal function and gender selection the function “selectInput” is used while to mention the presence of opioids, Asian race and cirrhosis “checkbxInput” functions are used. For the separation of the plots of both drugs and their concentrations-time profiles in both compartments, tabs are created in ui.R of Attachment 13, where three tabs are made with “tabPanel” function named as Glecaprevir (displays the plot of mean concentrations of Glecaprevir in both compartments against time) and named as Pibrentasvir (displays the plot of mean concentrations of Pibrentasvir in both compartments against time), , and also named Pibrentasvir & Glecaprevir tab that shows the plot of the mean concentrations of both drugs in first compartment against time. However, due to some complications faced in this model, another models were built to overcome the problems and for the accuracy of the results. Below are the equations for clearance used in Attachment 13:

Glecaprevir: *Cl <- 1150* exp(-0.330*input$AGE)* exp(1.03*MRI)* exp(0.706*MSRI)* exp(0.530*ESI)* exp(0.814*SEX_C)* exp(0.763*input$C)* exp(0.900*input$O)*

Pibrentasvir: *Cl <- 6340* exp(0.778*SEX_C)* exp(0.988*MRI)* exp(0.918*MSRI)* exp(0.646*ESI)* exp(0.810*input$AR)* exp(-0.148*input$AGE)* exp(0.912*input$C)*

Where the clearance resulted in “zero” with the zero value of any of the covariates, thus each of these covariates was dealt separately in their individual models. MRI = Mild Renal Impairment, MSRI = Moderate + Severe renal Impairment, ESI = End Stage Impairment, AR = Asian Race, SEX_C = Gender, and C = Cirrhosis.

8.4. GLE and PIB modelling (without Shiny)

Further models were created without Shiny to deal with the variabilities separately. The total number of models built was seven, each of them having different covariates and one without any covariate. In these models, the age was calculated with the help of FDA submission report [71] for mavyret and results of clinical trials practised in that report.

Phase 2 studies	Age [Units: Years] Mean (Standard Deviation)
M13-595	54.1 (9.17)
M14-867	54.1 (9.98)
M14-868	< 65 years = 616 >= 65 years = 75
M15-410	55.9 (7.88)

Table 4: Average ages of Phase 2 clinical studies made in FDA submission report for Mavyret® [71]**, accessed in clinical trial website** [74]**.**

The average ages of all the clinical studies for both drugs were accessed from the clinical trials website [74] by entering the clinical trial numbers as mentioned in FDA submission report for Mavyret® [71], where four Phase 2 studies were made (Studies M13-595, M14-867, M14-868 and M15-410) (Table 4) and six Phase 3 studies were made (Studies M13-583, M13-590, M13-594, M14-172, M15-462, and M15-464) (Table 5) [71]. The table of calculations of the mean age from these studies result and the formula used can be seen in (Attachment 21).

Phase 3 studies	Age [Units: Years] Mean (Standard Deviation)
M13-583	52.66 (10.95)
M13-590	50.93 (11.77)
M13-594	46.61 (11.32)
M14-172	60.12 (10.43)
M15-462	57.52 (11.14)
M15-464	57.04 (12.53)

Table 5: Average ages of Phase 3 clinical studies made in FDA submission report for Mavyret® [71]**, accessed in clinical trial website** [74]**.**

- **Model without any covariate**

Model constructed without any covariate is accessible in Attachment 14, in which none of the covariates is included and the population with (n <- 1000) is simulated by setting the values of each covariate equals to "0" in order to predict their absence in the individuals. Here the clearance (CL) for both drugs is written alone such as (**GLE:** *Cl <- 1150) and* (**PIB:** *Cl <- 6340).*

- **Model with mild renal impairment**

Model built with mild renal impairment is presented in Attachment 15, where the input for mild renal impairment (as MRI in attachment 15) is set equal to "1" to show the existence of the covariate, while the rest of them are still equal to "0". The equation for clearance (Cl) possess the values of mild renal impairment according to the Table 2 and Table 3 for each drug which ultimately multiplies the value with "1" and the resulting clearance can be achieved that will affect the concentration-time profiles of the drugs. Clearance in the model is stated as *Cl <- 6340*(0.988*MRI)* for pibrentasvir, and *Cl <- 1150*(1.03*MRI)* for glecaprevir.

- **Model with mild renal impairment and cirrhosis**

Model created with mild renal impairment and cirrhosis is obtainable in Attachment 16, in which the input for both mild renal impairment (MRI) and cirrhosis (C) (in Attachment 16) are set to "1" for counting their presence in the model with the other inputs equal to "0". Here the equation for clearance (Cl) retains the values for mild renal impairment and cirrhosis as illustrated in Table 2 and Table 3 for glecaprevir and pibrentasvir respectively. In the model the clearance is stated as *Cl <- 1150*(1.03*MRI)*(0.763*C)* for glecaprevir, and *Cl <- 6340*(0.988*MRI)*(0.912*C)* for pibrentasvir.

- **Model with moderate + severe renal impairment**

Model produced with moderate plus severe renal impairment is accessible in Attachment 17, in which the input value of moderate + severe renal impairment (as MSRI in model) is set to “1” and rest equal to “0” to calculate the clearance for each drug that has the values of moderate + severe renal impairment in its equations. The equations in the model are; **GLE:** *Cl <- 1150*(0.706*MSRI)* and **PIB:** *Cl <- 6340*(0.918*MSRI).*

- **Model with moderate + severe renal impairment and cirrhosis**

Model built with moderate plus severe renal impairment and cirrhosis is available in Attachment 18, where the input values for moderate + severe renal impairment (as MSRI) and cirrhosis (as C) is equal to “1” to predict the existence of both at the same time, whereas other covariates are neglected by considering them as “0”. The equation for clearance for both drugs consist the values of cirrhosis and moderate + severe renal impairment as shown in Table 2 and Table 3 and are written in the model as *Cl <- 1150*(0.706*MSRI)*(0.763*C)* for glecaprevir and as *Cl <- 6340*(0.918*MSRI)*(0.912*C)* for pibrentasvir.

- **Model with end stage impairment**

Model constructed with end stage impairment is presented in Attachment 19, in which the input for end stage impairment (as ESI in model) is set equal to “1” by considering other inputs absent and equal to “0”. As the clearance will be affected by the end stage impairment, its value will be included in the equation of each drug such as **GLE***: Cl <- 1150*(0.763*C) and* **PIB:** *Cl <- 6340*(0.912*C)* (As shown in the Attachment 19).

- **Model with end stage impairment and cirrhosis**

Model created with end stage renal impairment and cirrhosis is accessible in Attachment 20. In this model the input for both end stage impairment and cirrhosis is considered as “1” as both of them is assumed to be present in the population. Similar to other models, the clearance for each drug will also be affected here. Therefore, the values for end stage impairment and cirrhosis as stated in Table 2 and Table 3 are included in the clearance equation in the model such as **GLE:** *Cl <- 1150*(0.530*ESI)*(0.763*C) and* **PIB:** *Cl <- 6340*(0.646*ESI)*(0.912*C).*

9. Pharmacokinetics parameters calculations

At the end of each of the above Glecaprevir and Pibrentasvir model (Attachment 14-20), a calculation is done to evaluate the AUC (area under the curve) and Cmax (maximum concentration) of each plot. First, the data frames were created for each of the drug comprising the subjects and AUC with an interval of (85-86) day and named as Auc.gle and Auc.pib, later these two data frames (for both drugs) were merged together against subjects and a new data frame was formed (Auc.data.df).

Similarly, two more data frames were created in order to calculate the Cmax of each drug and the data frames created were cmax.gle (for glecaprevir) and cmax.pib (for pibrentasvir). Then both of these data frames were merged together in one data frame with the name of cmax.data.df in which the Cmax of both drugs were calculated against subjects.

At the end, both of the merged data frames of AUC (Auc.data.df) and Cmax (cmax.data.df) were merged in one more new data frame with the subjects that were named as AUC.CMAX on the models. This final data frame (AUC.CMAX) was now comprised of all the required information in one data frame such as AUC and Cmax for both drugs that was later used to make further pharmacokinetic calculations.

Lastly, the required calculations were arithmetic mean, geometric mean, standard deviation and coefficient of variation of AUC and Cmax for both drugs. Thus, a matrix was created with 4 numbers of columns and 4 numbers of rows to calculate each of the required calculation and arrange them in a table. The name given to the matrix in the models was (summary_table), in which the rows were named as Arithmetic mean, Geometric mean, standard deviation and coefficient of variation while the columns were named as ("GLE.CMAX", "PIB.CMAX", "GLE.AUC", "PIB.AUC"). The arithmetic mean was calculated with the “mean” function in R, geometric mean was calculated with “geometric.mean" function, stanadard deviation with “sd” and coefficient of variation by dividing standard deviation by Arithmetic mean. The resultant table was then saved as a csv file in excel file by the command “write.csv(summary_table, "Calculations.csv")”.

III. Results and Conclusions

10. Modelling in R and Shiny

The results and plots obtained at the end of each model built in R and Shiny application are shown below. The order of the model results is maintained according to the models defined in materials and methods.

10.1. Basic code for plotting and PK parameters caculations

Proper log-linear curves were obtained for each of the products where the observed continuous points in the plot signify the data incorporated by a bi-exponential function which can be evaluated by mono-exponential function. Figure 4 shows an example of the plots obtained from the simple concentration-time profiles, whereas rest plots for both test and reference products for each subject can be seen in Attachment 3. Similarly, *Figure 5* shows a perfect curve for log concentration vs time plot for each formulation.

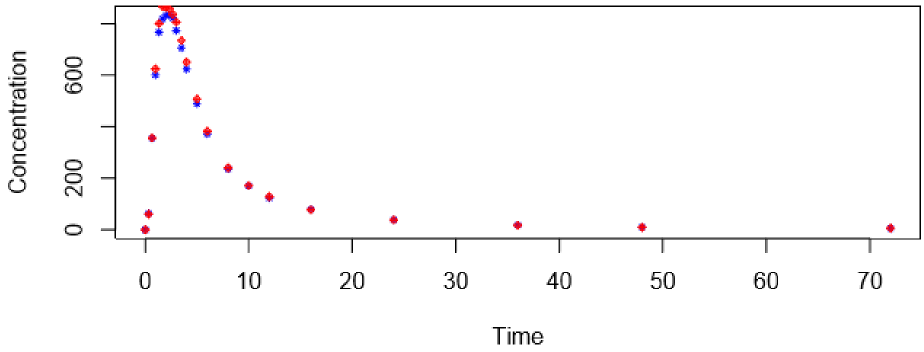


Figure 4: Average of plasma concentrations vs time plot for two drug products (Test and Reference) after oral administration. (Attachment 3)

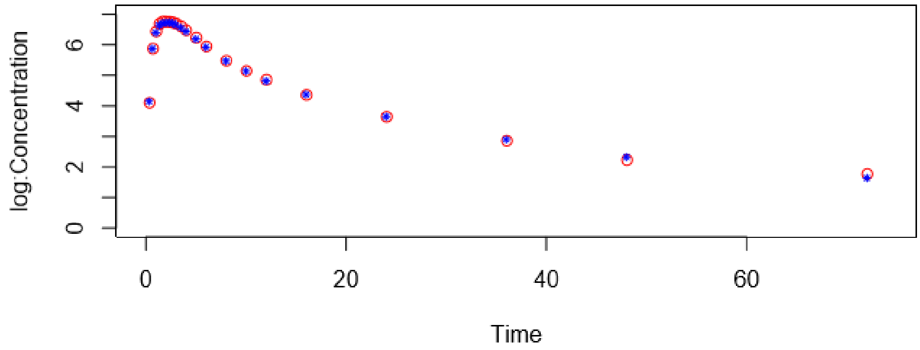


Figure 5: Average of log of concentrations vs time plot for two drug products (Test and Reference) after oral administration. (Attachment 3)

10.2. Model with Ordinary Differential Equation (ODE) in Shiny

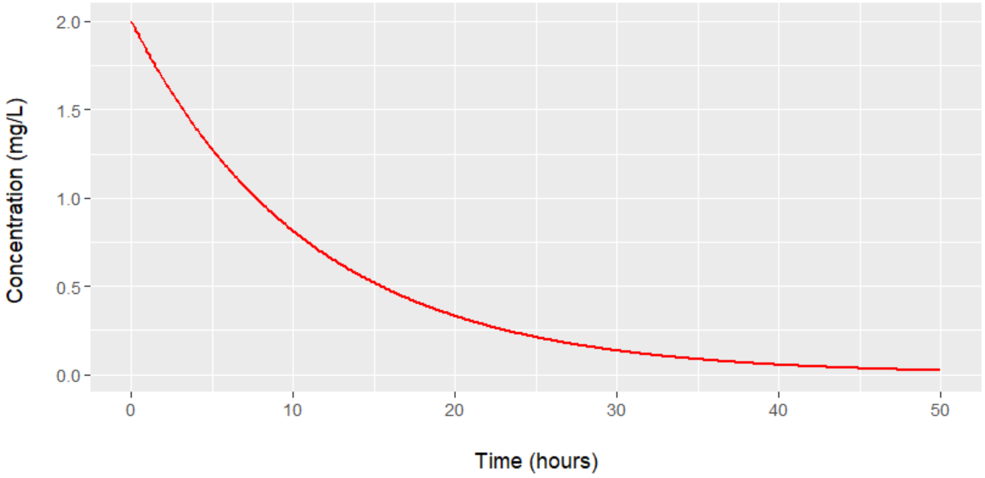


Figure 6: Concentration-time graph plotted in Shiny application after intravenous administration with the addition of Differential equations. (Attachment 5)

The plot shows a perfect curve of concentration-time where the bioavailability (F) of the drug is seen 100%.

10.3. Model with ETA in Shiny

The values of ETA in this model were added to see the difference in the plot and it resulted in a proper curve of concentration-time profile.

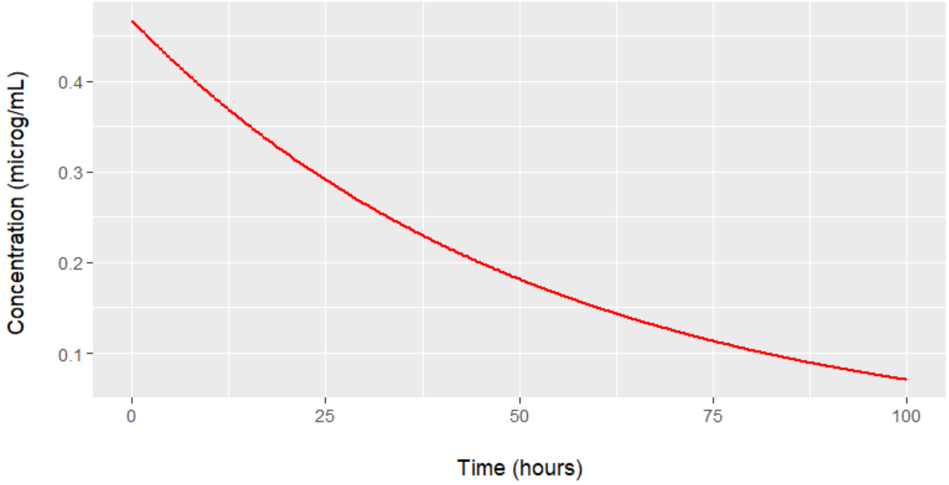


Figure 7: Concentration-time graph plotted in Shiny application after intravenous administration with the addition of ETA (random effects in the model). (Attachment 6)

10.4. Model with Confidence Interval (CI)

This model was built in Shiny application to deal with the values of confidence interval that worked successfully. The shaded ribbons of red colour are showing the area of the confidence interval where the true values of concentration is potentially present.

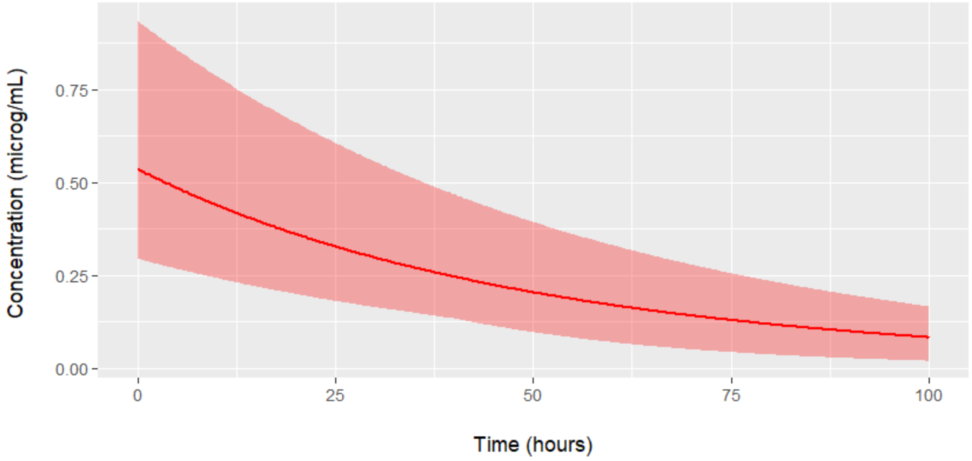


Figure 8: Simple concentration-time graph in Shiny with the calculation of their confidence interval (Attachment 7)

10.5. Model with Slide bars

This model was created in Shiny with the modifications in the previous model and with the addition of slide bars in the user interface of the application where the sliders help to view the instant change in the plot when they are moved from one value to another. The graph shown in *Figure 9* is updated with the change in slide bars shown in *Figure 10*.

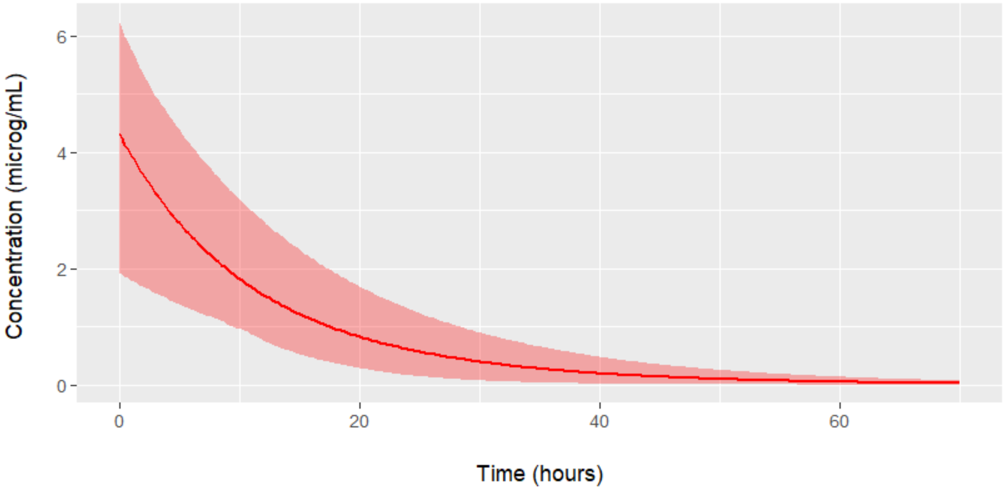
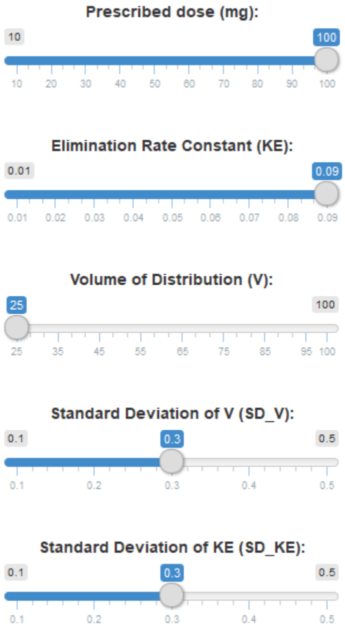


Figure 9: Concentration-time plot of a model in Shiny where slide bars are added to observe the change in graph simultaneously (Attachment 8)



Prescribed dose (mg):
10 100

Elimination Rate Constant (KE):
0.01 0.09

Volume of Distribution (V):
25 100

Standard Deviation of V (SD_V):
0.1 0.3 0.5

Standard Deviation of KE (SD_KE):
0.1 0.3 0.5

Figure 10: Slide bars in the model created to observe the change in plot upon moving the sliders from one value to another (Attachment 8)

10.6. Modelling with the two compartmental models (Attachment 9,10)

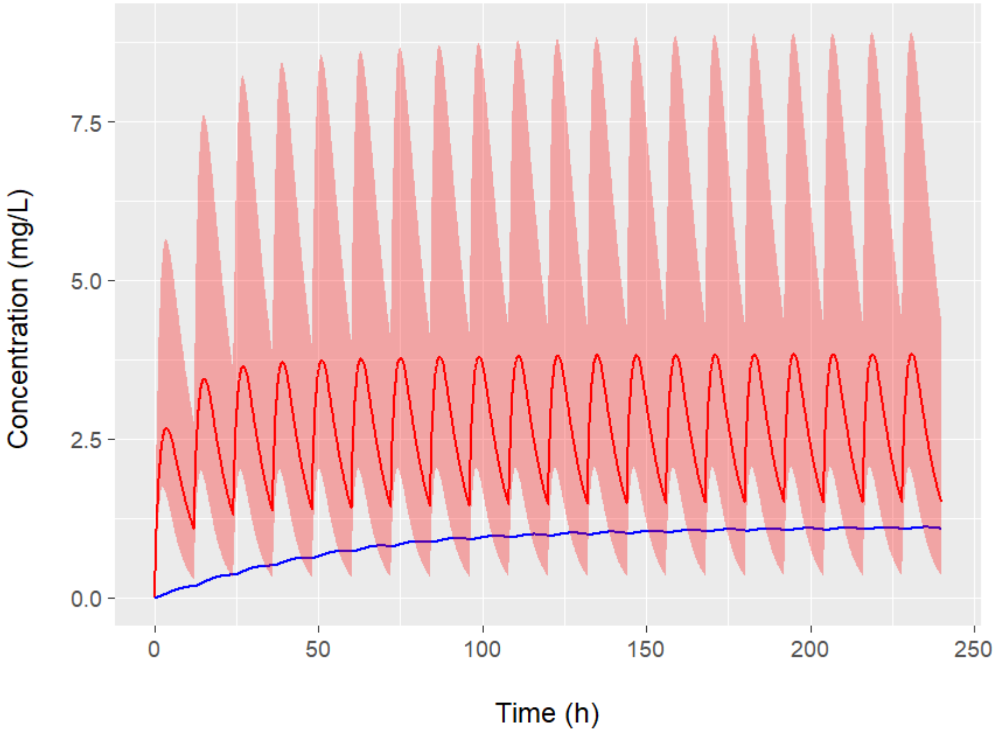


Figure 11: Model created with two compartmental model where concentrations in both compartments are plotted with the confidence interval of the concentration in central compartment

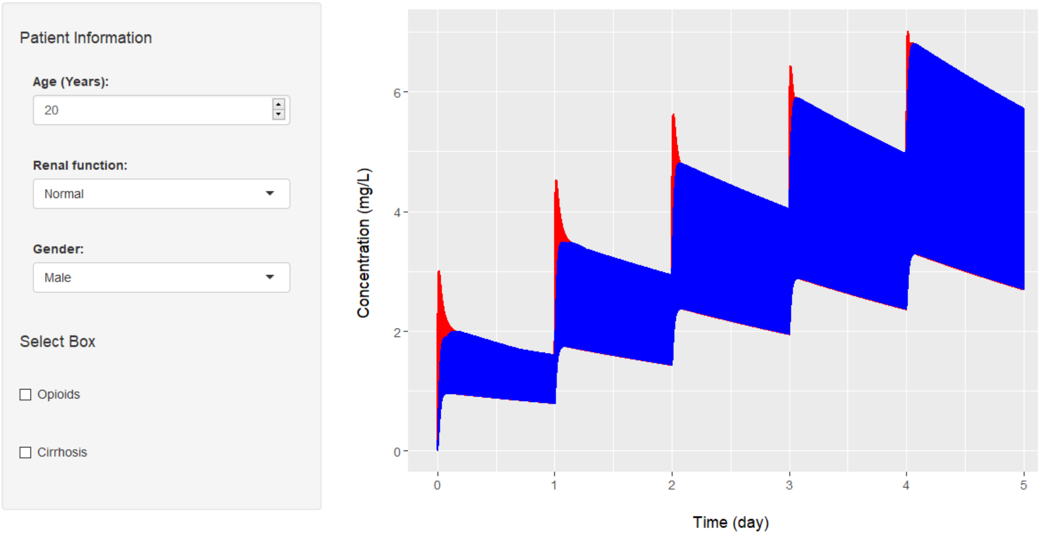
This model was created to practice the plotting of two concentrations at the same time, in the same chart. *Figure 11* shows the two concentrations of one drug that are in central and peripheral compartment, on the other hand the concentration in central compartment also show the confidence interval that is displayed as a shaded region in red colour. The successful plotting of two concentrations in one chart further enabled the plotting of two drug concentrations with their confidence intervals at the same time as illustrated in final modelling.

11. Final modelling

As all of the above models were created to practice for building the final model for the intended drugs (Glecaprevir and Pibrentasvir), further models were initiated that exhibited the following results.

11.1. Glecaprevir (GLE) modelling (Attachment 11)

One of the modelling drugs (Glecaprevir) to be analysed was used to build a model in the initial level of modelling in order to observe it alone and to practice for further combined modelling. The model created was in Shiny where the selection boxes and check boxes were used to create choosing options for age, renal impairment, gender, opioids and cirrhosis. The subsequent plot displayed in the shiny application shows the two concentrations in two compartmental model for Glecaprevir, which show a rise in the initial days but seem to get at steady state levels later on *Figure 12*.



Patient Information

Age (Years): 20

Renal function: Normal

Gender: Male

Select Box

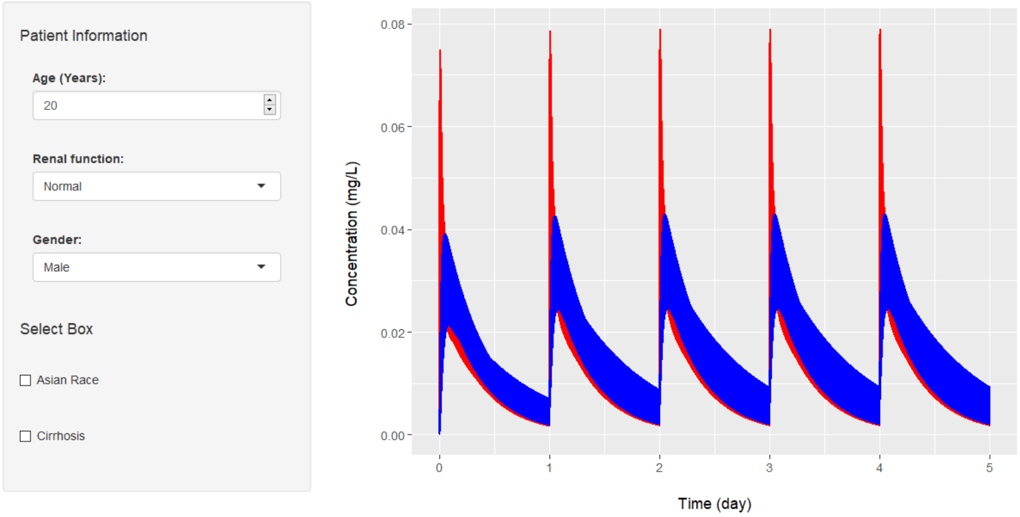
□ Opioids

□ Cirrhosis

Figure 12: This figure shows the screenshoot of the resultant display of user interface used in the building model of Glecaprevir in Shiny (Attachment 11)

11.2. Pibrentasvir (PIB) modelling (Attachment 12)

Further, a model was created with the intended modelling drug (Pibrentasvir) where all the possibilities were practiced to see the resulting display of Shiny application in the browser and the plot of each concentration values of Pibrentasvir in central and peripheral compartment. The selection boxes and check boxes are used to change the desired covariate in the model and observe the updated plot simultaneously. The resultant graph is displaying the rise in concentrations at the initial phase whereas it gets steady state later *Figure 13*.



Patient Information

Age (Years): 20

Renal function: Normal

Gender: Male

Select Box

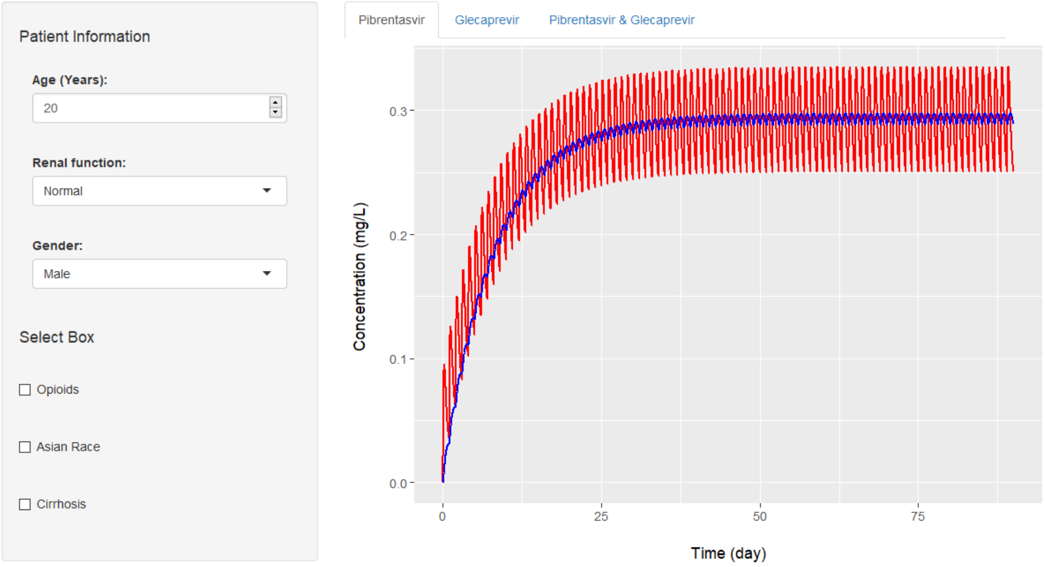
□ Asian Race

□ Cirrhosis

Figure 13: This figure shows the screenshoot of the resultant display of user interface used in the building model of Pibrentasvir in Shiny (Attachment 12)

11.3. GLE and PIB modelling (with Shiny) (Attachment 13)

After building the individual models of Glecaprevir and Pibrentasvir in Shiny, a combined model was created also in Shiny to deal with all the covariates and the drug concentrations in one window. Figure 14 demonstrate the view of the final work in the model where three tabs were created to observe the concentrations of each drug with the change of the given covariates. By changing the tabs, the plot is updated and the concentrations as per drug are shown. The first two tabs of Pibrentasvir and Glecaprevir display the two concentrations of each one in two compartmental model, whereas the third tab show the concentrations of both drugs in the central compartment. All of these concentrations are affected by changing any of the covariate. To practice all of the stated information (Attachment 13) can be used.



Patient Information

Age (Years): 20

Renal function: Normal

Gender: Male

Select Box

□ Opioids

□ Asian Race

□ Cirrhosis

Figure 14: This figure shows the screenshoot of the resultant display of user interface used in the building model of Glecaprevir and Pibrentasvir in Shiny (Attachment 13)

11.4. GLE and PIB modelling (without Shiny)

Following models were created without Shiny in order to reach to accurate values by setting different covariates at one time in a specific group of population:

- **Model without any covariate (Attachment 14)**

	GLE.CMAX	PIB.CMAX	GLE.AUC	PIB.AUC
Arithmetic Mean	0.824095	0.038707	0.261131	0.019951
Geometric Mean	0.822467	0.036797	0.259168	0.019147
Standard Deviation	0.05187	0.01241	0.032251	0.005804
Coefficient of variation	0.062941	0.320616	0.123506	0.290919

Table 6: The table display the calculated values from AUC (mg.h/L) and Cmax (mg/L) of both drugs (Glecaprevir and Pibrentasvir) when no covariate is used in the model

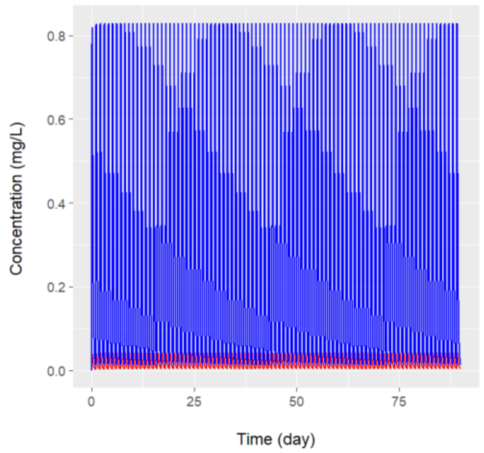


Figure 15: The graph in this figure shows the mean concentrations of Pibrentasvir in red and Glecaprevir in blue in central compartment without using any covariate in the subjects

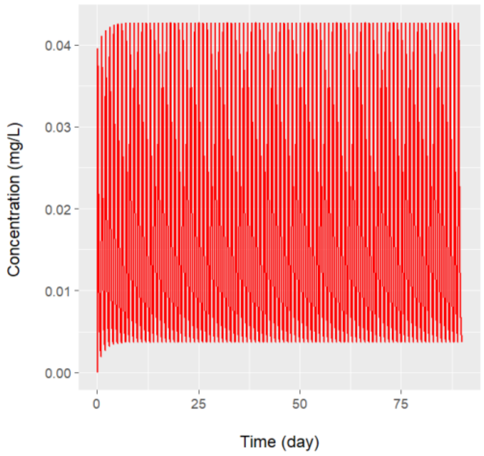


Figure 16: The plot in this figure shows the mean concentration of Pibrentasvir in central compartment without with using any covariate in the subjects

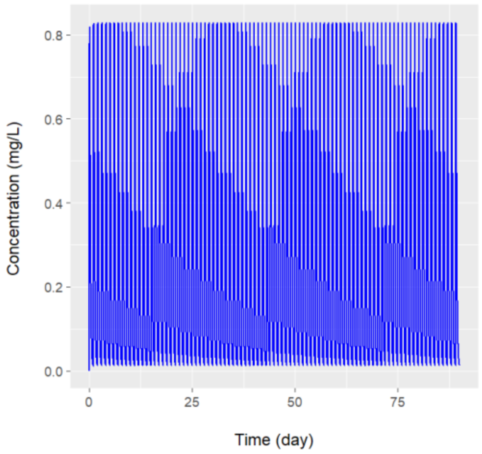


Figure 17: The graph in this figure shows the mean concentration of Glecaprevir only in central compartment without using any covariate in the subjects

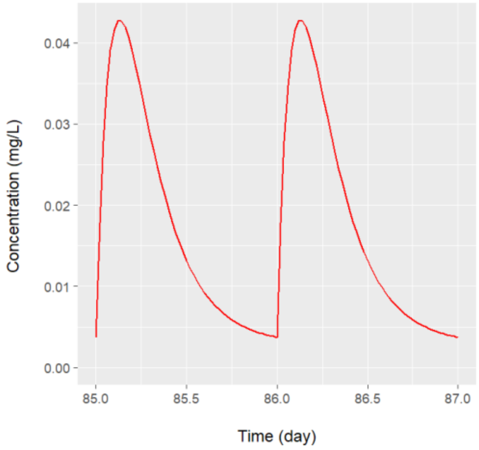


Figure 18: The graph in this figure shows the mean concentrations of Pibrentasvir in central compartment with the interval of 85-87 days and without using any covariate in the subjects

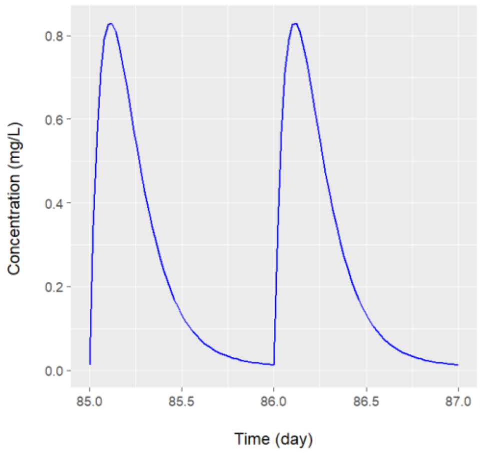
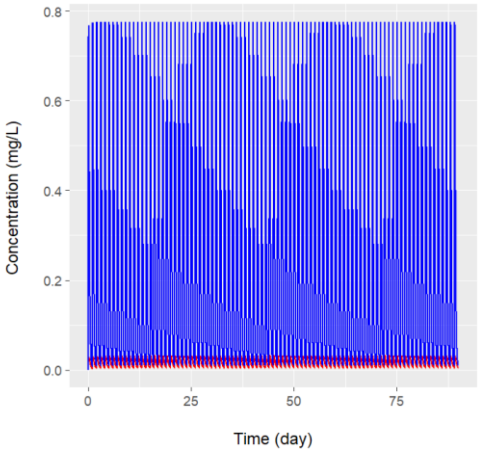


Figure 19: The graph in this figure shows the mean concentrations of Glecaprevir in central compartment with the interval of 85-87 days and without using any covariate in the subjects

- **Model with mild renal impairment (Attachment 15)**



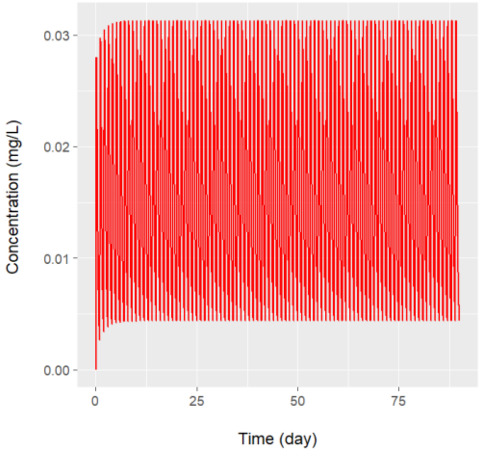


Figure 20: The graph in this figure shows the mean concentrations of Pibrentasvir in red and Glecaprevir in blue in central compartment with the group of patients that are suffering with mild renal impairment

Figure 21: The graph in this figure shows the mean concentrations of Pibrentasvir in central compartment with the group of patients that are suffering with mild renal impairment

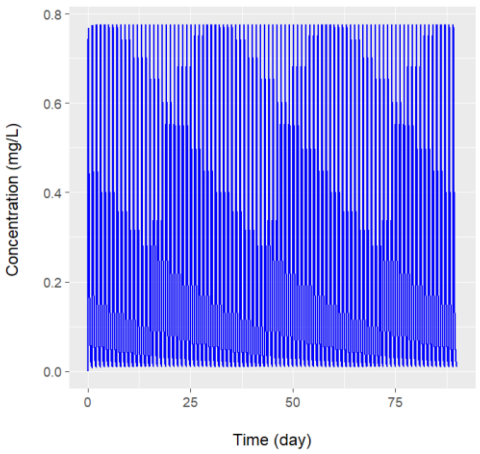


Figure 22: The graph in this figure shows the mean concentrations of Glecaprevir in central compartment with the group of patients that are suffering with mild renal impairment

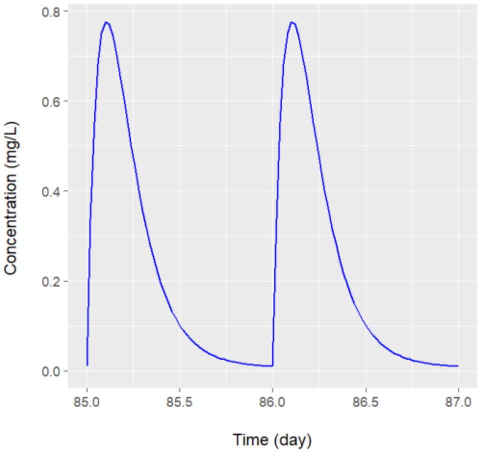


Figure 24: The graph in this figure shows the mean concentration of Glecaprevir in central compartment with the interval of 85-87 days and with the group of patients that are suffering with mild renal impairment

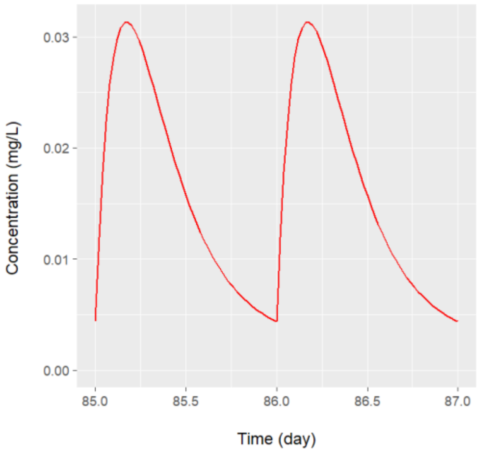


Figure 23: The graph in this figure shows the mean concentration of Pibrentasvir in central compartment with the interval of 85-87 days and with the group of patients that are suffering with mild renal impairment

	GLE.CMAX	PIB.CMAX	GLE.AUC	PIB.AUC
Arithmetic Mean	0.814563	0.038977	0.255169	0.019799
Geometric Mean	0.813065	0.036911	0.253392	0.018982
Standard Deviation	0.049457	0.013318	0.030281	0.005854
Coefficient of variation	0.060716	0.341684	0.118671	0.295689

Table 7: The table display the calculated values from AUC (mg.h/L) and Cmax (mg/L) of both drugs (Glecaprevir and Pibrentasvir) when a model with a patients of mild renal impairment is used

- **Model with mild renal impairment and cirrhosis (Attachment 16)**

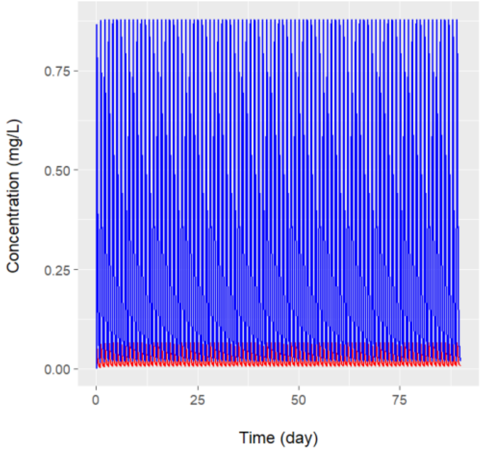


Figure 25: The graph in this figure shows the mean concentrations of Pibrentasvir in red and Glecaprevir in blue in central compartment with the group of patients that are suffering with mild renal impairment and Cirrhosis

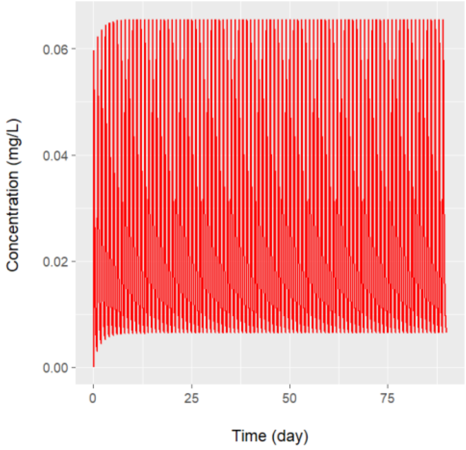


Figure 26: The graph in this figure shows the mean concentrations of Pibrentasvir in central compartment with the group of patients that are suffering with mild renal impairment and Cirrhosis

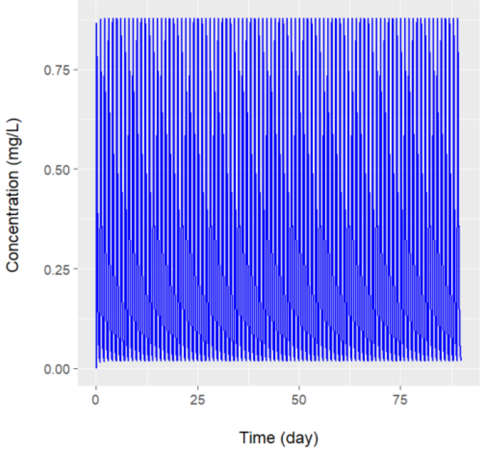


Figure 27: The graph in this figure shows the mean concentrations of Glecaprevir in central compartment with the group of patients that are suffering with mild renal impairment and Cirrhosis

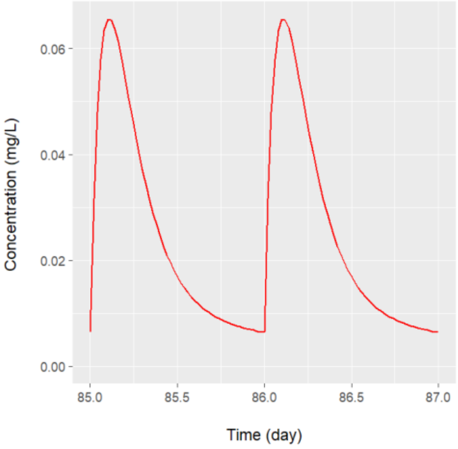


Figure 28: The graph in this figure shows the mean concentration of Pibrentasvir in central compartment with the interval of 85-87 days and with the group of patients that are suffering with mild renal impairment and Cirrhosis

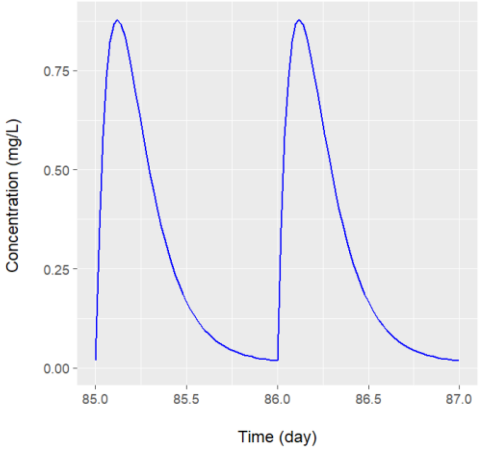


Figure 29: The graph in this figure shows the mean concentration of Glecaprevir in central compartment with the interval of 85-87 days and with the group of patients that are suffering with mild renal impairment and Cirrhosis

	GLE.CMAX	PIB.CMAX	GLE.AUC	PIB.AUC
Arithmetic Mean	0.931279	0.042044	0.333353	0.0216
Geometric Mean	0.929869	0.039949	0.331217	0.020768
Standard Deviation	0.051246	0.013595	0.037857	0.006111
Coefficient of variation	0.055028	0.323359	0.113565	0.282894

Table 8: The table display the calculated values from AUC (mg.h/L) and Cmax (mg/L) of both drugs (Glecaprevir and Pibrentasvir) when a model with a patients of mild renal impairment and cirrhosis is used

- **Model with moderate + severe renal impairment (Attachment 17)**

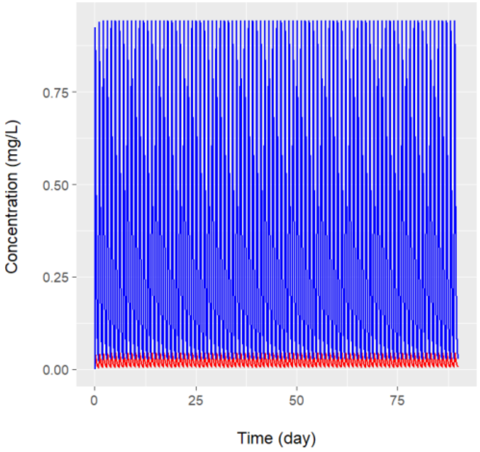


Figure 30: The graph in this figure shows the mean concentrations of Pibrentasvir in red and Glecaprevir in blue in central compartment with the group of patients that are suffering with moderate + severe renal impairment

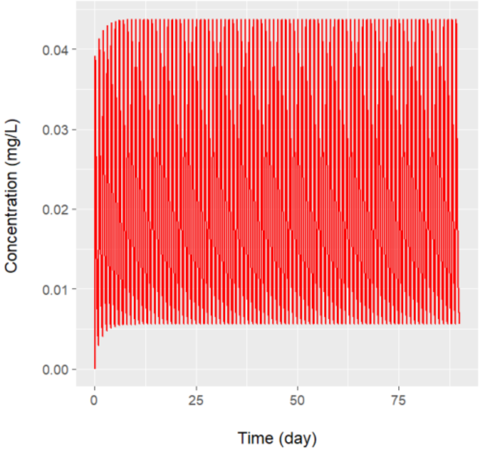


Figure 31: The graph in this figure shows the mean concentrations of Pibrentasvir in central compartment with the group of patients that are suffering with moderate + severe renal impairment

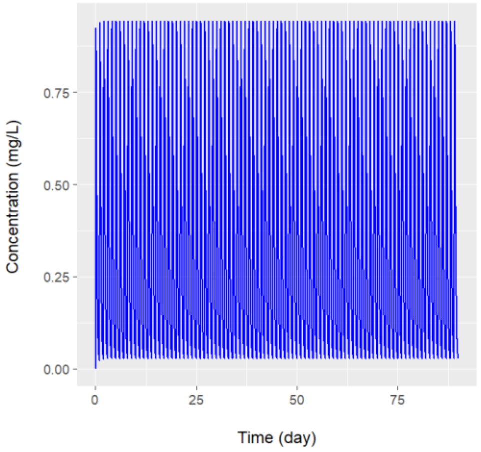


Figure 32: The graph in this figure shows the mean concentrations of Glecaprevir in central compartment with the group of patients that are suffering with moderate + severe renal impairment

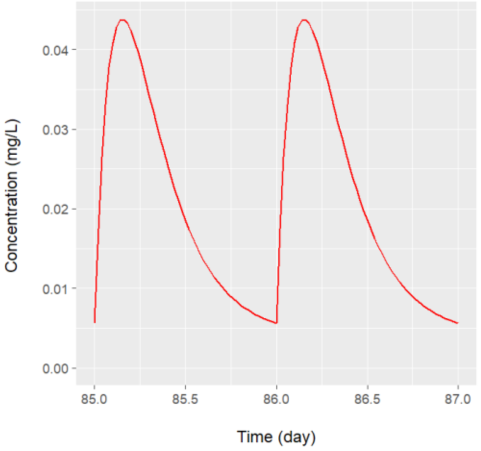


Figure 33: The graph in this figure shows the mean concentration of Pibrentasvir in central compartment with the interval of 85-87 days and with the group of patients that are suffering with moderate + severe renal impairment

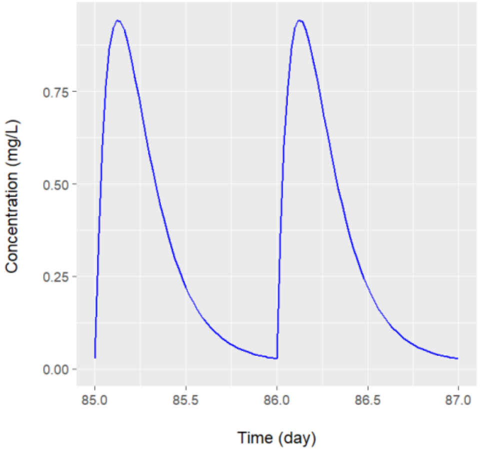


Figure 34: The graph in this figure shows the mean concentration of Glecaprevir in central compartment with the interval of 85-87 days and with the group of patients that are suffering with moderate + severe renal impairment

	GLE.CMAX	PIB.CMAX	GLE.AUC	PIB.AUC
Arithmetic Mean	0.978824	0.041447	0.369546	0.021321
Geometric Mean	0.977349	0.039282	0.367161	0.020451
Standard Deviation	0.053846	0.013902	0.042226	0.006291
Coefficient of variation	0.055011	0.335419	0.114264	0.29507

Table 9: The table display the calculated values from AUC (mg.h/L) and Cmax (mg/L) of both drugs (Glecaprevir and Pibrentasvir) when a model with patients of moderate + severe renal impairment is used

- **Model with moderate + severe renal impairment and cirrhosis (Attachment 18)**

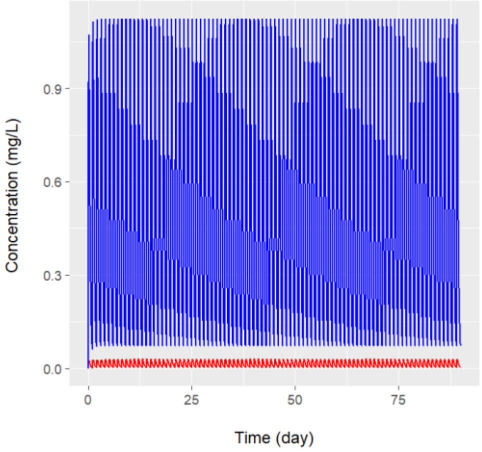


Figure 35: The graph in this figure shows the mean concentrations of Pibrentasvir in red and Glecaprevir in blue in central compartment with the group of patients that are suffering with moderate + severe renal impairment and cirrhosis

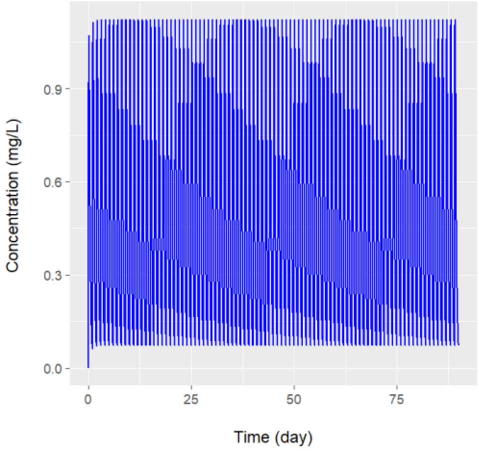


Figure 37: The graph in this figure shows the mean concentrations of Glecaprevir in central compartment with the group of patients that are suffering with moderate + severe renal impairment and cirrhosis

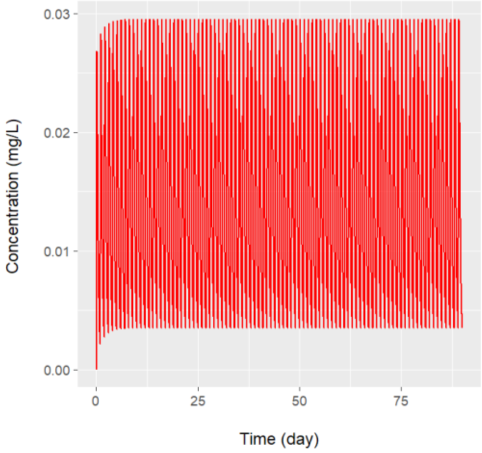


Figure 36: The graph in this figure shows the mean concentrations of Pibrentasvir in central compartment with the group of patients that are suffering with moderate + severe renal impairment and cirrhosis

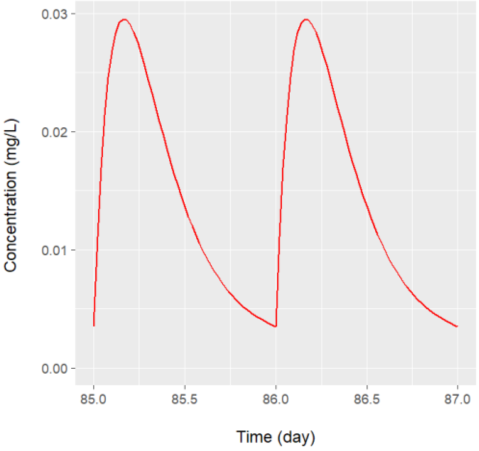


Figure 38: The graph in this figure shows the mean concentration of Pibrentasvir in central compartment with the interval of 85-87 days and with the group of patients that are suffering with moderate + severe renal impairment and cirrhosis

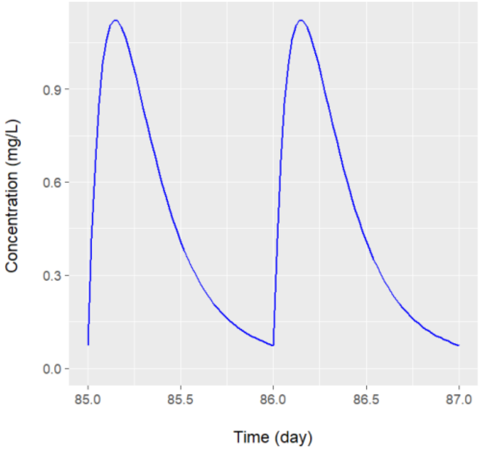


Figure 39: The graph in this figure shows the mean concentration of Glecaprevir in central compartment with the interval of 85-87 days and with the group of patients that are suffering with moderate + severe renal impairment and cirrhosis

	GLE.CMAX	PIB.CMAX	GLE.AUC	PIB.AUC
Arithmetic Mean	1.115564	0.044258	0.484677	0.023477
Geometric Mean	1.113829	0.041907	0.481451	0.022452
Standard Deviation	0.062414	0.015033	0.056307	0.007222
Coefficient of variation	0.055949	0.339669	0.116174	0.30764

Table 10: The table display the calculated values from AUC (mg.h/L) and Cmax (mg/L) of both drugs (Glecaprevir and Pibrentasvir) when a model with a patients of mild renal impairment and cirrhosis is used

- **Model with end stage impairment (Attachment 19)**

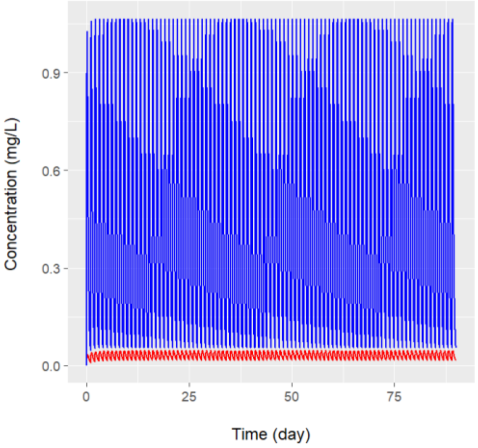


Figure 40: The plot in this figure shows the mean concentrations of Pibrentasvir in red and Glecaprevir in blue in central compartment with the group of patients that are suffering with end stage impairment

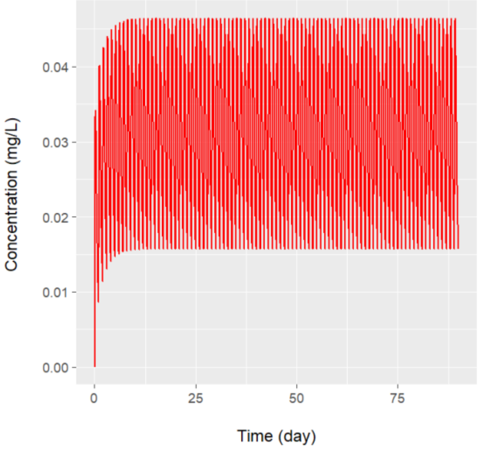


Figure 41: The graph in this figure shows the mean concentrations of Pibrentasvir in central compartment with the group of patients that are suffering with end stage impairment

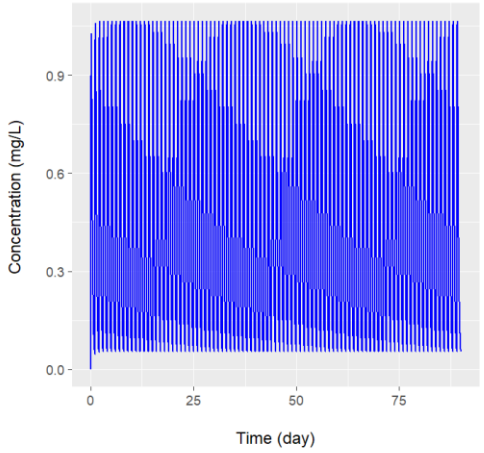


Figure 42: The graph in this figure shows the mean concentrations of Glecaprevir in central compartment with the group of patients that are suffering with end stage impairment

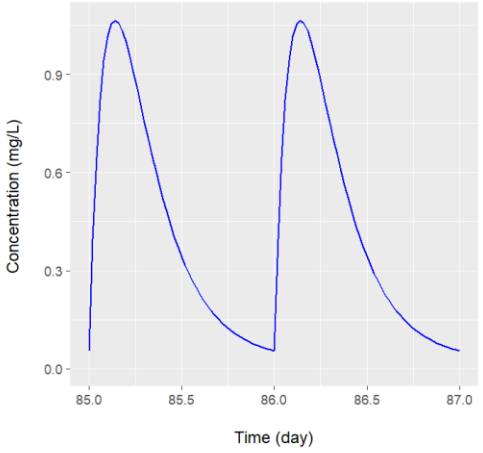


Figure 44: The graph in this figure shows the mean concentration of Glecaprevir in central compartment with the interval of 85-87 days and with the group of patients that are suffering with end stage impairment

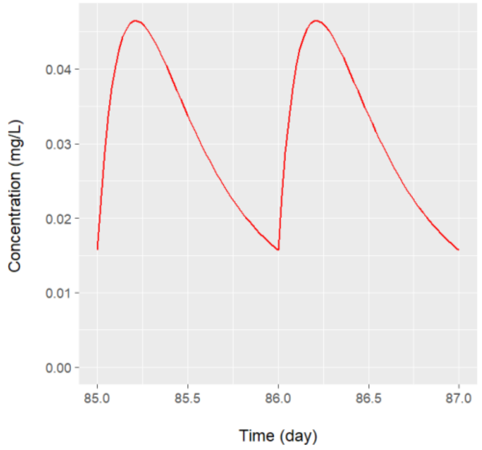


Figure 43: The plot in this figure shows the mean concentration of Pibrentasvir in central compartment with the interval of 85-87 days and with the group of patients that are suffering with end stage impairment

	GLE.CMAX	PIB.CMAX	GLE.AUC	PIB.AUC
Arithmetic Mean	1.127802	0.051868	0.495759	0.03053
Geometric Mean	1.125976	0.049584	0.492348	0.02933
Standard Deviation	0.06429	0.016269	0.058372	0.008838
Coefficient of variation	0.057005	0.313667	0.117742	0.289473

Table 11: The table display the calculated values from AUC (mg.h/L) and Cmax (mg/L) of both drugs (Glecaprevir and Pibrentasvir) when a model with patients end stage impairment is used

- **Model with end stage impairment and cirrhosis (Attachment 20)**

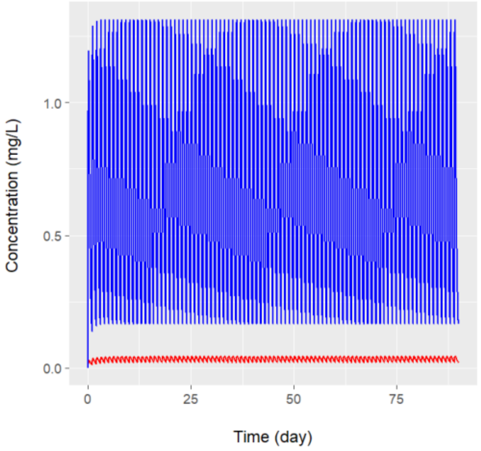


Figure 45: The plot in this figure shows the mean concentrations of Pibrentasvir in red and Glecaprevir in blue in central compartment in the group of patients that are suffering with end stage impairment and cirrhosis

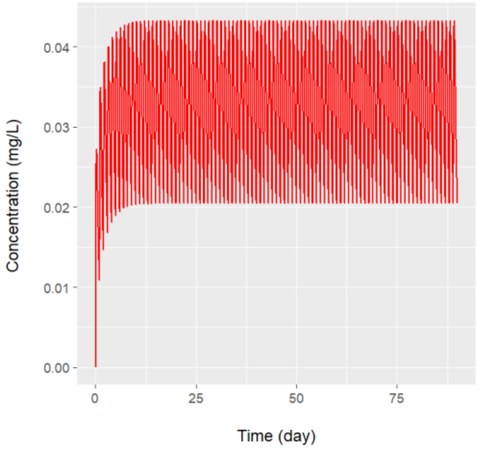


Figure 46: The graph in this figure shows the mean concentrations of Pibrentasvir in central compartment in the group of patients that are suffering with end stage impairment and cirrhosis

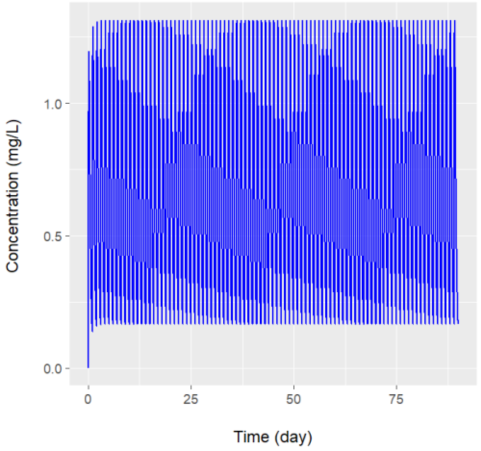


Figure 47: The graph in this figure shows the mean concentrations of Glecaprevir in central compartment in the group of patients that are suffering with end stage impairment and cirrhosis

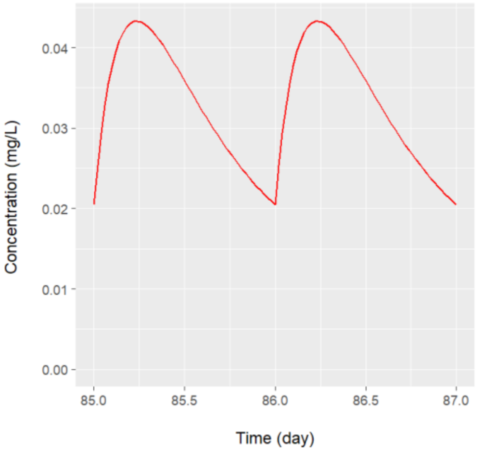


Figure 48: The plot in this figure shows the mean concentration of Pibrentasvir in central compartment with the interval of 85-87 days and in the group of patients that are suffering with end stage impairment and cirrhosis

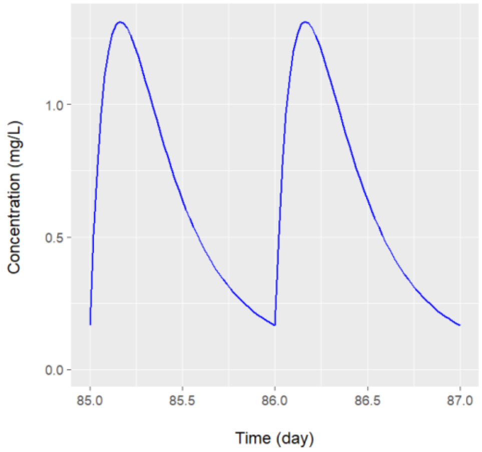


Figure 49: The graph in this figure shows the mean concentration of Glecaprevir in central compartment with the interval of 85-87 days and with the group of patients that are suffering with end stage impairment and cirrhosis

	GLE.CMAX	PIB.CMAX	GLE.AUC	PIB.AUC
Arithmetic Mean	1.28864	0.055244	0.647627	0.03342
Geometric Mean	1.286278	0.052398	0.643145	0.031993
Standard Deviation	0.078388	0.018702	0.07681	0.010143
Coefficient of variation	0.06083	0.338529	0.118602	0.303504

Table 12: The table display the calculated values from AUC (mg.h/L) and Cmax (mg/L) of both drugs (Glecaprevir and Pibrentasvir) when a model with a patients of end stage impairment and cirrhosis is used

All of the above plots of the models with different covariates define the profiles of each of the concentrations almost similarly. As it can be seen that the mean concentrations of both drugs in each compartment show similar profiles except for the two covariates, moderate + severe renal impairment with cirrhosis (*Figure 35*) and end stage impairment with cirrhosis (Figure 45) where the mean concentrations of Glecaprevir show higher profile with more gap in between than the mean concentrations of Pibrentasvir. This difference from other might be resulted due to higher clearance rate as the group of patients with renal impairment and cirrhosis are used in both models. All of the plots (except for the interval of 85-87 days) show unusual profiles that are due to the dense number of predictions of 1000 individuals in 90 days. If we look closely to each of them as shown in interval figures between 85-87 days, they show a proper and nice plotting where the interval is selected at the steady state of plots.

Table 6 to Table 12 show the arithmetic mean, geometric mean, standard deviation and coefficient of variation for the PK parameters AUC and Cmax of each drug's concentrations, based on simulation of PK profiles for 1000 subjects. Tables containing individual data for AUC and Cmax for each drug, with different covariates, can be accessed from Attachment 22 to Attachment 28. These tables can also be obtained by running each of the models (in Attachments) till the end.

IV. Conclusion

Using R for pharmacometric models require skill and Shiny package of R has allowed the access of pharmacometric models up to wide extent such as drug development. In all the process of this study and learning, I was able to build models in R successfully for the drugs with different complexities. Despite the implementation of the population pharmacokinetics models, as described in the FDA submission report for Mavyret® (a fixed dose combination drug product containing Glecaprevir/Pibrentasvir, indicated for the treatment of chronic hepatitis C virus), no further analysis related to the use of pharmacoeconomic methodologies was performed, part due to lack of time and time consuming learning of R basics to R modelling.

V. Bibliography

[1] R. N. Upton and D. R. Mould, "Basic concepts in population modeling, simulation, and model-based drug development: Part 3-introduction to pharmacodynamic modeling methods," *CPT Pharmacometrics Syst. Pharmacol.*, vol. 3, no. 1, pp. 1–16, 2014.

[2] B. Charles, "Population pharmacokinetics: An overview," *Aust. Prescr.*, vol. 37, no. 6, pp. 210–213, 2014.

[3] Aarons L, "Population pharmacokinetics: theory and practice," *Br. J. clin. Pharmac*, vol. 32, pp. 669–670, 1991.

[4] J. G. Wagner, "History of pharmacokinetics," *Pharmacol. Ther.*, vol. 12, no. 3, pp. 537–562, 1981.

[5] L. B. Sheiner, B. Rosenberg, and K. L. Melmon, "Modelling of individual pharmacokinetics for computer-aided drug dosage," *Comput. Biomed. Res.*, vol. 5, no. 5, pp. 441–459, 1972.

[6] D. R. Mould and R. N. Upton, "Basic Concepts in Population Modeling, Simulation, and Model-Based Drug Development," *CPT Pharmacometrics Syst. Pharmacol.*, vol. 1, no. 9, p. e6, 2012.

[7] H. Russo and F. Bressolle, "Pharmacodynamics and pharmacokinetics of thiopental," *Clin. Pharmacokinet.*, vol. 35, no. 2, pp. 95–134, 1998.

[8] L. B. Sheiner and S. L. Beal, "Evaluation of methods for estimating population pharmacokinetic parameters. III. Monoexponential model: Routine clinical pharmacokinetic data," *J. Pharmacokinet. Biopharm.*, vol. 11, no. 3, pp. 303–319, 1983.

[9] L. Sheinerm, "The Population Approach to Pharmacokinetic Data Analysis: Rationale and Standard Data Analysis Methods," *Drug Metab. Rev.*, vol. 1, no. 2, pp. 153–171, 1984.

[10] L. Z. Benet, "General treatment of linear mammillary models with elimination from any compartment as used in pharmacokinetics," *J. Pharm. Sci.*, vol. 61, no. 4, pp. 536–541, 1972.

[11] Y. Cherruault and V. B. Sarin, "A five compartment linear mammillary model," *Kybernetes*, vol. 16, no. 4, pp. 247–250, 1987.

[12] I. Nestorov, "Whole-body physiologically based pharmacokinetic models," *Expert Opin. Drug Metab. Toxicol.*, vol. 3, no. 2, pp. 235–249, 2007.

[13] S. Jambhekar and P. Breen, *Basic Pharmacokinetics*. 2009.

[14] G. Glass, "Primary, Secondary, and Meta-Analysis of Research," *Educ. Res.*, vol. 5, no. 10, pp. 3–8, 1976.

[15] J. W. Mandema, M. Gibbs, R. A. Boyd, D. R. Wada, and M. Pfister, "Model-based meta-analysis for comparative efficacy and safety: Application in drug development and beyond," *Clin. Pharmacol. Ther.*, vol. 90, no. 6, pp. 766–769, 2011.

[16] B. Khoury *et al.*, "Mindfulness-based therapy: A comprehensive meta-analysis," *Clin.*

Psychol. Rev., vol. 33, no. 6, pp. 763–771, 2013.

[17] D. R. Mould, “Models for disease progression: New approaches and uses,” *Clin. Pharmacol. Ther.*, vol. 92, no. 1, pp. 125–131, 2012.

[18] J. L. French and P. Ravva, “When and how should I combine patient- level data and literature data in a meta- analysis ?,” no. June, 2010.

[19] J. F. Bobb, F. Dominici, and R. D. Peng, “A Bayesian model averaging approach for estimating the relative risk of mortality associated with heat waves in 105 U.S. cities,” *Biometrics*, vol. 67, no. 4, pp. 1605–1616, 2011.

[20] N. H. G. Holford and L. B. Sheiner, “Understanding the Dose-Effect Relationship:,” vol. 453, pp. 429–453, 1981.

[21] O. Ovaskainen and B. Meerson, “Stochastic models of population extinction.,” *Trends Ecol. Evol.*, vol. 25, no. 11, pp. 643–52, 2010.

[22] E. I. Ette, P. J. Williams, and J. R. Lane, “Population pharmacokinetics III: Design, analysis, and application of population pharmacokinetic studies,” *Ann. Pharmacother.*, vol. 38, no. 12, pp. 2136–2144, 2004.

[23] V. Perera, M. J. Dolton, A. J. McLachlan, V. J. Carr, and R. O. Day, “Pharmacometrics: An underused resource in Australian clinical research,” *Med. J. Aust.*, vol. 200, no. 2, pp. 82–83, 2014.

[24] J. Doyle, D. Davidson, S. Katz, M. Varela, D. Demeglio, and J. DeCristofaro, “Apnea of prematurity and caffeine pharmacokinetics: Potential impact on hospital discharge,” *J. Perinatol.*, vol. 36, no. 2, pp. 141–144, 2016.

[25] E. Samara and R. Granneman, “Role of Population Pharmacokinetics A Pharmaceutical Industry Perspective,” *Clin. Pharmacokinet.*, vol. 32, no. 4, pp. 294–312, 1997.

[26] H. Sun *et al.*, “Population pharmacokinetics. A regulatory perspective,” *Clin. Pharmacokinet.*, vol. 37, no. 1, pp. 41–58, 1999.

[27] Committee for medicinal products for human use and (CHMP), “Guideline on Reporting the Results of Population Pharmacokinetic Analyses,” *Accessed EMEA website*, no. June 2007, pp. 1–11, 2007.

[28] FDA, “Guidance for Industry Population Pharmacokinetics,” *FDA Guid.*, no. February, p. 31, 1999.

[29] J. P. Patel, B. Green, R. K. Patel, M. S. Marsh, J. G. Davies, and R. Arya, “Population pharmacokinetics of enoxaparin during the antenatal period,” *Circulation*, vol. 128, no. 13, pp. 1462–1469, 2013.

[30] K. Patel, J. A. Roberts, J. Lipman, S. E. Tett, M. E. Deldot, and C. M. Kirkpatrick, “Population pharmacokinetics of fluconazole in critically ill patients receiving continuous venovenous hemodiafiltration: Using Monte Carlo simulations to predict doses for specified pharmacodynamic targets,” *Antimicrob. Agents Chemother.*, vol. 55, no. 12, pp. 5868–5873, 2011.

[31] J. K. Duong *et al.*, “Population pharmacokinetics of metformin in healthy subjects and patients with type 2 diabetes mellitus: Simulation of doses according to renal function,” *Clin. Pharmacokinet.*, vol. 52, no. 5, pp. 373–384, 2013.

[32] B. Charles, R. Norris, X. Xiao, and W. Hague, “Population Pharmacokinetics of Metformin in Late Pregnancy. [Article],” *Ther Drug Monit*, vol. 28, no. 1, pp. 67–72, 2006.

[33] X. Jiang, P. Galettis, M. Links, P. L. Mitchell, and A. J. McLachlan, “Population pharmacokinetics of gemcitabine and its metabolite in patients with cancer: Effect of oxaliplatin and infusion rate,” *Br. J. Clin. Pharmacol.*, vol. 65, no. 3, pp. 326–333, 2008.

[34] L. E. Friberg, G. K. Isbister, L. P. Hackett, and S. B. Duffull, “The population pharmacokinetics of citalopram after deliberate self-poisoning: A Bayesian approach,” *J. Pharmacokinet. Pharmacodyn.*, vol. 32, no. 3–4, pp. 571–605, 2005.

[35] B. G. Charles, S. R. Townsend, P. A. Steer, V. J. Flenady, P. H. Gray, and A. Shearman, “Caffeine citrate treatment for extremely premature infants with apnea: Population pharmacokinetics, absolute bioavailability, and implications for therapeutic drug monitoring,” *Ther. Drug Monit.*, vol. 30, no. 6, pp. 709–716, 2008.

[36] N. Holford, “The Visual Predictive Check Superiority to Standard Diagnostic (Rorschach) Plots Warfarin Predictive Check • Estimate parameters with Warfarin Data Warfarin Turnover lose one dimension (time or prediction) Visual check is more informative and diagnost,” *Pharmacology*, pp. 92019–92019, 2005.

[37] C. Therapeutics, V. Wizemann, and T. M. Pardue, “PERSPEC TIVES Diagnosing Model Diagnostics,” *N. Engl. J. Med. Clin. Pharm. Ther. Gend. Med*, vol. 82, no. 2, pp. 1507–1514, 2007.

[38] S. Reza, “Growing Needs in Drug Industry for NONMEM Programmers Using SAS ®,” pp. 1–11, 2015.

[39] J. R. J. R. Wade, S. S. L. Beal, and N. C. N. C. Sambol, “Interaction between structural, statistical, and covariate models in population pharmacokinetic analysis,” *J. Pharmacokinet. Biopharm.*, vol. 22, no. 2, pp. 165–177, 1994.

[40] B. M. Validation, “Guidance for Industry Bioanalytical Method Validation Guidance for Industry Bioanalytical Method Validation,” *Fda*, no. May, pp. 1–22, 2018.

[41] S. L. Beal, “Ways to fit a PK model with some data below the quantification limit,” *J. Pharmacokinet. Pharmacodyn.*, vol. 28, no. 5, pp. 481–504, 2001.

[42] J. E. Ahn, M. O. Karlsson, A. Dunne, and T. M. Ludden, “Likelihood based approaches to handling data below the quantification limit using NONMEM VI,” *J. Pharmacokinet. Pharmacodyn.*, vol. 35, no. 4, pp. 401–421, 2008.

[43] X. S. Xu, A. Dunne, H. Kimko, P. Nandy, and A. Vermeulen, “Impact of low percentage of data below the quantification limit on parameter estimates of pharmacokinetic models,” *J. Pharmacokinet. Pharmacodyn.*, vol. 38, no. 4, pp. 423–432, 2011.

[44] M. Bergstrand and M. O. Karlsson, “Handling Data Below the Limit of Quantification

in Mixed Effect Models," *AAPS J.*, vol. 11, no. 2, pp. 371–380, 2009.

[45] R. N. Upton, D. J. R. Foster, L. L. Christrup, O. Dale, K. Moksnes, and L. Popper, "A physiologically-based recirculatory meta-model for nasal fentanyl in man," *J. Pharmacokinet. Pharmacodyn.*, vol. 39, no. 5, pp. 561–576, 2012.

[46] Y. Wang, "Derivation of various NONMEM estimation methods," *J. Pharmacokinet. Pharmacodyn.*, vol. 34, no. 5, pp. 575–593, 2007.

[47] T. K. L. Kiang, C. M. . Sherwin, M. G. Spigarelli, and M. H. H. Ensom, "Fundamentals of Population Pharmacokinetic Modelling," *Clin. Pharmacokinet.*, vol. 51, no. 8, pp. 515–525, 2012.

[48] L. Gibiansky, E. Gibiansky, and R. Bauer, "Comparison of Nonmem 7.2 estimation methods and parallel processing efficiency on a target-mediated drug disposition model," *J. Pharmacokinet. Pharmacodyn.*, vol. 39, no. 1, pp. 17–35, 2012.

[49] L. Zhang, M. Pfister, and B. Meibohm, "Concepts and Challenges in Quantitative Pharmacology and Model-Based Drug Development," *AAPS J.*, vol. 10, no. 4, pp. 552–559, 2008.

[50] P. L. Bonate, "What happened to the modeling and simulation revolution?," *Clin. Pharmacol. Ther.*, vol. 96, no. 4, pp. 416–417, 2014.

[51] K. Ito and D. Murphy, "Tutorial: Application of ggplot2 to pharmacometric graphics," *CPT Pharmacometrics Syst. Pharmacol.*, vol. 2, no. 10, pp. 1–16, 2013.

[52] T. Create, E. Data, V. Using, and G. Description, "Package 'ggplot2' Title Create Elegant Data Visualisations Using the Grammar of Graphics," 2018.

[53] T. e. a. m. R Development Core, "R: a Language and Environment for Statistical computing," vol. 1, 2011.

[54] Rs. Inc, "shiny: Web Application Framework for R," *R package version 0.10.1.*, 2014. [Online]. Available: http://cran.r-project.org/package5shiny.

[55] A. Krause and P. J. Lowe, "Visualization and communication of pharmacometric models with Berkeley Madonna," *CPT Pharmacometrics Syst. Pharmacol.*, vol. 3, no. 5, pp. 1–20, 2014.

[56] "Shiny RStudio," *Accessed 18 September 2014*, 2014. [Online]. Available: http://shiny.rstudio.com/.

[57] C. Mueller, C. Schur, and J. O. Connell, "Prescription drug spending: the impact of age and chronic disease status," *Amrerican J. Public Heal.*, vol. 87, no. 10, pp. 1626–1629, 1997.

[58] R. J. G. Arnold and S. Ekins, "Time for cooperation in health economics among the modelling community," *Pharmacoeconomics*, vol. 28, no. 8, pp. 609–613, 2010.

[59] M. Barry and J. Feely, "Pharmacoeconomics in Ireland: Concepts and terminology," *Ir. J. Med. Sci.*, vol. 169, no. 1, pp. 63–64, 2000.

[60] K. Pathak and K. Zaman, "Current Pharmaceutical & Clinical," vol. 4, no. 2, pp. 71–

75, 2014.

[61] N. H. Service and C. Excellence, "Pharmacoeconomics (1) An introduction to health economics How can we meet the ever increasing demand for health care , given our limited resources ? Health economics helps to address this issue . It can be seen as a tool to help prioritise health care the population within the resource constraints that exist .," no. 1, pp. 1–2, 2009.

[62] T. Walley and A. Haycox, "Pharmacoeconomics : basic concepts and terminology," pp. 343–348, 1997.

[63] National Medicines Information Centre, "An Introduction to Pharmacoeconomics," *NMIC Bull.*, vol. 8, no. 5, pp. 1–4, 2002.

[64] L. E. X. Muntjewerf, "Letters To," vol. 11, no. page 30, pp. 258–259, 1997.

[65] W. Society, "Author (s): Daniel M . Keppie Published by : Wiley on behalf of the Wildlife Society Stable URL : http://www.jstor.org/stable/3784964 REFERENCES Linked references are available on JSTOR for this article : You may need to log in to JSTOR to access the li," vol. 34, no. 1, pp. 242–246, 2016.

[66] J. Mason, M. Drummond, and G. Torrance, "Some guidelines on the use of cost effectiveness league tables.," *BMJ*, vol. 306, no. 6877, pp. 570–2, 1993.

[67] M. Gu and C. M. Rice, "Structures of hepatitis C virus nonstructural proteins required for replicase assembly and function," *Curr Opin Virol*, vol. 3, no. 2, 2014.

[68] A. Abutaleb, S. Kottilil, and E. Wilson, "Glecaprevir/pibrentasvir expands reach while reducing cost and duration of hepatitis C virus therapy," *Hepatol. Int.*, vol. 12, no. 3, pp. 214–222, 2018.

[69] D. Interactions, "Glecaprevir / pibrentasvir (Mavyret ™) New Drug Update," vol. 450, no. August, pp. 1–8, 2017.

[70] F. Poordad *et al.*, "Glecaprevir/Pibrentasvir in patients with hepatitis C virus genotype 1 or 4 and past direct-acting antiviral treatment failure," *Hepatology*, vol. 67, no. 4, pp. 1253–1260, 2018.

[71] C. Pharmacology, "Center for Drug Evaluation and Clinical Pharmacology and Biopharmaceutics Review (S) FDA submission Report for Mavyret," *Appl. number 209394Orig1s000*, pp. 1–5, 2009.

[72] K. Soetaert, T. Petzoldt, and R. W. Setzer, "Package deSolve : Solving Initial Value Differential Equations in R," *J. Stat. Softw.*, vol. 33, no. 9, pp. 1–25, 2010.

[73] H. Wickham, "The Split-Apply-Combine Strategy for Data Analysis," *J. Stat. Softw.*, vol. 40, no. 1, 2011.

[74] C. Trial, "No Title." [Online]. Available: https://clinicaltrials.gov/.

VI. Attachments

Following is a zipped folder that contains all the attachments mentioned that are accessible electronically. However, the codes for 2 attachments (Attachment 13) and Attachment 14) can be seen directly in the document to have an idea what codes other attachments contain that can be electronically accessed.

Attachment 13 is related to the R file built in Shiny application of GLE and PIB modelling. Following are the code for the Attachment 13 in which two scripts are present (Server.R and ui.R).

Attachment 14 as shown below is related to the R file that was built for GLE and PIB modelling without any covariate and without Shiny application. Other final attachments such as (Attachment 15-Attachment 20 electronically available) contain similar code to Attachment 13, the only difference is the code for covariates used in each of them and the change in the clearance (CL) equations.

Attachment 23 is also available in the document but in shown partially as it has 1000 subjects and AUC and Cmax is calculated for each of them. This part of the attachment just has only 100 subjects and their AUC and Cmax for each of the drugs (Glecaprevir and Pibrentasvir) where the table presented is the partial result of the model with mild renal impairment. Attachments such as (Attachment 22 – Attachment 28) will illustrate similar tables of calculations but different results due to the differentiation of the covariates in every model.

Electronic supplementary material as Attachments.Zip

Attachments.zip

Attachment 13

- **Server.R file**

```
#Load package libraries
library(shiny)
library(deSolve)
library(ggplot2)
library(plyr)
library(compiler)

#Code for functions and variables which are not reactive (not dependent on "input$X")
#ggplot2 theme
theme_custom <- theme_set(theme_grey(18))
```

```
#Function containing differential equations for amount in each compartment
DES <- function(T, A, THETA) {

  Cl <- THETA[1]
  V1 <- THETA[2]
  V2 <- THETA[3]
  Q <- THETA[4]
  Ka <- THETA[5]

  dA <- vector(length = 3)

  dA[1] = -Ka*A[1]
  dA[2] = Ka * A[1] - (Cl/V1 + Q/V1) * A[2] + Q/V2 * A[3]
  dA[3] =  Q/V1 * A[2] -  Q/V2 * A[3]
  list(dA)
}

#Compile DES function
DES.cmpf <- cmpfun(DES)

#TIME sequence for concentrations to be calculated
TIME <- seq(from = 0, to = 90, by = 0.02)

#TIMElast is used in later functions for assigning dose events
TIMElast <- max(TIME)

#Define user-input dependent functions for output
shinyServer(function(input, output) {

  #Reactive expression to generate the plot, this is called whenever the input changes
  sim.data <- reactive({

    #Collect input from user-widgets
    SEX <- input$SEX
    C <- input$C
    AR <- input$AR
    AGE <- input$AGE
    RI <- input$RI
    O <- input$O

    if (input$RI == 1) {
      MRI  <- 0
      MSRI <- 0
      ESI <- 0
    }

    if (input$RI == 2) {
      MRI  <- 1
      MSRI <- 0
      ESI <- 0
    }
    if (input$RI == 3) {
      MRI  <- 0
      MSRI <- 1
      ESI <- 0
    }

    if (input$RI == 4) {
      MRI  <- 0
      MSRI <- 0
      ESI <- 1
    }

    if (input$SEX == 1) {
      SEX_C <- 0
    }

    if (input$SEX == 2) {
      SEX_C <- 1
    }

    #Function for calculating median, upper and lower confidence intervals for x
    #Where x will be concentrations for GLE and PIB
    sumfuncx <- function(x) {
      stat1 <-  mean(x)
      stat2 <-  quantile(x, probs=0.025, names=F)
      stat3 <-  quantile(x, probs=0.975, names=F)
      result <- c("mean"=stat1, "low"=stat2, "hi"=stat3)
      result
    }

#Equations and values
    Cl <- 6340* exp(0.778*SEX_C)* exp(0.988*MRI)* exp(0.918*MSRI)* exp(0.646*ESI)*
    exp(0.810*input$AR)* exp(-0.148*input$AGE)* exp(0.912*input$C) #L/day
    V1 <- 1380     # (L) (Central Compartment)
```

```
V2 <- 2250     #(L) (peripheral compartment)
Q <- 1660      #(L/Day)
Ka <- 6.13
K12 <- Q/V1
K21 <- Q/V2
K10 <- Cl/V
#Simulate random
n <- 10
par.data <- seq(from = 1, to = n, by = 1)
par.data <- data.frame(par.data)
names(par.data) <- "ID"

#Define population values
POPCl <- Cl
POPV1 <- V1
POPV2 <- V2
POPQ <- Q
POPKa <- Ka

#Define population parameter variability
ETACl <- rnorm(n, mean = 0, sd = 0.289)
ETAV1<- rnorm(n, mean = 0, sd = 0.578)
ETAV2 <- rnorm(n, mean = 0, sd = 0)
ETAQ <- rnorm(n, mean = 0, sd = 0)
ETAKa <- rnorm(n, mean = 0, sd = 0)

#Simulate individual values
par.data$Cl <- POPCl*exp(ETACl)
par.data$V1 <- POPV1*exp(ETAV1)
par.data$V2 <- POPV2*exp(ETAV2)
par.data$Q <- POPQ*exp(ETAQ)
par.data$Ka <- POPKa*exp(ETAKa)

#Input doses specific to dosing frequency
DOSE <- 120
freq <- 1
ndoses <- TIMElast/freq + 1

DOSEdata <- data.frame(var    = rep(1, times = ndoses),
                       time   = seq(0,TIMElast,freq),
                       value  = rep(DOSE, times = ndoses),
                       method = rep("add", times = ndoses))
```

```
simulate.conc <- function(par.data) {

  #Parameter vector
  THETAlist <- c("Cl"= par.data$Cl,
                 "V1"= par.data$V1,
                 "V2"= par.data$V2,
                 "Q"= par.data$Q,
                 "Ka"= par.data$Ka)

  #Set initial conditions in each compartment
  A_0 <- c(A1 = 0, A2 = 0, A3 = 0)

  #Run differential equation solver (deSolve package)
  sim.data.df.pib <- lsoda(A_0, TIME, DES, THETAlist,  events = list(data=DOSEdata))
}

  #Compile simulate.conc function          for pib
  simulate.conc.cmpf <- cmpfun(simulate.conc)
  sim.data.df.pib <- ddply(par.data, .(ID, Cl, V1, V2, Q, Ka), simulate.conc.cmpf)

  #GLE model

  Cl <- 1150* exp(-0.330*input$AGE)* exp(1.03*MRI)* exp(0.706*MSRI)*
  exp(0.530*ESI)* exp(0.814*SEX_C)* exp(0.763*input$C)* exp(0.900*input$O) #L/day
  V1 <- 130      # (L) (Central Compartment)
  V2 <- 39.6     #(L) ()perpheral compartment)
  Q <- 68        #(L/Day)
  K12 <- Q/V1
  K21 <- Q/V2
  K10 <- Cl/V1
  Ka <- 8.63    #(1/day)

  #Simulate random
  par.data <- seq(from = 1, to = n, by = 1)
  par.data <- data.frame(par.data)
  names(par.data) <- "ID"

  #Define population values
  POPCl <- Cl
  POPV1 <- V1
  POPV2 <- V2
  POPQ <- Q
  POPKa <- Ka
```

```
#Define population parameter variability
ETACl <- rnorm(n, mean = 0, sd = 0.118)
ETAV1<- rnorm(n, mean = 0, sd = 0)
ETAV2 <- rnorm(n, mean = 0, sd = 0)
ETAQ <- rnorm(n, mean = 0, sd = 0)
ETAKa <- rnorm(n, mean = 0, sd = 0)

#Simulate individual values
par.data$Cl <- POPCl*exp(ETACl)
par.data$V1 <- POPV1*exp(ETAV1)
par.data$V2 <- POPV2*exp(ETAV2)
par.data$Q <- POPQ*exp(ETAQ)
par.data$Ka <- POPKa*exp(ETAKa)

#Input doses specific to dosing frequency
DOSE <- 300
freq <- 1 #per day
ndoses <- TIMElast/freq + 1

DOSEdata <- data.frame(var    = rep(1, times = ndoses),
                       time   = seq(0,TIMElast,freq),
                       value  = rep(DOSE, times = ndoses),
                       method = rep("add", times = ndoses))

simulate.conc <- function(par.data) {

  #Parameter vector
  THETAlist <- c("Cl"= par.data$Cl,
                 "V1"= par.data$V1,
                 "V2"= par.data$V2,
                 "Q"= par.data$Q,
                 "Ka"= par.data$Ka)

  #Set initial conditions in each compartment
  A_0 <- c(A1 = 0, A2 = 0, A3 = 0)
  sim.data.df.gle <- lsoda(A_0, TIME, DES, THETAlist,  events = list(data=DOSEdata))
}

#Compile simulate.conc function          for gle
simulate.conc.cmpf <- cmpfun(simulate.conc)
sim.data.df.gle <- ddply(par.data, .(ID, Cl, V1, V2, Q, Ka), simulate.conc.cmpf)
```

```
    #Process the simulated output
    sim.data.df.pib <- as.data.frame(sim.data.df.pib)
    sim.data.df.pib$CONC1 <- sim.data.df.pib$A2/sim.data.df.pib$V1
    sim.data.df.pib$CONC2 <- sim.data.df.pib$A3/sim.data.df.pib$V2
    sim.data.df.pib$CONC3 <- sim.data.df.gle$A2/sim.data.df.gle$V1
    sim.data.df.pib$CONC4 <- sim.data.df.gle$A3/sim.data.df.gle$V2
    sim.data.df.pib$DAYS <- sim.data.df.pib$time/24
    sim.data.df.pib <- as.data.frame(sim.data.df.pib)

    #Concentrations in Central compartment for GLE and PIB
    statsCONC1 <- ddply(sim.data.df.pib, .(TIME), function(sim.data.df.pib) sumfuncx(sim.data.df.pib$CONC1))
    names(statsCONC1)[c(2,3,4)] <- c("Pmean","Plow","Phi")
    statsCONC3 <- ddply(sim.data.df.pib, .(TIME), function(sim.data.df.pib) sumfuncx(sim.data.df.pib$CONC3))
    names(statsCONC3)[c(2,3,4)] <- c("Gmean", "Glow", "Ghi")

#Concentrations in Peripheral compartment for GLE and PIB
    statsCONC2 <- ddply(sim.data.df.pib, .(TIME), function(sim.data.df.pib) sumfuncx(sim.data.df.pib$CONC2))
    names(statsCONC2)[c(2,3,4)] <- c("P2mean","P2low","P2hi")
    statsCONC4 <- ddply(sim.data.df.pib, .(TIME), function(sim.data.df.pib.) sumfuncx(sim.data.df.pib$CONC4))
    names(statsCONC4)[c(2,3,4)] <- c("G2mean", "G2low", "G2hi")

    #Combine both datasets
    sim.data.df.pib <- merge(statsCONC1,statsCONC3,by=c("TIME"),all=T)
    sim.data.df.pib <- merge(sim.data.df.pib,statsCONC2,by=c("TIME"),all=T)
    sim.data.df.pib <- merge(sim.data.df.pib,statsCONC4,by=c("TIME"),all=T)

   }) #Brackets closing "reactive" expression

  #Generate a plot of the data
  #Also uses the inputs to build the plot (ggplot2 package)
 output$plotCONC1 <- renderPlot({

   plotobj <- ggplot(sim.data()) +
    geom_line(aes(x = TIME, y = Pmean), colour = "red", size = 1) +
    geom_line(aes(x = TIME, y = P2mean), colour = "blue", size = 1) +
    geom_ribbon(aes(x = TIME, ymin = Plow, ymax = Phi), fill = "red", alpha = 0.3) +
    geom_ribbon(aes(x = TIME, ymin = P2low, ymax = P2hi), fill = "blue", alpha = 0.3) +
    scale_y_continuous("Concentration (mg/L) \n") +
    scale_x_continuous("\nTime (day)")
  print(plotobj)
 })
   output$plotCONC3 <- renderPlot({

    plotobj <- ggplot(sim.data()) +
      geom_line(aes(x = TIME, y = Gmean), colour = "blue", size = 1) +
      geom_line(aes(x = TIME, y = G2mean), colour = "blue", size = 1) +
      geom_ribbon(aes(x = TIME, ymin = Glow, ymax = Ghi), fill = "blue", alpha = 0.3) +
      geom_ribbon(aes(x = TIME, ymin = G2low, ymax = G2hi), fill = "blue", alpha = 0.3) +
      scale_y_continuous("Concentration (mg/L) \n") +
      scale_x_continuous("\nTime (day)")
    print(plotobj)
  })

  output$plotCONC13 <- renderPlot({

    plotobj <- ggplot(sim.data()) +
      geom_line(aes(x = TIME, y = Pmean), colour = "red", size = 1) +
      geom_line(aes(x = TIME, y = Gmean), colour = "blue", size = 1) +
      geom_ribbon(aes(x = TIME, ymin = Plow, ymax = Phi), fill = "red", alpha = 0.3) +
      geom_ribbon(aes(x = TIME, ymin = Glow, ymax = Ghi), fill = "blue", alpha = 0.3) +
      scale_y_continuous("Concentration (mg/L) \n") +
      scale_x_continuous("\nTime (day)")
    print(plotobj)

  })    #Brackets closing "renderPlot" function
 })     #Brackets closing "shinyServer"
```

- **ui.R file**

```
fixedPage(
 #Logo and Application Title
 fixedRow(
  column(10,
      h2("Mavyret Glecaprevir/Pibrentasvir Model", align = "center"), offset = 1)
            ),        #Brackets closing "fixedRow"

  hr(), #Add a break with a horizontal line

      #Sidebar Panel with Widgets
      sidebarLayout(
       sidebarPanel(
        #Heading
        h4("Patient Information"),

  #Select values for rate constant
  fluidPage(

   numericInput("AGE",
           "Age (Years):",
           min = 20,
           max = 60,
           value = 20,
           step = 1),

br(), #To add a break
   selectInput("RI",
          "Renal function:",
          choices = list("Normal" = 1,
                   "Mild impairment" = 2,
                   "Moderate + Severe impairment" = 3,
                   "End Stage impairment" = 4),
          selected = "Normal"),

   br(), #To add a break
   selectInput("SEX",
          "Gender:",
          choices = list("Male" = 1,
                   "Female" = 2),
          selected = "Male")
  ),
  br(), #To add a break

  h4("Select Box"),

  br(), #To add a break
  checkboxInput("O", "Opioids", FALSE),
  verbatimTextOutput("O"),

  br(), #To add a break
  checkboxInput("AR", "Asian Race", FALSE),
  verbatimTextOutput("AR"),

  br(), #To add a break
  checkboxInput("C", "Cirrhosis", FALSE),
    verbatimTextOutput("C"),

    br() #To add a break
         ),        #Brackets closing "siderbarPanel"

    #Plot output
    mainPanel(

      tabsetPanel(type = "tabs",
                  tabPanel("Pibrentasvir", plotOutput("plotCONC1", height = 600, width = 800)),
                  tabPanel("Glecaprevir", plotOutput("plotCONC3", height = 600, width = 800)),
                  tabPanel("Pibrentasvir & Glecaprevir", plotOutput("plotCONC13", height = 600,
width = 800))
                 # tabPanel("Table1", dataTableOutput("table1"))
  )        #Brackets closing "mainPanel"
      )    #Brackets closing "sidebarLayout"
       )  #Brackets closing "fixedPage"
)
```

Attachment 14

```
#Load library packages
library(deSolve)
library(ggplot2)
library(plyr)
library(compiler)
library (PKNCA)
library(psych)

#Code for functions and variables which are not reactive (not dependent on "input$X")
#ggplot2 theme
theme_custom <- theme_set(theme_grey(18))

#Function containing differential equations for amount in each compartment
DES <- function(T, A, THETA) {
  Cl <- THETA[1]
  V1 <- THETA[2]
  V2 <- THETA[3]
  Q <- THETA[4]
  Ka <- THETA[5]

  dA <- vector(length = 3)
  dA[1] = -Ka*A[1]
  dA[2] = Ka * A[1] - (Cl/V1 + Q/V1) * A[2] + Q/V2 * A[3]
  dA[3] =  Q/V1 * A[2] -  Q/V2 * A[3]
  list(dA)
}

#Compile DES function
DES.cmpf <- cmpfun(DES)

#TIME sequence for concentrations to be calculated
TIME <- seq(from = 0, to = 90, by = 0.02)

#TIMElast is used in later functions for assigning dose events
TIMElast <- max(TIME)

#PIB Model
    #Collect input from user-widgets
    SEX <- 0
    C <- 0
    AR <- 0
    AGE <- 54.72
    MRI <- 0
    MSRI <- 0
    ESI <- 0
    O <- 0

    #Function for calculating median, upper and lower confidence intervals for x
    #Where x will be concentrations for GLE and PIB
    sumfuncx <- function(x) {
      stat1 <-  mean(x)
      stat2 <-  quantile(x, probs=0.025, names=F)
      stat3 <-  quantile(x, probs=0.975, names=F)
      result <- c("mean"=stat1, "low"=stat2, "hi"=stat3)
      result
    }

    Cl <- 6340
    V1 <- 1380     # (L) (Central Compartment)
    V2 <- 2250     #(L) (perpheral compartment)
    Q <- 1660      #(L/Day)
    Ka <- 6.13
    K12 <- Q/V1
    K21 <- Q/V2
    K10 <- Cl/V1
```

```
#Simulate random
n <- 1000
par.data <- seq(from = 1, to = n, by = 1)
par.data <- data.frame(par.data)
names(par.data) <- "ID”

#Define population values
POPCl <- Cl
POPV1 <- V1
POPV2 <- V2
POPQ <- Q
POPKa <- Ka

#Define population parameter variability
ETACl <- rnorm(n, mean = 0, sd = 0.289)
ETAV1<- rnorm(n, mean = 0, sd = 0.78)
ETAV2 <- rnorm(n, mean = 0, sd = 0)
ETAQ <- rnorm(n, mean = 0, sd = 0)
ETAKa <- rnorm(n, mean = 0, sd = 0)

#Simulate individual values
par.data$Cl <- POPCl*exp(ETACl)
par.data$V1 <- POPV1*exp(ETAV1)
par.data$V2 <- POPV2*exp(ETAV2)
par.data$Q <- POPQ*exp(ETAQ)
par.data$Ka <- POPKa*exp(ETAKa)

#Input doses specific to dosing frequency
DOSE <- 120 # To change to micrograms
freq <- 1  #DAY
ndoses <- TIMElast/freq + 1
DOSEdata <- data.frame(var    = rep(1, times = ndoses),
                       time   = seq(0,TIMElast,freq),
                       value  = rep(DOSE, times = ndoses),
                       method = rep("add", times = ndoses))

simulate.conc <- function(par.data) {
  #Parameter vector
  THETAlist <- c("Cl"= par.data$Cl,
                 "V1"= par.data$V1,
                 "V2"= par.data$V2,
                 "Q"= par.data$Q,
                 "Ka"= par.data$Ka)
  #Set initial conditions in each compartment
  A_0 <- c(A1 = 0, A2 = 0, A3 = 0)

  #Run differential equation solver (deSolve pacKage)
  sim.data.df.pib <- lsoda(A_0, TIME, DES, THETAlist,  events = list(data=DOSEdata))
}

#Compile simulate.conc functionfor pib
simulate.conc.cmpf <- cmpfun(simulate.conc)
sim.data.df.pib <- ddply(par.data, .(ID, Cl, V1, V2, Q, Ka), simulate.conc.cmpf)

#GLE model
Cl <- 1150
V1 <- 130      # (L) (Central Compartment)
V2 <- 39.6     #(L) (perpheral compartment)
Q <- 68        #(L/Day)
K12 <- Q/V1
K21 <- Q/V2
K10 <- Cl/V1
Ka <- 8.63    #(1/day)

#Simulate random
par.data <- seq(from = 1, to = n, by = 1)
par.data <- data.frame(par.data)
names(par.data) <- "ID"

#Define population values
POPCl <- Cl
POPV1 <- V1
POPV2 <- V2
POPQ <- Q
POPKa <- Ka

#Define population parameter variability
ETACl <- rnorm(n, mean = 0, sd = 0.118)
ETAV1<- rnorm(n, mean = 0, sd = 0)
ETAV2 <- rnorm(n, mean = 0, sd = 0)
ETAQ <- rnorm(n, mean = 0, sd = 0)
ETAKa <- rnorm(n, mean = 0, sd = 0)

#Simulate individual values
par.data$Cl <- POPCl*exp(ETACl)
par.data$V1 <- POPV1*exp(ETAV1)
par.data$V2 <- POPV2*exp(ETAV2)
par.data$Q <- POPQ*exp(ETAQ)
par.data$Ka <- POPKa*exp(ETAKa)

#Input doses specific to dosing frequency
DOSE <- 300
freq <- 1  #DAY
ndoses <- TIMElast/freq + 1

DOSEdata <- data.frame(var    = rep(1, times = ndoses),
                       time   = seq(0,TIMElast,freq),
                       value  = rep(DOSE, times = ndoses),
                       method = rep("add", times = ndoses))

simulate.conc <- function(par.data) {
  #Parameter vector
  THETAlist <- c("Cl"= par.data$Cl,
                 "V1"= par.data$V1,
                 "V2"= par.data$V2,
                 "Q"= par.data$Q,
                 "Ka"= par.data$Ka)

  #Set initial conditions in each compartment
  A_0 <- c(A1 = 0, A2 = 0, A3 = 0)
 sim.data.df.gle <- lsoda(A_0, TIME, DES, THETAlist,  events = list(data=DOSEdata))
 }

#Compile simulate.conc function for gle
simulate.conc.cmpf <- cmpfun(simulate.conc)
sim.data.df.gle <- ddply(par.data, .(ID, Cl, V1, V2, Q, Ka), simulate.conc.cmpf)

#Process the simulated output
sim.data.df.pib <- as.data.frame(sim.data.df.pib)
sim.data.df.pib$CONC1 <- sim.data.df.pib$A2/sim.data.df.pib$V1
sim.data.df.pib$CONC2 <- sim.data.df.pib$A3/sim.data.df.pib$V2
sim.data.df.pib <- as.data.frame(sim.data.df.pib)

#Process the simulated output
sim.data.df.gle <- as.data.frame(sim.data.df.gle)
sim.data.df.gle$CONC3 <- sim.data.df.gle$A2/sim.data.df.gle$V1
sim.data.df.gle$CONC4 <- sim.data.df.gle$A3/sim.data.df.gle$V2
sim.data.df.gle <- as.data.frame(sim.data.df.gle)
```

```
    statsCONC1 <- ddply(sim.data.df.pib, .(TIME), function(sim.data.df.pib)
sumfuncx(sim.data.df.pib$CONC1))
    names(statsCONC1)[c(2,3,4)] <- c("Pmean","Plow","Phi")
    statsCONC3 <- ddply(sim.data.df.gle, .(TIME), function(sim.data.df.gle)
sumfuncx(sim.data.df.gle$CONC3))
    names(statsCONC3)[c(2,3,4)] <- c("Gmean", "Glow", "Ghi")

    sim.data.df <- merge(statsCONC1,statsCONC3,by=c("TIME"),all=T)
    sim.data.df <- as.data.frame(sim.data.df)

   #AUC for GLE
    simulate.conc.cmpf <- cmpfun(simulate.conc)
    testgle.auc <- sim.data.df.gle
    Auc.gle <- ddply(testgle.auc, .(ID), summarise, pk.calc.auc(CONC3, time, interval = c(85,
86)))

    #Change the names of columns
    names(Auc.gle) <- c("Subjects", "GLE.AUC")

#AUC for PIB
    simulate.conc.cmpf <- cmpfun(simulate.conc)
    testpib.auc <- sim.data.df.pib
    Auc.pib <- ddply(testpib.auc, .(ID), summarise, pk.calc.auc(CONC1, time, interval = c(85,
86)))

    #Change the names of columns
names(Auc.pib) <- c("Subjects", "PIB.AUC")

    #Merge both tables of AUC
    Auc.data.df <- merge(Auc.gle,Auc.pib,by=c("Subjects"),all=T)
    Auc.data.df <- as.data.frame(Auc.data.df)

#Cmax for GLE
    simulate.conc.cmpf <- cmpfun(simulate.conc)
    testgle.cmax <- sim.data.df.gle
    cmax.gle <- ddply(testgle.cmax, .(ID), summarise, pk.calc.cmax(CONC3, check = TRUE))

 #Change the names of columns
    names(cmax.gle) <- c("Subjects", "GLE.CMAX")

  #Cmax for PIB
    simulate.conc.cmpf <- cmpfun(simulate.conc)
  testpib.cmax <- sim.data.df.pib
  cmax.pib <- ddply(testpib.cmax, .(ID), summarise, pk.calc.cmax(CONC1, check = TRUE))

  #Change the names of columns
  names(cmax.pib) <- c("Subjects", "PIB.CMAX")

  #Merge both tables of Cmax
  Cmax.data.df <- merge(cmax.gle,cmax.pib,by=c("Subjects"),all=T)
  Cmax.data.df <- as.data.frame(Cmax.data.df)

   #Merge both tables of AUC and Cmax
  AUC.CMAX <- merge(Auc.data.df,Cmax.data.df,by=c("Subjects"),all=T)
  AUC.CMAX <- as.data.frame(AUC.CMAX)

  #Save file in excel
  write.csv(AUC.CMAX, "AUC&CMAX.csv")

#Generate a plot of the data
#Also uses the inputs to build the plot (ggplot2 package)
   plotobj <- ggplot(sim.data.df) +
    geom_line(aes(x = TIME, y = Pmean), colour = "red", size = 1) +
    geom_line(aes(x = TIME, y = Gmean), colour = "blue", size = 1) +
    geom_ribbon(aes(x = TIME, ymin = Plow, ymax = Phi), fill = "red", alpha = 0.3) +
    geom_ribbon(aes(x = TIME, ymin = Glow, ymax = Ghi), fill = "blue", alpha = 0.3) +
    scale_y_continuous("Concentration (mg/L) \n") +
    scale_x_continuous("\nTime (day)")
  print(plotobj)

  plotPIB <- ggplot(sim.data.df) +
    geom_line(aes(x = TIME, y = Pmean), colour = "red", size = 1) +
    geom_ribbon(aes(x = TIME, ymin = Plow, ymax = Phi), fill = "red", alpha = 0.3) +
    scale_y_continuous("Concentration (mg/L) \n") +
    scale_x_continuous("\nTime (day)")
  print(plotPIB)

  plotGLE <- ggplot(sim.data.df) +
    geom_line(aes(x = TIME, y = Gmean), colour = "blue", size = 1) +
    geom_ribbon(aes(x = TIME, ymin = Glow, ymax = Ghi), fill = "blue", alpha = 0.3) +
    scale_y_continuous("Concentration (mg/L) \n") +
    scale_x_continuous("\nTime (day)")
  print(plotGLE)

  plotPIB <- ggplot(sim.data.df) +
    geom_line(aes(x = TIME, y = Pmean), colour = "red", size = 1) +
    geom_ribbon(aes(x = TIME, ymin = Plow, ymax = Phi), fill = "red", alpha = 0.3) +
    scale_y_continuous("Concentration (mg/L) \n") +
    scale_x_continuous("\nTime (day)", limits = c(85,87))
  print(plotPIB)

  plotGLE_SS <- ggplot(sim.data.df) +
    geom_line(aes(x = TIME, y = Gmean), colour = "blue", size = 1) +
    geom_ribbon(aes(x = TIME, ymin = Glow, ymax = Ghi), fill = "blue", alpha = 0.3) +
    scale_y_continuous("Concentration (mg/L) \n") +
    scale_x_continuous("\nTime (day)", limits = c(85,87))
  print(plotGLE_SS)

  #Create a matrix for calculations
  summary_table <- matrix(nrow = 4, ncol = 4)
  colnames(summary_table) <- c( "GLE.CMAX", "PIB.CMAX", "GLE.AUC", "PIB.AUC")
  rownames(summary_table) <- c("Arithmetic Mean", "Geometic Mean", "Standard Deviation", "Coefficient of variation")

  #Arithmetic mean
  summary_table[1,1] <- mean(AUC.CMAX$GLE.CMAX)
  summary_table[1,2] <- mean(AUC.CMAX$PIB.CMAX)
  summary_table[1,3] <- mean(AUC.CMAX$GLE.AUC)
  summary_table[1,4] <- mean(AUC.CMAX$PIB.AUC)

  #Geometric mean
  summary_table[2,1] <- geometric.mean(AUC.CMAX$GLE.CMAX)
  summary_table[2,2] <- geometric.mean(AUC.CMAX$PIB.CMAX)
  summary_table[2,3] <- geometric.mean(AUC.CMAX$GLE.AUC)
  summary_table[2,4] <- geometric.mean(AUC.CMAX$PIB.AUC)

  #Standard Deviation
  summary_table[3,1] <- sd(AUC.CMAX$GLE.CMAX)
  summary_table[3,2] <- sd(AUC.CMAX$PIB.CMAX)
  summary_table[3,3] <- sd(AUC.CMAX$GLE.AUC)
  summary_table[3,4] <- sd(AUC.CMAX$PIB.AUC)

  #Coefficient of variation
  summary_table[4,1] <- summary_table[3,1]/ summary_table[1,1]
  summary_table[4,2] <- summary_table[3,2]/ summary_table[1,2]
  summary_table[4,3] <- summary_table[3,3]/ summary_table[1,3]
  summary_table[4,4] <- summary_table[3,4]/ summary_table[1,4]
```

```
#Save table in excel
write.csv(summary_table, "Calculations.csv")
```

Attachment 23

Subjects	GLE.AUC	PIB.AUC	GLE.CMAX	PIB.CMAX
1	0.231129777	0.016260092	0.77616308	0.031293748
2	0.230022985	0.013360185	0.77430177	0.033052714
3	0.25023479	0.021087669	0.806703267	0.048680278
4	0.240895921	0.014965226	0.79213761	0.018736928
5	0.3020232	0.021302775	0.890079407	0.02715586
6	0.241622726	0.030972387	0.793295303	0.036165043
7	0.313621988	0.016730246	0.906655431	0.037489188
8	0.287553933	0.018062922	0.868390553	0.050471407
9	0.257759024	0.028583166	0.819612045	0.036041073
10	0.249854097	0.022147955	0.806122317	0.027619165
11	0.274080426	0.0116456	0.847071461	0.031931566
12	0.290311741	0.016350804	0.87261652	0.022535211
13	0.275115392	0.023342173	0.84875034	0.028845449
14	0.24945543	0.015151343	0.805512801	0.032970077
15	0.269825087	0.031982742	0.84009299	0.049099457
16	0.324344077	0.019912866	0.921398521	0.04497202
17	0.256419453	0.023278446	0.817271971	0.034269807
18	0.201108411	0.021091734	0.721517866	0.02763191
19	0.235271725	0.02591229	0.78303508	0.027450524
20	0.212431047	0.02864286	0.743197385	0.047660288
21	0.244049401	0.015461923	0.797130606	0.021295313
22	0.225820466	0.012577975	0.767135467	0.041770696
23	0.303629622	0.017494551	0.892416313	0.034311907
24	0.242690056	0.018712045	0.794987855	0.038103372
25	0.250444413	0.041733381	0.807022707	0.048297621
26	0.273158093	0.017778887	0.84556928	0.034645165
27	0.256300519	0.020027619	0.817063542	0.031467938
28	0.223876254	0.009716144	0.763765976	0.015014078
29	0.224012028	0.023404268	0.764002416	0.024856275
30	0.300674339	0.011327397	0.888106601	0.037238514
31	0.213250022	0.023823678	0.744712459	0.04076454
32	0.26108126	0.015922354	0.825357099	0.029563208
33	0.253735723	0.018417665	0.81254221	0.041449846
34	0.215300236	0.022124695	0.748475261	0.043916404
35	0.27279381	0.014354554	0.84497441	0.034647559
36	0.250562998	0.013461904	0.807203275	0.019008784
37	0.298053071	0.019392511	0.884244766	0.030731227
38	0.287854222	0.022841151	0.868852893	0.054032875
39	0.277869576	0.014876782	0.853183745	0.019901229

40	0.246221673	0.012917835	0.800525255	0.034714105
41	0.220647895	0.018251011	0.758093082	0.032082367
42	0.272849084	0.020373946	0.845064729	0.064287124
43	0.221362479	0.013635898	0.759357245	0.046649371
44	0.221716391	0.023940147	0.759981546	0.030809762
45	0.310421121	0.015580158	0.902148651	0.034801635
46	0.271347262	0.021228718	0.842603374	0.051873834
47	0.238051253	0.022744862	0.787565861	0.036887459
48	0.24946776	0.018370698	0.805531669	0.047988421
49	0.279118873	0.039021712	0.855178511	0.043238169
50	0.262553316	0.01220547	0.82787646	0.022316474
51	0.205955337	0.015942728	0.730969451	0.028087519
52	0.245947303	0.015468803	0.800098477	0.035354274
53	0.210487138	0.024757288	0.739573409	0.030864304
54	0.285987014	0.007788972	0.865969283	0.023763348
55	0.230887411	0.011739667	0.775756403	0.030330449
56	0.284761646	0.021516498	0.864065449	0.032627366
57	0.272707432	0.020192935	0.844833225	0.053848735
58	0.250800564	0.025757385	0.807564706	0.042463481
59	0.302380719	0.025077247	0.890600678	0.083006058
60	0.238080426	0.012239564	0.787613077	0.02704338
61	0.251866162	0.028785747	0.809214016	0.040692085
62	0.288483183	0.024232411	0.869819529	0.02907636
63	0.2345135	0.047087968	0.781787984	0.056794457
64	0.291021333	0.02021368	0.873696648	0.068764123
65	0.244160902	0.01643901	0.79730573	0.020728931
66	0.340559104	0.018060936	0.942741369	0.032919173
67	0.291202709	0.033600438	0.873972263	0.062026812
68	0.277646465	0.014391748	0.852826443	0.024889763
69	0.302145822	0.013554629	0.890258269	0.022276331
70	0.286447252	0.025048698	0.866681996	0.03523086
71	0.226127828	0.019461444	0.767664979	0.02786538
72	0.218759601	0.021834388	0.754728922	0.051962123
73	0.202581572	0.018564065	0.724418481	0.044380833
74	0.253303998	0.013428004	0.811776103	0.036244508
75	0.299455423	0.020211695	0.886315447	0.033125813
76	0.226685987	0.020041421	0.76862436	0.033794731
77	0.247044221	0.012566918	0.801801311	0.026452201
78	0.286036409	0.016966472	0.866045836	0.025886331
79	0.289974609	0.026648931	0.872102316	0.029758256
80	0.217921658	0.034638725	0.753224969	0.051556058
81	0.243043297	0.030003904	0.795546055	0.043029186
82	0.239244394	0.015930567	0.789491327	0.055660253
83	0.218106745	0.019385718	0.753557757	0.041624376
84	0.248112103	0.023521307	0.803450392	0.04083192
85	0.317786391	0.01953349	0.912444937	0.032896136

86	0.275346076	0.027984433	0.849123579	0.032514246
87	0.274904051	0.027968547	0.848408088	0.04457222
88	0.259841491	0.02233257	0.823222864	0.044214305
89	0.205214909	0.03210984	0.729542554	0.048813844
90	0.244471849	0.022638372	0.797793603	0.036809408
91	0.236872349	0.019695896	0.78565197	0.058181329
92	0.270643298	0.015459042	0.841444364	0.044928406
93	0.239916239	0.02368553	0.790570475	0.053236094
94	0.307142924	0.020710551	0.897480298	0.039139367
95	0.281163209	0.014900723	0.858421233	0.034412486
96	0.258716802	0.018259254	0.821276806	0.046197232
97	0.229381977	0.02432527	0.773218867	0.067254848
98	0.222722002	0.017618274	0.761748985	0.044205389
99	0.276410952	0.019646609	0.850841975	0.055553737
100	0.205547269	0.017106243	0.7301838	0.028314596